

**LAPORAN PENELITIAN**  
**SKEMA PENELITIAN: PRI-PENUGASAN**

**PENGEMBANGAN DESAIN PRAKTIKUM KOMPUTASI, VISUALISASI,  
OPTIMASI, DAN PEMROGRAMAN BERBASIS CLOUD COMPUTING  
PROGRAM STUDI MATEMATIKA FST UT**



**UNIVERSITAS TERBUKA**

Ketua Peneliti : Sitta Alief Farihati, S.Si., M.Si  
(NIP. 19781222002122002)

Anggota Peneliti :

1. Dra. Asmara Iriani Tarigan, M.Si. (NIP. 196601011997032001)
2. Darsih Idayani, M.Si. (NIP. 198704042020122006)
3. Drs. Hasoloan Siregar, M.Si. (NIP. 196009291989101001)
4. Kani, S.Kom., M.Si. (NIP. 197806142003121002)
5. Siti Umamah Naili Muna, M.Si. (NIP. 199511282022032021)
6. Daru Setiawan (NIM. 042150266)
7. Wahyu Dwi Cahyani (NIM. 042505667)

**FAKULTAS SAINS DAN TEKNOLOGI**  
**UNIVERSITAS TERBUKA**

**2024**

**LEMBAR PENGESAHAN LAPORAN PENELITIAN  
LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT  
UNIVERSITAS TERBUKA**

1	a	Judul Penelitian	:	Pengembangan Desain Praktikum Komputasi, Visualisasi, Optimasi, dan Pemrograman Berbasis Cloud Computing Program Studi Matematika FST UT
	b	Skema Penelitian	:	Penugasan
	c	Rumpun Ilmu	:	Matematika Terapan
	d	Tingkat Kesiapterapan Teknologi (TKT)	:	3
2	Ketua Peneliti			
	a	Nama Lengkap & Gelar	:	Sitta Alief Farihati, S.Si., M.Si
	b	NIP/NIDN	:	197812262002122002/0026127802
	c	Golongan Kepangkatan	:	III.d / Penata Tk. I
	d	Jabatan Akademik	:	Lektor
	e	Fakultas	:	FST
	f	Unit Kerja	:	FST
	g	Program Studi	:	Matematika
3	Anggota Peneliti			
	a	Nama Lengkap & NIP	:	Asmara Iriani Tarigan (NIP. 196601011997032001)
	b	Nama Lengkap & NIP	:	Darsih Idayani (NIP. 198704042020122006)
	c	Nama Lengkap & NIP	:	Hasoloan Siregar (NIP. 196009291989101001)
	d	Nama Lengkap & NIP	:	Kani (NIP. 197806142003121002)
	e	Nama Lengkap & NIP	:	Siti Umamah Naili M. (NIP. 199511282022032021)
	f	Nama Lengkap & NIM	:	Daru Setiawan (NIM. 042150266)
	g	Nama Lengkap & NIM	:	Wahyu Dwi Cahyani (NIM. 042505667)
4	a	Tahun Penelitian	:	2024
	b	Lama Penelitian	:	1 (satu) tahun
5	Biaya Penelitian			
	a	Diusulkan	:	101.096.000
	b	Disetujui	:	92.596.000
6	Sumber Biaya		:	UT

Mengetahui,  
Dekan

Dr. Subekti Nurmawati, M.Si.  
NIP. 196705181991032001

Menyetujui,  
Ketua LPPM-UT,

Prof. Dewi A. Padmo P, M.A., Ph.D.  
NIP. 196107241987102003

Tangerang Selatan, 30 November 2024  
Ketua Peneliti,

Sitta Alief Farihati, S.Si., M.Si.  
NIP. 197812262002122002

Ketua PRI PTJJ,

Prof. Dr. Etty Puji Lestari, S.E., M.Si.  
NIP. 197404162002122001

## ABSTRAK

Mulai tahun akademik 2024/2025 ganjil, UT akan menerapkan kurikulum baru pada seluruh program studi jenjang sarjana. Untuk mempersiapkan kurikulum tersebut, Program Studi Matematika FST telah melakukan penyesuaian kurikulum berdasarkan rekomendasi Himpunan Matematika Indonesia (IndoMS). Salah satu perubahan mendasar pada kurikulum Program Studi Matematika FST adalah perumusan Capaian Pembelajaran Lulusan dan penetapan mata kuliah praktikum. Pada kurikulum baru Program Studi Matematika FST, terdapat dua mata kuliah praktikum yaitu STMA4224 Praktikum Komputasi dan Pemrograman dan STMA4325 Praktikum Visualisasi dan Optimasi. Pada tahun 2024 ini, Program Studi Matematika FST UT harus melakukan persiapan pelaksanaan praktikum dengan membentuk model praktikum komputasi, visualisasi, optimasi, dan pemrograman yang sesuai dengan karakteristik perguruan tinggi jarak jauh. Selain itu, model praktikum ini juga harus mengakomodasi materi dalam praktikum sehingga capaian pembelajaran praktikum terpenuhi. Untuk memenuhi tujuan tersebut, Program Studi Matematika FST melakukan penelitian pengembangan desain praktikum Komputasi, Visualisasi, Optimasi, dan Pemrograman Berbasis Cloud Computing. Dalam penelitian ini, dihasilkan dua buku panduan yaitu panduan Praktikum Komputasi dan Pemrograman dan panduan Praktikum Visualisasi dan Optimasi. Namun desain praktikum yang dirancang ini belum sempurna karena masih ada tahapan pengembangan yang belum terpenuhi. Oleh karena itu, penelitian ini akan dilanjutkan pada tahun 2025.

**Kata kunci:** *cloud computing*, komputasi, kurikulum baru, praktikum, *software*.

## DAFTAR ISI

HALAMAN SAMPUL .....	i
LEMBAR PENGESAHAN LAPORAN PENELITIAN .....	ii
ABSTRAK .....	iii
DAFTAR ISI .....	iv
BAB I. PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian.....	4
BAB II. TINJAUAN PUSTAKA .....	5
2.1 Praktikum dan E-learning .....	5
2.2 Metode ADDIE .....	6
BAB III. METODE PENELITIAN .....	8
BAB IV. HASIL DAN PEMBAHASAN .....	9
4.1 Analisis .....	8
4.2 Desain .....	10
4.3 Pengembangan .....	10
BAB V. KESIMPULAN DAN SARAN .....	12
5.1 Kesimpulan .....	12
5.2 Saran .....	12
DAFTAR PUSTAKA .....	14

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Universitas Terbuka (UT) sebagai perguruan tinggi negeri telah menyelenggarakan pendidikan terbuka dan jarak jauh sejak tahun 1984. Salah satu program studi yang ditawarkan adalah Program Studi Matematika FST UT. Dalam kurun waktu empat dekade tersebut, UT selalu mendesain kurikulum program studi sesuai dengan kebijakan yang diterapkan Kemendikbudristek. Hal ini sebagai bentuk standarisasi pendidikan tinggi nasional dan tanggung jawab UT pada kebutuhan pendidikan masyarakat.

Pada tahun 2020 Kemendikbud telah menetapkan Standar Nasional Pendidikan Tinggi (Permendikbud No. 3 Tahun 2020) yang kemudian diperbaharui pada tahun 2023 tentang Penjaminan Mutu Pendidikan Tinggi (Permendikbudristek No. 53 Tahun 2023). Dalam standar pendidikan tinggi nasional tersebut, perguruan tinggi diminta untuk mengimplementasikan kebijakan Merdeka Belajar Kampus Merdeka dalam kurikulumnya. Oleh karena itu, dalam rangka penyesuaian kurikulum, UT akan menerapkan kurikulum baru pada seluruh program studi jenjang sarjana mulai tahun akademik 2024/2025 ganjil. Untuk mempersiapkan kurikulum tersebut, Program Studi Matematika FST telah melakukan penyesuaian kurikulum berdasarkan rekomendasi IndoMS (Himpunan Matematika Indonesia). Rekomendasi tersebut tercantum dalam SK Presiden IndoMS No. 028/Pres/IndoMS/SK/I/2022 tanggal 3 Januari 2022 [1].

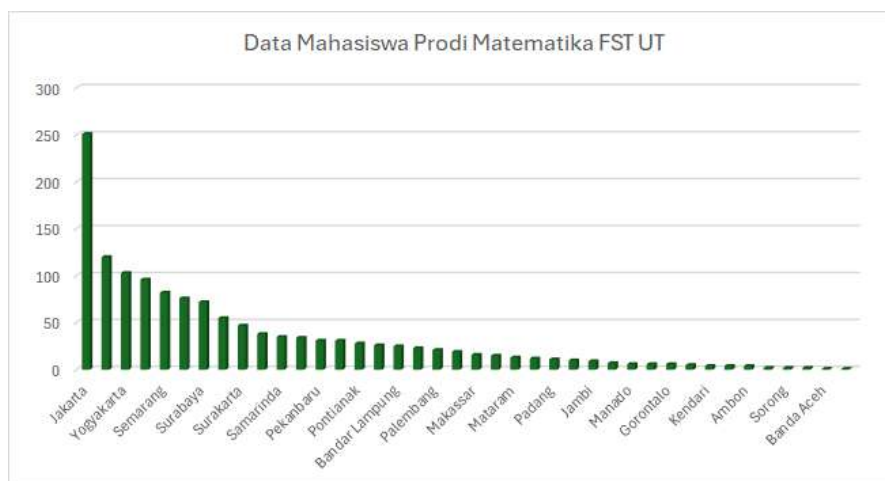
Salah satu perubahan mendasar pada kurikulum Program Studi Matematika FST UT adalah perumusan Capaian Pembelajaran Lulusan (CPL) dan penetapan mata kuliah praktikum. CPL terdiri atas aspek sikap, pengetahuan, keterampilan khusus, dan keterampilan umum. Dalam hal ini, kemampuan berkolaborasi dan kemampuan sebagai pembelajar sepanjang hayat menjadi kata kunci dalam penyusunan CPL. Sementara itu literasi data, literasi teknologi, dan literasi manusia menjadi kunci penyusunan kurikulum [1]. Hal ini terkait dengan semakin pesatnya perkembangan industri dan teknologi di tingkat global yang harus didukung oleh SDM yang cepat tanggap dan adaptif.

Dalam kurikulum baru Program Studi Matematika FST UT, terdapat dua mata kuliah praktikum yaitu STMA4224 Praktikum Komputasi dan Pemrograman dan STMA4325 Praktikum Visualisasi dan Optimasi. Praktikum Komputasi dan Pemrograman ditawarkan pada semester 4 (empat) sedangkan Praktikum Visualisasi dan Optimasi ditawarkan pada semester 6 (enam). Pada mata kuliah Praktikum Komputasi dan Pemrograman, mahasiswa diminta

untuk mempraktekkan cara instalasi *software* atau bahasa pemrograman Java, Python, dan Matlab, kemudian mempraktekkan penggunaan tipe data, variabel, dan operator, perulangan dan kondisi, array dan *method / function*, serta pembuatan program penyelesaian akar persamaan, sistem persamaan linear, interpolasi polinomial dan masalah nilai awal menggunakan bahasa pemrograman Java, Python, dan Matlab. Ketiga bahasa pemrograman tersebut dipilih karena gratis dan mudah diakses oleh mahasiswa melalui internet. Pada matakuliah Praktikum Visualisasi dan Optimasi, mahasiswa diminta memvisualisasikan fungsi dan graf menggunakan GeoGebra, serta memvisualisasikan data menggunakan Ms. Excel. Selain itu, mahasiswa juga diminta mempraktekkan penyelesaian masalah optimasi dalam pemrograman matematika menggunakan GeoGebra dan Solver yang meliputi masalah pemrograman linear, transportasi, penugasan, dan pemrograman tak linear. Kedua *software* ini dipilih karena gratis dan dapat diakses melalui internet.

Mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman dan STMA4325 Praktikum Visualisasi dan Optimasi merupakan kegiatan belajar yang memberikan pengalaman praktis kepada mahasiswa untuk memperdalam pengetahuan dan keterampilan dengan menggunakan teknologi. Dalam praktikum, mahasiswa akan memperoleh pendampingan dan umpan balik terstruktur dari dosen pengampu atau pembimbing praktikum. Oleh karena itu, dalam proses pembelajaran matakuliah STMA4224 Praktikum Komputasi dan Pemrograman dan STMA4325 Praktikum Visualisasi dan Optimasi, mahasiswa akan menjalani praktikum terbimbing sehingga mahasiswa diharapkan mempunyai keterampilan komputasi dalam menyelesaikan masalah matematis dengan menggunakan *software* tertentu.

Proses praktikum terbimbing pada perguruan tinggi jarak jauh tidaklah mudah karena perlu adanya komitmen yang kuat dari mahasiswa dan pendeskripsian mekanisme serta strategi pelaksanaan praktikum yang lebih detail. Untuk dapat menyusun mekanisme dan strategi praktikum yang tepat, perlu dipertimbangkan masalah ketersebaran mahasiswa, fasilitas penunjang, dan SDM pendukung. Berdasarkan data registrasi <https://registrasi.ut.ac.id> tanggal 15 Maret 2024, berikut ini grafik ketersebaran mahasiswa aktif Program Studi Matematika FST tahun 2023/2024 genap.



Gambar 1. Ketersebaran mahasiswa aktif Program Studi Matematika FST tahun 2023/2024 genap

Terkait fasilitas penunjang, Program Studi Matematika FST tidak memiliki laboratorium komputer khusus yang dapat digunakan oleh dosen maupun mahasiswa sebagai tempat praktikum komputasi. Namun, Program Studi Matematika dapat memanfaatkan fasilitas *Cloud Computing* yang dimiliki UT sebagai sarana praktikum komputasi. Disisi lain, UT daerah mempunyai fasilitas komputasi berupa ruang ujian *online* di kantor UT daerah atau di gedung mitra UT. Namun, fasilitas tersebut belum diperiksa dan dicek ketersediaan alat dan *support system*-nya. Adapun masalah SDM pendukung, selain memerlukan dosen atau tutor, pelaksanaan praktikum juga memerlukan laboran atau petugas IT. Oleh karena itu, Prodi Matematika FST UT perlu melakukan persiapan pelaksanaan praktikum sehingga terbentuk model praktikum komputasi, visualisasi, optimasi, dan pemrograman yang sesuai dengan karakteristik perguruan tinggi jarak jauh. Selain itu, penentuan desain praktikum juga harus mengakomodasi materi dalam praktikum sehingga capaian pembelajaran praktikum terpenuhi.

## 1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, maka rumusan masalah penelitian ini adalah bagaimana desain praktikum untuk kedua mata kuliah praktikum Program Studi Matematika FST UT?

## 1.3 Tujuan Penelitian

Berdasarkan rumusan masalah, tujuan penelitian ini adalah mengembangkan desain praktikum untuk kedua mata kuliah praktikum Program Studi Matematika FST UT.

#### **1.4 Manfaat Penelitian**

Hasil penelitian ini diharapkan dapat memberikan manfaat bagi:

1. Program studi, sebagai penyiapan pelaksanaan praktikum komputasi di Prodi Matematika FST UT.
2. Peneliti, sebagai sarana implementasi menulis buku praktikum yang merupakan salah satu bentuk kegiatan pengajaran.



## BAB II TINJAUAN PUSTAKA

### 2.1 Praktikum dan E-learning

Menurut Kamus Besar Bahasa Indonesia, praktikum adalah bagian dari pengajaran yang bertujuan agar mahasiswa mendapat kesempatan untuk menguji dan mengaplikasikan dalam keadaan nyata apa yang diperoleh dari teori. Oleh karena itu praktikum merupakan bagian penting dalam proses pembelajaran khususnya bidang terapan. Di era digital saat ini, keterampilan menggunakan teknologi menjadi salah satu kompetensi yang wajib dimiliki oleh setiap lulusan pendidikan tinggi. Oleh karena itu, pembiasaan menggunakan teknologi sederhana berupa gadget dan komputer sudah harus dilakukan oleh perguruan tinggi maupun program studi sejak awal mulai studi. Dalam hal ini, salah satu bentuk praktis yang bisa diberikan oleh program studi adalah praktikum komputasi dan penggunaan e-learning (*electronic learning*) atau dikenal sebagai LMS (*Learning Management System*).

Terkait dengan praktikum komputasi, pada tahun 2021, Ranius [2] melakukan penelitian tentang Pengembangan Model Pembelajaran Praktikum Algoritma dan Pemrograman Berbasis Virtual. Penelitian ini bertujuan untuk meningkatkan proses belajar dengan membuat model pengembangan pembelajaran praktikum komputer matakuliah Algoritma dan Pemrograman Pascal secara virtual. Dari penelitian tersebut disimpulkan bahwa (1) Model ini memberikan lebih banyak waktu kepada mahasiswa untuk belajar selain dari waktu perkuliahan di laboratorium karena mahasiswa dapat belajar diluar jadwal perkuliahan dan dapat dilakukan tanpa batas waktu dan tempat (2) Setelah dilakukan uji efektivitas model dengan melakukan perbandingan tes diawal dan diakhir proses pembelajaran, hasilnya menunjukkan adanya peningkatan pencapaian hasil belajar; (3) Model pembelajaran berbasis virtual ini akan mempermudah dosen dalam melakukan pembelajaran; (4) Model ini memberikan lebih banyak waktu kepada mahasiswa untuk belajar selain dari waktu perkuliahan.

Terkait dengan penggunaan e-learning dalam pembelajaran, e-learning kini telah didukung oleh layanan dari *cloud computing*. E-learning merupakan cara dalam proses belajar mengajar yang menggunakan media elektronik khususnya internet sebagai sistem pembelajarannya [3]. E-learning juga dikatakan sebagai media pembelajaran yang menggunakan perangkat teknologi berupa *web-based*, *web-distributed*, *web-capable* [4]. Adapun *cloud computing* merupakan media yang memungkinkan akses ke jaringan dan

aplikasi secara mudah. *Cloud computing* terdiri dari dua kata yaitu *cloud* yang berarti awan dan *computing* yang berarti komputasi. *Cloud* dapat dikatakan sebagai internet yang merupakan suatu lingkungan yang cukup besar yang didalamnya terdapat komputer yang saling berhubungan [5]. Oleh karena itu, e-learning dan *cloud computing* saling terkait. Dengan menerapkan konsep *cloud computing* diharapkan dapat meringankan biaya operasional dalam membangun sistem e-learning [6].

## **2.2 Metode ADDIE**

Metode ADDIE adalah kerangka kerja yang umum digunakan dalam pengembangan pembelajaran. Nama ADDIE adalah akronim dari lima tahap utamanya, yaitu *Analysis* (Analisis), *Design* (Desain), *Development* (Pengembangan), *Implementation* (Implementasi), dan *Evaluation* (Evaluasi). Setiap tahap berkontribusi dalam menciptakan program pembelajaran yang terstruktur, efektif, dan terukur. Berikut ini penjelasan dari setiap tahapan dalam metode ADDIE.

### **1. *Analysis* (Analisis)**

Pada tahap ini, dilakukan identifikasi kebutuhan pembelajaran, masalah yang dihadapi, dan sasaran pembelajaran. Adapun aktivitas yang dilakukan adalah (1) menganalisis kebutuhan pembelajaran, (2) mengidentifikasi tujuan pembelajaran, dan menilai sumber daya yang tersedia. Hasil yang diharapkan adalah diperolehnya pemahaman mendalam tentang apa yang diperlukan untuk mencapai tujuan pembelajaran.

### **2. *Design* (Desain)**

Pada tahap ini, dilakukan perencanaan terperinci untuk pelaksanaan program pembelajaran. Adapun aktivitas yang dilakukan adalah (1) menentukan tujuan pembelajaran yang spesifik, (2) merancang struktur kurikulum atau modul, dan (3) memilih metode pengajaran dan media pembelajaran. Hasil yang diharapkan adalah *blueprint* atau rancangan pembelajaran.

### **3. *Development* (Pengembangan)**

Pada tahap ini, rancangan diwujudkan ke dalam bentuk produk atau materi pembelajaran. Aktivitas yang dilakukan adalah (1) membuat konten pembelajaran, seperti video, modul, atau aplikasi, (2) melakukan uji coba kecil terhadap materi, dan (3) merevisi produk berdasarkan masukan. Hasil yang diharapkan adalah materi atau alat pembelajaran yang siap digunakan.

#### **4. *Implementation* (Implementasi)**

Pada tahap ini, dilakukan peluncuran program pembelajaran kepada target peserta. Aktivitas yang dilakukan adalah (1) menyampaikan materi kepada peserta, (2) mengelola pelaksanaan pembelajaran. Hasil yang diharapkan adalah pelatihan dilakukan dan peserta mulai belajar menggunakan produk.

#### **5. *Evaluation* (Evaluasi)**

Pada tahap ini, dilakukan pengukuran efektivitas pembelajaran dan menilai apakah tujuan telah tercapai. Adapun aktivitas yang dilakukan adalah evaluasi formatif (dilakukan selama proses untuk perbaikan) dan evaluasi sumatif (dilakukan di akhir pembelajaran untuk mengukur hasil keseluruhan). Hasil yang diharapkan adalah penilaian tentang keberhasilan program dan rekomendasi perbaikan.

### **BAB III**

#### **METODE PENELITIAN**

Penelitian ini dimaksudkan untuk melihat proses bisnis pada pembelajaran praktikum *online* atau *offline* serta menganalisa bentuk dukungan *cloud computing* pada pelaksanaan praktikum Program Studi Matematika FST UT. Dalam hal ini, pengembangan desain praktikum dilakukan berdasarkan metode ADDIE (*Analysis, Design, Development, Implementation, Evaluation*).

## BAB IV

### HASIL DAN PEMBAHASAN

#### 3.1 Analisis

Pada tahapan analisis, peneliti mengidentifikasi kebutuhan pembelajaran dan luaran yang dituju. Dalam hal ini, praktikum komputasi Prodi Matematika bertujuan untuk menekankan pentingnya pengalaman menggunakan *software* dan bahasa pemrograman untuk menyelesaikan masalah matematika. Adapun tahapan penyelesaian masalah dimulai dari kemampuan menginterpretasikan suatu masalah lalu memvisualisasikannya untuk menentukan solusi sehingga mahasiswa dapat membuat kesimpulan dari permasalahan tersebut.

Sebagai perguruan tinggi jarak jauh, layanan pembelajaran yang sering digunakan Prodi Matematika adalah LMS yang menggunakan aplikasi Moodle. Layanan yang disebut tutorial *online* ini dapat dimaksimalkan penggunaannya dengan memanfaatkan fasilitas dan sumber daya yang ada untuk pelaksanaan praktikum komputasi. Untuk itu, pada tahapan ini dilakukan identifikasi kebutuhan pembelajaran sebagai berikut:

(1) menganalisis kebutuhan pembelajaran;

Ada dua jenis praktikum komputasi yang dilakukan. Untuk Praktikum Komputasi dan Pemrograman (PKP), diperlukan *tools* laboratorium komputasi *online* yang dapat dilakukan mahasiswa di LMS. Untuk Praktikum Visualisasi dan Optimasi (PVO), perlu skema monitoring praktikum yang dilakukan mahasiswa secara mandiri menggunakan *software open source*.

(2) mengidentifikasi tujuan pembelajaran

Sesuai dengan praktikum komputasi yang dipilih, tujuan pelaksanaan PKP dan PVO adalah mendampingi dan membimbing mahasiswa agar dapat memenuhi kompetensi mata kuliah yang ditentukan. Dalam hal ini, capaian pembelajaran lulusan yang dibebankan pada kedua mata kuliah tersebut adalah mahasiswa mampu (1) menguasai prinsip-prinsip matematika dasar, pemrograman linear, dan metode numerik, (2) mengembangkan pemikiran matematis, yang diawali dengan penguasaan prosedural/komputasi hingga eksplorasi dan penalaran logis, (3) mengidentifikasi dan memecahkan masalah melalui pendekatan matematis dengan melibatkan bantuan teknologi, (4) merekonstruksi, berpikir secara terstruktur, mengkaji keakuratan dan

menginterpretasikan hasil, dan (5) mengambil keputusan yang tepat dengan menggunakan berbagai alternatif pemecahan masalah matematis yang telah tersedia.

(3) ketersediaan sumber daya

Terkait dengan sumber daya, beberapa komponen yang dibutuhkan dan perlu dipastikan ketersediaannya adalah:

1. Mahasiswa harus menyediakan perangkat komputer dan jaringan internet untuk pelaksanaan praktikum;
2. Program Studi harus menyediakan tutor atau laboran IT untuk mendampingi mahasiswa praktikum;
3. UT harus menyediakan LMS yang memberikan berbagai tools yang mendukung pelaksanaan praktikum *online*.

### 3.2 Desain

Pada tahap pengembangan desain, peneliti mengidentifikasi kebutuhan pengembangan desain praktikum sebagai berikut:

1. rancangan mata kuliah PKP dan PVO;
2. buku panduan PKP dan PVO;
3. buku pedoman PKP dan PVO;
4. kelas master praktikum *online*;
5. skema penilaian;
6. format pelaporan dan penilaian;
7. rancangan pelatihan pelaksanaan praktikum *online* bagi tutor dan laboran;
8. skema praktikum offline (jika diperlukan) bagi mahasiswa yang lokasinya tidak terjangkau oleh akses internet;
9. standar operasional prosedur pelaksanaan PKP dan PVO bagi program studi, UT daerah, dan unit operasional terkait.

Dari daftar kebutuhan tersebut, pada penelitian tahun ini ditargetkan dapat diselesaikan poin nomor 1 sampai dengan 6. Adapun poin selanjutnya, akan dilaksanakan pada penelitian selanjutnya.

### 3.3 Pengembangan

Pada tahap pengembangan desain praktikum, dilakukan beberapa kegiatan yaitu:

1. Menyusun buku Panduan Praktikum. Penyusunan materi panduan praktikum dilakukan oleh tim penulis yang merupakan dosen Prodi Matematika dan Prodi Sistem Informasi FST.
2. Melakukan ujicoba langkah-langkah pemrograman dan kode program untuk PKP dan PVO. Pelaksanaan ujicoba ini dilakukan bersama dengan mahasiswa dengan cara melibatkan beberapa mahasiswa dalam proses penulisan materi panduan praktikum.
3. Melakukan verifikasi proses PKP dan PVO.  
Kegiatan ini dilakukan dalam bentuk *Focus Discussion Group* (FDG) secara *online* dan *offline*. Didalamnya dilakukan kegiatan ujicoba dan verifikasi materi panduan praktikum.
4. Melakukan *benchmarking* praktikum ke Prodi Matematika UI dan ITS.  
Kegiatan *benchmarking* ini bertujuan untuk melihat model tugas dan mekanisme praktikum sebagai acuan capaian pembelajaran praktikum. Selain itu, kegiatan ini juga bertujuan untuk melakukan peninjauan kerjasama praktikum dengan institusi tersebut. UI dipilih karena jumlah mahasiswa Prodi Matematika yang terdaftar di UT Jakarta terbanyak dibandingkan UT daerah lain, selain itu karena keilmuan Matematika UI sesuai dengan kebutuhan praktikum komputasi dan pemrograman. Adapun ITS dipilih karena keilmuan Matematika ITS sesuai dengan kebutuhan praktikum visualisasi dan optimasi. Jumlah mahasiswa Prodi Matematika di UT Surabaya juga cukup banyak dan berada di urutan keempat dari seluruh UT daerah.
5. Melakukan evaluasi materi dan desain praktikum berdasarkan hasil *benchmarking*.  
Dalam proses penulisan panduan praktikum, rancangan awal praktikum mengalami perubahan berdasarkan hasil *benchmarking*. Hal ini dilakukan agar kompetensi yang dicapai dalam proses praktikum nantinya dapat setara dengan praktikum di institusi lain.

Proses implementasi dan evaluasi belum dilakukan oleh peneliti karena proses pengembangan desain praktikum belum selesai. Beberapa kebutuhan pengembangan yang belum dilaksanakan adalah:

1. penyusunan pedoman praktikum;
2. pembuatan kelas master praktikum *online*;
3. rancangan pelatihan pelaksanaan praktikum *online* bagi tutor dan laboran;
4. skema praktikum *offline* (jika diperlukan) bagi mahasiswa yang lokasinya tidak terjangkau oleh akses internet;
5. standar operasional prosedur pelaksanaan PKP dan PVO bagi program studi, UT daerah, dan unit operasional terkait.

Oleh karena itu peneliti akan melanjutkan pengembangan desain praktikum komputasi ini tahun 2025.



## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **1. Kesimpulan**

Penelitian ini masih dalam proses dan belum selesai. Hasil-hasil yang diperoleh masih sementara dan belum diujicoba dalam skala yang lebih besar. Oleh karena itu, beberapa kegiatan penelitian yang akan dilakukan selanjutnya adalah:

1. pembuatan kelas master praktikum *online*;
2. menguji skema pelaksanaan praktikum *online*;
3. melakukan ujicoba pelaksanaan praktikum *online* melalui LMS bersama mahasiswa, terutama mahasiswa yang berasal dari daerah 3T;
4. membuat skema praktikum *offline* (jika diperlukan) bagi mahasiswa yang lokasinya tidak terjangkau oleh akses internet;
5. melakukan evaluasi materi dan desain praktikum bersama pakar;
6. merevisi materi panduan;
7. menyusun standar operasional prosedur pelaksanaan PKP dan PVO bagi program studi, UT daerah, dan unit operasional terkait.
8. menyusun pedoman praktikum;
9. melakukan sosialisasi ke mahasiswa, UT daerah, dan unit operasional terkait
10. merancang pelatihan pelaksanaan praktikum *online* bagi tutor dan laboran;

#### **2. Saran**

Pengembangan desain praktikum ini melibatkan berbagai pihak baik internal maupun eksternal. Oleh karena itu, diperlukan dukungan dan penguatan baik dari segi teknologi, jaringan, sumber daya manusia, kerjasama mitra, dan kebijakan tentang ketentuan laboran praktikum untuk memudahkan pelaksanaan praktikum *online*.

## DAFTAR PUSTAKA

- [1] Surat Keputusan Presiden Himpunan Matematika Indonesia (*The Indonesian Mathematical Society / IndoMS*) Nomor: 028/Pres/IndoMS/SK/I/2022 Tanggal 3 Januari 2022 Tentang Penetapan Rekomendasi IndoMS Untuk Kurikulum Minimal Program Studi Sarjana Matematika di Indonesia
- [2] Ranius, A.Y. . (2021) *Pengembangan Model Pembelajaran Praktikum Algoritma dan Pemrograman Berbasis Laboratorium Virtual*. Doktor Thesis, Universitas Negeri Jakarta.
- [3] Yani, H. (2012). *Pengembangan E-Learning Dengan Metode Learning Technology System Architecture (LTSA)*. STIKOM Dinamika Bangsa Jambi : Thesis.
- [4] Elkom (2012). *Cloud Computing Aplikasi Berbasis Web*. Edisi I. Yogyakarta : Penerbit Andi
- [5] Saksono, M.R. (2014). *Pengantar Cloud Computing*. 4-15
- [6] Hashemi, S., Monfareidi, K., dan Hashemi, S. Y. (2015). Cloud Computing for Secure Services in EGovernment Architecture. *J. Inf. Technol. Res.*, 8(1), 43–61. <http://doi.org/10.4018/JITR.2015010104>

Kover dalam

Halaman  
Prancis

## » Kata Pengantar

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa yang telah memberikan wawasan dan ilmu pengetahuan serta pemahaman untuk mempelajarinya.

Mulai tahun akademik 2025/2026, Program Studi S1 Matematika FST UT menawarkan mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman sebanyak 2 (dua) sks. Untuk menempuh mata kuliah tersebut, mahasiswa wajib mempunyai Panduan Praktikum Komputasi dan Pemrograman ini. Karena dalam panduan, mahasiswa akan mendapatkan penjelasan tentang konsep, prosedur, dan petunjuk pelaksanaan praktikum mulai persiapan, pelaksanaan, dan asesmen praktikum.

Panduan Praktikum Komputasi dan Pemrograman ini dapat terwujud atas kerjasama tim penulis dengan beberapa mahasiswa. Kami mengucapkan terima kasih dan penghargaan setinggi-tingginya atas waktu, ilmu, dan kerja cerdasnya sehingga panduan ini dapat selesai tepat pada waktunya. Kami menyadari bahwa panduan ini masih banyak kekurangan. Untuk itu, saran dan koreksi dari para pengguna sangat berguna untuk menyempurnakan materi dalam panduan ini. Apabila ada hal-hal yang akan disampaikan, silakan mengirimkan email ke [prodimatematikafst@ecampus.ut.ac.id](mailto:prodimatematikafst@ecampus.ut.ac.id).

Tangerang Selatan, Desember 2024  
Salam,

**Tim Penulis:**

1. Kani
2. Sitta Alief Farihati
3. Hasoloan Siregar

**Mahasiswa:**

Daru Setiawan  
Jihan Kristal Yasmin  
Mikail Nurwahid  
Siti Latifatun Nisa  
Yitro Vanem

# » Daftar Isi

<b>Kata Pengantar</b>	iii
<b>Daftar isi</b>	iv
<b>BAB I Pendahuluan</b>	
Latar Belakang	7
Tujuan	7
Ruang Lingkup	7
Struktur Panduan	8
Persyaratan Peserta	8
<b>BAB II Prosedur Pelaksanaan Praktikum Komputasi Menggunakan Java</b>	
Tahap Persiapan	10
Tahap Pelaksanaan	11
<b>BAB III Prosedur Pelaksanaan Praktikum Pemrograman Numerik Menggunakan Python</b>	16
Tahap Persiapan	16
Tahap Pelaksanaan	17
<b>BAB IV Pelaporan dan Penilaian</b>	30
<b>Lampiran</b>	
<b>Daftar Pustaka</b>	40



# BAB I

## Pendahuluan



# » Pendahuluan

## I.A. LATAR BELAKANG

Pada era digital ini, kebutuhan akan lulusan yang terampil dalam menggunakan teknologi informasi dan komunikasi semakin besar. Oleh karena itu, Program Studi S1 Matematika FST UT melakukan perubahan kurikulum menjadi berorientasi luaran atau *Outcome Based Education*. Salah satu bentuk perubahan tersebut adalah dengan memberikan mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman.

Mata kuliah Praktikum Komputasi dan Pemrograman mulai ditawarkan kepada mahasiswa pada 2025/2026 Ganjil. Mata kuliah ini merupakan mata kuliah yang wajib ditempuh mahasiswa. Selama mempelajari mata kuliah ini, mahasiswa akan mengikuti proses pembelajaran yang memberikan pengalaman dan keterampilan komputasi dan pemrograman. Oleh karena itu, mahasiswa harus menggunakan komputer dan aplikasi pemrograman secara langsung. Agar proses tersebut berjalan dengan baik, pendampingan praktikum akan dilaksanakan melalui tutorial online sehingga mahasiswa wajib mengikuti tutorial online.

## I.B. TUJUAN

Setelah mempelajari mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman, mahasiswa diharapkan mampu menggunakan *software* atau bahasa pemrograman untuk menentukan solusi suatu masalah matematika. Adapun capaian pembelajaran lulusan yang dibebankan pada mata kuliah ini adalah mahasiswa mampu (1) menguasai prinsip-prinsip matematika dasar dan metode numerik, (2) mengembangkan pemikiran matematis, yang diawali dengan penguasaan prosedural/komputasi hingga eksplorasi dan penalaran logis, (3) mengidentifikasi dan memecahkan masalah melalui pendekatan matematis dengan melibatkan bantuan teknologi, (4) merekonstruksi, berpikir secara terstruktur, mengkaji keakuratan dan menginterpretasikan hasil, dan (5) mengambil keputusan yang tepat dengan menggunakan berbagai alternatif pemecahan masalah matematis yang telah tersedia.

## I.C. RUANG LINGKUP

Panduan Praktikum Komputasi dan Pemrograman ini menjelaskan tentang tahapan pelaksanaan praktikum yang terdiri atas persiapan, pelaksanaan, dan evaluasi.

1. Pada tahapan persiapan dijelaskan tentang proses instalasi aplikasi pemrograman dan materi pendukung yang diperlukan.
2. Pada tahapan pelaksanaan diberikan materi dan contoh praktikum yang dapat dipraktikkan secara mandiri. Selain itu, diberikan pula soal-soal latihan untuk dikerjakan secara mandiri. Adapun verifikasi program dan hasil output yang diperoleh akan dibahas dalam praktikum online.
3. Pada tahapan praktikum online, akan dijelaskan aktivitas pembelajaran, pelaporan, dan penilaian praktikum online. Selain itu, akan disampaikan pula format pelaporan dan skema penilaian mata kuliah ini. Sebagai bukti proses praktikum, Anda akan diminta membuat laporan akhir praktikum dengan menggunakan format yang terdapat dalam lampiran.

#### I.D. STRUKTUR PANDUAN

Secara garis besar, Panduan Praktikum Komputasi dan Pemrograman ini terdiri atas tiga topik, yaitu:

1. Topik 1 membahas tentang Prosedur Pelaksanaan Praktikum Komputasi Menggunakan Java.
2. Topik 2 membahas tentang Prosedur Pelaksanaan Praktikum Pemrograman Numerik Menggunakan Python.
3. Topik 3 membahas tentang Praktikum Online.

Dalam panduan praktikum ini, materi yang digunakan merujuk pada BMP MSIM4203 Algoritma dan Pemrograman Edisi 2, BMP MATA4110 Kalkulus I Edisi 3, BMP MATA4112 Aljabar Linear Elementer I Edisi 3, dan MATA213 Metode Numerik Edisi 3. Mahasiswa dapat menggunakan bahan ajar yang sesuai untuk mendukung proses pembelajaran dalam praktikum ini.

#### I.E. PERSYARATAN PESERTA

Mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman merupakan mata kuliah wajib bagi mahasiswa Program Studi S1 Matematika FST UT. Dengan demikian, peserta mata kuliah ini adalah mahasiswa yang telah melakukan registrasi mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman. Agar dapat mengikuti mata kuliah ini dengan baik, mahasiswa diharapkan telah atau sedang menempuh matakuliah Algoritma dan Pemrograman, atau Komputer I. Sebagai prasyarat mempelajari mata kuliah ini, mahasiswa sebaiknya telah menempuh mata kuliah Kalkulus I/Kalkulus Diferensial, Aljabar Linear Elementer I/Aljabar Linear Elementer, Persamaan Diferensial Biasa, dan Metode Numerik.

# BAB II

## Prosedur Pelaksanaan Praktikum Komputasi Menggunakan Java

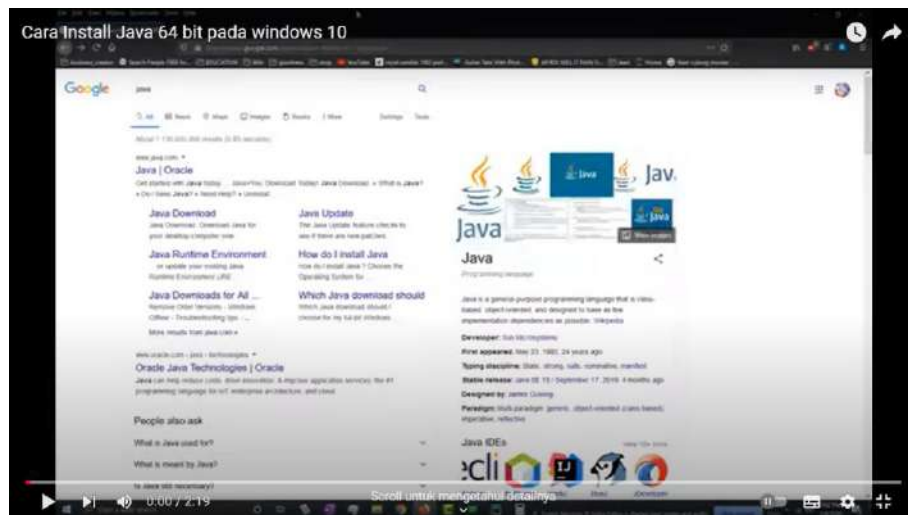
Praktikum dilakukan dalam tiga tahap yaitu persiapan, pelaksanaan, dan pelaporan dan penilaian.

## II.A. TAHAP PERSIAPAN

Untuk memulai praktikum komputasi, mahasiswa wajib melakukan hal-hal berikut:

- Menyediakan komputer (PC atau laptop) yang terkoneksi internet,
- Mengunduh installer Java,
- Melakukan instalasi sesuai dengan materi instalasi Java dalam BMP MSIM4203 Algoritma dan Pemrograman.

Anda dapat melihat video cara menginstal Java pada link berikut ini:



<https://www.youtube.com/watch?v=ReQ-Hze50zM>

Anda juga dapat mengerjakan pemrograman Java via website atau aplikasi smartphone melalui *compiler* berikut ini:

- <https://onecompiler.com/>
- <https://www.jdoodle.com/online-java-compiler>
- <https://replit.com/languages/java>
- [https://www.onlinegdb.com/online java compiler](https://www.onlinegdb.com/online_java_compiler)
- <https://codesandbox.io/s/new>
- atau online compiler lainnya

- Mahasiswa juga dapat mempelajari secara mandiri penggunaan perintah Java seperti yang dijelaskan dalam BMP Algoritma dan Pemrograman.

## II.B. TAHAP PELAKSANAAN

Pada tahap ini, kita akan mempelajari terlebih dahulu materi praktikum dan contoh output yang dihasilkan oleh Java. Anda diharapkan mempraktikkan materi dan contoh yang diberikan sehingga menghasilkan output seperti yang ditampilkan. Selanjutnya, Anda dapat mengerjakan latihan yang diberikan dalam panduan ini secara mandiri. Pendampingan pelaksanaan praktikum akan dilakukan dalam tutorial online. Mahasiswa dapat memanfaatkan menu VPL (*Virtual Programming Library*) yang disediakan dalam kelas tutorial online.

### II.B.1. Materi Kalkulus

Dalam mata kuliah Kalkulus Diferensial, kita telah mempelajari materi fungsi pada bilangan real. Pada kegiatan praktikum ini, kita akan melakukan komputasi fungsi-fungsi dalam kalkulus.

#### Praktikum II.B.1.1. Fungsi Logaritma

Pada Modul 3 BMP MATA4110 Kalkulus I, kita telah belajar tentang fungsi logaritma. Anda dapat membaca kembali materi tersebut. Dalam praktikum kali ini, kita akan mengerjakan contoh soal berikut ini. Contoh ini merupakan soal Contoh 3.2.1 dalam BMP.

Tentukan nilai:

- a.  $\ln 2$
- b.  $\ln 1001$
- c.  $\ln \frac{1}{2}$
- d.  $\ln \frac{4}{100}$

Dari soal yang diberikan tersebut, perhatikan program Java yang digunakan dan output yang diperoleh berikut ini.

- a. **Program** menentukan nilai  $\ln 2$  dengan program Java

```
import java.lang.Math;

public class LogarithmLn2 {
    public static void main(String[] args) {
        double ln2 = Math.log(2);
        System.out.println("ln(2) = " + ln2);
    }
}
```

**Output:**

```
ln(2) = 0.6931471805599453
```

b. **Program** menentukan nilai  $\ln 1001$  dengan program Java

```
import java.lang.Math;

public class LogarithmLn1001 {
    public static void main(String[] args) {
        double ln1001 = Math.log(1001);
        System.out.println("ln(1001) = " + ln1001);
    }
}
```

**Output:**

```
ln(1001) = 6.90875477931522
```

c. **Program** mencari  $\ln \frac{1}{2}$  dengan program Java

```
public class LogarithmLnSeperdua {
    public static void main(String[] args) {
        double lnHalf = Math.log(1.0 / 2);
        System.out.println("ln(1/2) = " + lnHalf);
    }
}
```

**Output:**

```
ln(1/2) = -0.6931471805599453
```

d. **Program** menentukan nilai  $\ln \frac{4}{100}$  dengan program Java

```
import java.lang.Math;

public class LogarithmLn_4per100 {
    public static void main(String[] args) {
        double lnFourOverHundred = Math.log(4.0 / 100);
    }
}
```

```
System.out.println("ln(4/100) = " +  
lnFourOverHundred);  
}  
}
```

Output:

```
ln(4/100) = -3.2188758248682006
```

Sebagai latihan, silakan kerjakan soal berikut.

1. Tentukan nilai  $\ln 5$ ,  $\ln 0$ , dan  $\ln(-1)$ .
2. Apakah nilai  $\ln(5 + 2)$  sama dengan  $\ln 5 + \ln 2$ ? Tunjukkan dalam bentuk program Java dan outputnya.

### Praktikum II.B.1.2. Fungsi Eksponensial

Materi fungsi eksponensial dijelaskan pada Modul 4 BMP MATA4110 Kalkulus I. Anda dapat membaca kembali materi tersebut. Berikut ini contoh 3.2.3 dalam BMP. Kita akan mengerjakan contoh soal tersebut.

Tentukan nilai:

- a.  $e^2$
- b.  $e^{\frac{1}{2}}$
- c.  $e^{-3}$
- d.  $e^{-\frac{1}{5}}$

Dari soal tentang fungsi eksponensial yang diberikan, perhatikan program Java yang digunakan dan output yang diperoleh berikut ini.

- a. **Program** menentukan nilai  $e^2$  menggunakan Program Java

```
import java.lang.Math;  
  
public class Exponential_EDua {  
    public static void main(String[] args) {  
        double e2 = Math.exp(2);  
        System.out.println("e^2 = " + e2);  
    }  
}
```

Output:

```
e^2 = 7.38905609893065
```

b. **Program** menentukan nilai  $e^{\frac{1}{2}}$  menggunakan Program Java

```
import java.lang.Math;

public class Exponential_ESeperDua {
    public static void main(String[] args) {
        double e_seperdua = Math.exp(0.5); // 1/2 = 0.5
        System.out.println("e^(1/2) = " + e_seperdua);
    }
}
```

**Output:**

```
e^(1/2) = 1.6487212707001282
```

c. **Program** menentukan nilai  $e^{-3}$  menggunakan Program Java

```
import java.lang.Math;

public class Exponential_ENegatifTiga {
    public static void main(String[] args) {
        double eNegatif3 = Math.exp(-3);
        System.out.println("e^(-3) = " + eNegatif3);
    }
}
```

**Output:**

```
e^(-3) = 0.049787068367863944
```

d. **Program** menentukan nilai  $e^{-\frac{1}{5}}$  menggunakan Program Java

```
import java.lang.Math;

public class Eksponensial_ENegatifSeperLima {
    public static void main(String[] args) {
        double eNegatif_SeperLima = Math.exp(-1.0 / 5); // -1/5
    }
}
```



```
System.out.println("e^(-1/5) = " + eNegatif_SeperLima);
}
}
```

**Output:**

```
e^(-1/5) = 0.8187307530779818
```

Sebagai latihan, silakan kerjakan soal berikut.

1. Tentukan nilai  $e^{-0,6}$ ,  $e^0$ , dan  $e^{0,8}$ .
2. Apakah nilai  $e^{1+2}$  sama dengan  $e^1 + e^2$ ? Tunjukkan dalam bentuk program Java dan outputnya.

**Praktikum II.B.1.3. Fungsi Trigonometri**

Pada Modul 3 BMP MATA4110 Kalkulus I, kita juga telah belajar tentang fungsi trigonometri. Anda dapat membaca kembali materi tersebut. Dalam praktikum kali ini, kita akan mengerjakan Contoh 3.2.5 sampai dengan Contoh 3.2.8 dalam BMP.

Tentukan nilai:

- a.  $\sin\left(\frac{\pi}{6}\right)$ ,  $\cos\left(\frac{\pi}{6}\right)$ ,  $\sin(30^\circ)$ ,  $\cos(30^\circ)$
- b.  $\sin(62.50^\circ)$ ,  $\cos(62.50^\circ)$
- c.  $\tan(3.213)$
- d.  $\sec(49.12)$

**Program:**

- a. Program menentukan nilai  $\sin\left(\frac{\pi}{6}\right)$ ,  $\cos\left(\frac{\pi}{6}\right)$ ,  $\sin(30^\circ)$ , dan  $\cos(30^\circ)$  dengan program Java

```
public class Trigonometri_Contoh325a {
    public static void main(String[] args) {
        // Sudut dalam radian untuk  $\pi/6$ 
        double angleInRadians = Math.PI / 6; //  $\pi/6$ 
        // Sudut dalam derajat untuk  $30^\circ$ 
        double angleInDegrees = 30; //  $30^\circ$ 

        // Menghitung nilai sin dan cos untuk  $\pi/6$ 
    }
}
```

```

double sinValueRadians = Math.sin(angleInRadians);
double cosValueRadians = Math.cos(angleInRadians);

// Menghitung nilai sin dan cos untuk 30°
double sinValueDegrees =
Math.sin(Math.toRadians(angleInDegrees));
double cosValueDegrees =
Math.cos(Math.toRadians(angleInDegrees));

// Menampilkan hasil
System.out.printf("Nilai sin(π/6) = %.4f\n",
sinValueRadians);
System.out.printf("Nilai cos(π/6) = %.4f\n",
cosValueRadians);
System.out.printf("Nilai sin(30°) = %.4f\n",
sinValueDegrees);
System.out.printf("Nilai cos(30°) = %.4f\n",
cosValueDegrees);
}
}

```

**Output:**

```

Nilai sin(π/6) = 0.5000
Nilai cos(π/6) = 0.8660
Nilai sin(30°) = 0.5000
Nilai cos(30°) = 0.8660

```

b. Program menentukan nilai  $\sin(62.50^\circ)$  dan  $\cos(62.50^\circ)$  dengan program Java

```

public class Trigonometri_Contoh326b {
    public static void main(String[] args) {
        // Sudut dalam derajat
        double angleInDegrees = 62.5;

        // Konversi ke radian
        double angleInRadians =
Math.toRadians(angleInDegrees);

        // Menghitung nilai sin dan cos
        double sinValue = Math.sin(angleInRadians);
        double cosValue = Math.cos(angleInRadians);
    }
}

```

```
        // Menampilkan hasil
        System.out.printf("Sudut: %.2f°, sin(%.2f°) = %.4f,
cos(%.2f°) = %.4f\n",
            angleInDegrees, angleInDegrees, sinValue,
            angleInDegrees, cosValue);
    }
}
```

Output:

```
Sudut: 62.50°, sin(62.50°) = 0.8870, cos(62.50°) = 0.4617
```

c.

```
public class Trigonometri_Contoh327a {
    public static void main(String[] args) {
        // Nilai sudut dalam radian
        double angle = 3.213;

        // Menghitung nilai tan(3.213)
        double tanValue = Math.tan(angle);

        // Menampilkan hasil
        System.out.printf("Nilai tan(3.213) = %.4f\n",
tanValue);
    }
}
```

Output:

```
Nilai tan(3.213) = 0.0715
```

d.

```
public class Trigonometri_Contoh327c {
    public static void main(String[] args) {
        // Nilai sudut dalam derajat
```

```

double angleInDegrees = 49.12;

double secValue = 1 / Math.cos(49.12); // sec(t) =
1/cos(t)

// Menampilkan hasil
System.out.printf("Nilai sec(49.12) = %.4f\n",
secValue);
}
}

```

**Output:**

```

Nilai sec(49.12) = 2.4236

```

Sebagai latihan, silakan kerjakan soal berikut.

1. Tentukan nilai  $\cos\left(\frac{\pi}{6}\right) + \sin(62,50^\circ)$
2. Apakah nilai  $\tan\left(\frac{\pi}{6}\right) + \tan(30^\circ)$  sama dengan  $\tan\left(\frac{\pi}{6} + 30^\circ\right)$ ?  
Tunjukkan dalam bentuk program Java dan outputnya.

**II.B.2. Materi Matriks**

Kita telah mengenal matriks dan operasinya dalam mata kuliah Aljabar Linear Elementer.

**Praktikum II.B.1.2. Penjumlahan Matriks**

Pada Modul 1 BMP MATA4112 Aljabar Linear Elementer I, kita telah belajar tentang jenis dan ukuran matriks. Anda dapat membaca kembali materi tersebut. Dalam praktikum kali ini, kita akan mengerjakan soal tentang penjumlahan matriks.

Tentukan  $A + B$ , untuk :

a.  $A = \begin{bmatrix} 1 & -1 & 2 \\ 0 & 3 & 1 \end{bmatrix}$  dan  $B = \begin{bmatrix} 2 & 1 & 0 \\ 1 & -3 & 1 \end{bmatrix}$ .

b.  $A = \begin{bmatrix} 1 & 5 \\ -4 & 3 \\ 2 & 6 \end{bmatrix}$  dan  $B = \begin{bmatrix} -1 & -5 \\ 4 & -3 \\ -2 & -6 \end{bmatrix}$

$$c. A = \begin{bmatrix} 3 & 1 & -1 \\ 4 & 2 & -1 \end{bmatrix} \text{ dan } B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

### a. Program

```
public class MatriksAaddB {
    public static void main(String[] args) {
        // Matriks A
        int[][] A = {
            {1, -1, 2},
            {0, 3, 1}
        };

        // Matriks B
        int[][] B = {
            {2, 1, 0},
            {1, -3, 1}
        };

        // Matriks hasil C
        int[][] C = new int[A.length][A[0].length];

        // Menghitung A + B
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[i].length; j++) {
                C[i][j] = A[i][j] + B[i][j];
            }
        }

        // Menampilkan hasil
        System.out.println("Hasil A + B untuk pasangan a:");
        for (int[] row : C) {
            for (int val : row) {
                System.out.print(val + " ");
            }
            System.out.println();
        }
    }
}
```

**Output:**

```

Hasil A + B untuk pasangan a:
3 0 2
1 0 2

```

**b. Program:**

```

public class MatriksAaddBb {
    public static void main(String[] args) {
        // Matriks A
        int[][] A = {
            {1, 5},
            {-4, 3},
            {2, 6}
        };

        // Matriks B
        int[][] B = {
            {-1, -5},
            {4, -3},
            {-2, -6}
        };

        // Matriks hasil penjumlahan A + B
        int[][] result = new int[3][2];

        // Penjumlahan elemen-elemen matriks
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[0].length; j++) {
                result[i][j] = A[i][j] + B[i][j];
            }
        }

        // Menampilkan hasil
        System.out.println("Hasil A + B untuk bagian
b:");
        for (int i = 0; i < result.length; i++) {
            for (int j = 0; j < result[0].length; j++) {
                System.out.print(result[i][j] + " ");
            }
            System.out.println();
        }
    }
}

```

**Output:**

```
Hasil A + B untuk bagian b:  
0 0  
0 0  
0 0
```

**c. Program**

```
public class MatrikAaddBc {  
    public static void main(String[] args) {  
        // Matriks A  
        int[][] A = {  
            {3, 1, -1},  
            {4, 2, -1}  
        };  
  
        // Matriks B  
        int[][] B = {  
            {1, 0, 1},  
            {0, 1, 0}  
        };  
  
        // Matriks hasil penjumlahan A + B  
        int[][] result = new int[2][3];  
  
        // Penjumlahan elemen-elemen matriks  
        for (int i = 0; i < A.length; i++) {  
            for (int j = 0; j < A[0].length; j++) {  
                result[i][j] = A[i][j] + B[i][j];  
            }  
        }  
  
        // Menampilkan hasil  
        System.out.println("Hasil A + B untuk bagian c:");  
        for (int i = 0; i < result.length; i++) {  
            for (int j = 0; j < result[0].length; j++) {  
                System.out.print(result[i][j] + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

**Output:**

```

Hasil A + B untuk bagian c:
4 1 0
4 3 -1

```

Sebagai latihan, silakan kerjakan soal berikut.

Diberikan  $A = \begin{bmatrix} 0 & 1 \\ 3 & 0 \\ 2 & -1 \end{bmatrix}$ ,  $B = \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 4 & 2 \end{bmatrix}$ , dan  $C = \begin{bmatrix} 0 & 1 \\ 0 & 3 \\ 0 & -2 \end{bmatrix}$

1. Tentukan hasil  $A + B$  dan  $B + C$ .
2. Apakah nilai  $A + C$  sama dengan  $C + A$ ? Tunjukkan dalam bentuk program Java dan outputnya.

**Praktikum II.B.2.2. Perkalian Matriks**

Pada Modul 1 BMP MATA4112 Aljabar Linear Elementer I, kita telah belajar tentang perkalian matriks. Anda dapat membaca kembali materi tersebut. Dalam praktikum kali ini, kita akan mengerjakan contoh-contoh soal berikut ini.

Tentukan nilai matriks berikut ini:

a.  $A = \begin{bmatrix} 0 & 3 & 2 \\ 2 & 1 & 1 \end{bmatrix}$  dan  $B = \begin{bmatrix} 2 & 0 \\ 3 & 1 \\ 1 & 2 \end{bmatrix}$ . Tentukan  $AB$  dan  $BA$ .

b.  $A = \begin{bmatrix} 1 & -1 & 3 \\ 0 & 1 & -4 \end{bmatrix}$ . Tentukan  $2A$ .

c.  $A = [1 \quad -1 \quad 2]$  dan  $B = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$ . Tentukan  $AB$ .

d.  $A = \begin{bmatrix} 1 \\ -1 \\ 2 \end{bmatrix}$  dan  $B = [2 \quad 2 \quad 1]$ . Tentukan  $AB$ .

**a. Program**

```

public class PerkalianMatriksA {

    public static void main(String[] args) {
        // Matriks A

```



```
int[][] A = {
    {0, 3, 2},
    {2, 1, 1}
};

// Matriks B
int[][] B = {
    {2, 0},
    {3, 1},
    {1, 2}
};

// Perkalian AB
int[][] AB = multiplyMatrices(A, B);
System.out.println("Hasil perkalian AB:");
printMatrix(AB);

// Matriks B untuk perkalian BA (transposed)
int[][] A_T = {
    {0, 3, 2},
    {2, 1, 1}
};

// Perkalian BA
int[][] BA = multiplyMatrices(B, A_T);
System.out.println("Hasil perkalian BA:");
printMatrix(BA);
}

// Fungsi untuk melakukan perkalian dua matriks
public static int[][] multiplyMatrices(int[][]
firstMatrix, int[][] secondMatrix) {
    int rowsA = firstMatrix.length;
    int columnsA = firstMatrix[0].length;
    int columnsB = secondMatrix[0].length;
    int[][] result = new int[rowsA][columnsB];

    for (int i = 0; i < rowsA; i++) {
        for (int j = 0; j < columnsB; j++) {
            for (int k = 0; k < columnsA; k++) {
                result[i][j] += firstMatrix[i][k] *
secondMatrix[k][j];
            }
        }
    }
    return result;
}
```

```
// Fungsi untuk mencetak matriks
public static void printMatrix(int[][] matrix) {
    for (int[] row : matrix) {
        for (int column : row) {
            System.out.print(column + " ");
        }
        System.out.println();
    }
}
}
```

**Output:**

```
Hasil perkalian AB:
11 7
8 3
Hasil perkalian BA:
0 6 4
2 10 7
4 5 4
```

**b. Program**

```
public class PerkalianMatriksB {

    public static void main(String[] args) {
        // Matriks A yang diberikan
        int[][] A = {
            {1, -1, 3},
            {0, 1, -4}
        };

        // Skalar yang ingin dikalikan dengan matriks A
        int skalar = 2;

        // Hasil perkalian skalar
        int[][] hasil = kaliSkalar(A, skalar);

        // Tampilkan hasil
        System.out.println("Hasil perkalian 2A:");
        tampilkanMatriks(hasil);
    }

    // Metode untuk melakukan perkalian skalar pada matriks
```

```
public static int[][] kaliSkalar(int[][] matriks, int
skalar) {
    int baris = matriks.length;
    int kolom = matriks[0].length;
    int[][] hasil = new int[baris][kolom];

    for (int i = 0; i < baris; i++) {
        for (int j = 0; j < kolom; j++) {
            hasil[i][j] = matriks[i][j] * skalar;
        }
    }

    return hasil;
}

// Metode untuk menampilkan matriks
public static void tampilkanMatriks(int[][] matriks) {
    for (int i = 0; i < matriks.length; i++) {
        for (int j = 0; j < matriks[i].length; j++) {
            System.out.print(matriks[i][j] + "\t");
        }
        System.out.println();
    }
}
}
```

### Output:

```
Hasil perkalian 2A:
2      -2      6
0      2      -8
```

### c. Program

```
public class PerkalianMatriksC {
    public static void main(String[] args) {
        // Matriks A (1x3)
        int[] A = {1, -1, 2};

        // Matriks B (3x1)
        int[] B = {3, 2, 1};

        // Hasil perkalian AB (1x1)
        int hasil = 0;
    }
}
```

```

// Perkalian elemen-elemen matriks
for (int i = 0; i < A.length; i++) {
    hasil += A[i] * B[i];
}

// Tampilkan hasil
System.out.println("Hasil perkalian AB: " + hasil);
}
}

```

**Output:**

```

Hasil perkalian AB: 3

```

**d. Program**

```

public class PerkalianMatriksD {
    public static void main(String[] args) {
        // Matriks A (3x1)
        int[] A = {1, -1, 2};

        // Matriks B (1x3)
        int[] B = {2, 2, 1};

        // Matriks hasil AB (3x3)
        int[][] hasil = new int[3][3];

        // Perkalian matriks A dan B
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < B.length; j++) {
                hasil[i][j] = A[i] * B[j];
            }
        }

        // Tampilkan hasil
        System.out.println("Hasil perkalian AB:");
        for (int i = 0; i < hasil.length; i++) {
            for (int j = 0; j < hasil[i].length; j++) {
                System.out.print(hasil[i][j] + "\t");
            }
            System.out.println();
        }
    }
}

```

**Output:**

```
Hasil perkalian AB:
2      2      1
-2     -2     -1
4      4      2
```

Sebagai latihan, silakan kerjakan soal berikut.

Diberikan  $A = \begin{bmatrix} 1 & 0 \\ 3 & -1 \\ 2 & 0 \end{bmatrix}$ ,  $B = \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 4 & 2 \end{bmatrix}$ , dan  $C = \begin{bmatrix} 1 & 3 & 5 \\ 8 & -2 & 1 \end{bmatrix}$

1. Tentukan hasil  $AC$  dan  $BC$
2. Apakah nilai  $BC$  sama dengan  $CB$ ? Tunjukkan dalam bentuk program Java dan outputnya.

**Praktikum II.B.2.3. Transpos Matriks**

Pada Modul 1 BMP MATA4112 Aljabar Linear Elementer I, kita telah belajar tentang transpos matriks. Anda dapat membaca kembali materi tersebut. Dalam praktikum kali ini, kita akan mengerjakan contoh-contoh soal berikut ini.

Tentukan transpos matriks berikut:

a.  $A = \begin{bmatrix} 2 & 0 & -1 \\ 1 & -1 & 3 \end{bmatrix}$

b.  $B = \begin{bmatrix} 1 & -1 \\ 0 & 2 \\ 8 & 3 \end{bmatrix}$

c.  $C = \begin{bmatrix} 3 & 2 & 1 \\ 1 & 0 & -2 \\ 1 & 4 & 2 \end{bmatrix}$

d.  $D = \begin{bmatrix} 1 & 1 & 2 & 0 \\ 1 & 2 & 4 & 5 \\ 2 & 4 & 0 & 2 \\ 0 & 5 & 3 & 1 \end{bmatrix}$

**a. Program:**

```
public class TransposeA {
    public static void main(String[] args) {
        // Matriks A
```

```

int[][] A = {
    {2, 0, -1},
    {1, -1, 3}
};

// Matriks transpos A^T
int[][] transposeA = new int[3][2];

// Menghitung transpos
for (int i = 0; i < A.length; i++) {
    for (int j = 0; j < A[0].length; j++) {
        transposeA[j][i] = A[i][j];
    }
}

// Menampilkan hasil transpos
System.out.println("Transpose Matriks A:");
for (int i = 0; i < transposeA.length; i++) {
    for (int j = 0; j < transposeA[0].length; j++) {
        System.out.print(transposeA[i][j] + " ");
    }
    System.out.println();
}
}
}

```

**Output:**

```

Transpose Matriks A:
2 1
0 -1
-1 3

```

**b. Program:**

```

public class TransposeB {
    public static void main(String[] args) {
        // Matriks B
        int[][] B = {
            {1, -1},
            {0, 2},
            {8, 3}
        };

        // Matriks transpos B^T
    }
}

```

```
int[][] transposeB = new int[2][3];

// Menghitung transpos
for (int i = 0; i < B.length; i++) {
    for (int j = 0; j < B[0].length; j++) {
        transposeB[j][i] = B[i][j];
    }
}

// Menampilkan hasil transpos
System.out.println("Transpose Matriks B:");
for (int i = 0; i < transposeB.length; i++) {
    for (int j = 0; j < transposeB[0].length; j++) {
        System.out.print(transposeB[i][j] + " ");
    }
    System.out.println();
}
}
```

**Output:**

```
Transpose Matriks B:
1 0 8
-1 2 3
```

**c. Program:**

```
public class TransposeC {
    public static void main(String[] args) {
        // Matriks C
        int[][] C = {
            {3, 2, 1},
            {1, 0, -2},
            {1, 4, 2}
        };

        // Matriks transpos C^T
        int[][] transposeC = new int[3][3];

        // Menghitung transpos
        for (int i = 0; i < C.length; i++) {
            for (int j = 0; j < C[0].length; j++) {
                transposeC[j][i] = C[i][j];
            }
        }
    }
}
```

```

    }

    // Menampilkan hasil transpos
    System.out.println("Transpose Matriks C:");
    for (int i = 0; i < transposeC.length; i++) {
        for (int j = 0; j < transposeC[0].length; j++) {
            System.out.print(transposeC[i][j] + " ");
        }
        System.out.println();
    }
}
}

```

**Output:**

```

Transpose Matriks C:
3 1 1
2 0 4
1 -2 2

```

**d. Program:**

```

public class TransposeD {
    public static void main(String[] args) {
        // Matriks D
        int[][] D = {
            {1, 1, 2, 0},
            {1, 2, 4, 5},
            {2, 4, 0, 2},
            {0, 5, 3, 1}
        };

        // Matriks transpos D^T
        int[][] transposeD = new int[4][4];

        // Menghitung transpos
        for (int i = 0; i < D.length; i++) {
            for (int j = 0; j < D[0].length; j++) {
                transposeD[j][i] = D[i][j];
            }
        }

        // Menampilkan hasil transpos
        System.out.println("Transpose Matriks D:");
        for (int i = 0; i < transposeD.length; i++) {

```



```
        for (int j = 0; j < transposeD[0].length; j++) {
            System.out.print(transposeD[i][j] + " ");
        }
        System.out.println();
    }
}
```

**Output:**

```
Transpose Matriks D:
1 1 2 0
1 2 4 5
2 4 0 3
0 5 2 1
```

Sebagai latihan, silakan kerjakan soal berikut.

Diberikan  $A = \begin{bmatrix} 1 & -2 \\ 0 & 1 \end{bmatrix}$ ,  $B = \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 4 & 2 \end{bmatrix}$ , dan  $C = \begin{bmatrix} 1 & 3 & 5 \\ 8 & -2 & 1 \end{bmatrix}$

1. Tentukan transpos  $A$ ,  $B$ , dan  $C$ .
2. Apakah nilai  $B^t + C$  sama dengan  $B + C^t$ ? Tunjukkan dalam bentuk program Java dan outputnya.

**Praktikum II.B.2.4. Invers Matriks**

Pada Modul 1 BMP MATA4112 Aljabar Linear Elementer I, kita telah belajar tentang invers matriks. Anda dapat membaca kembali materi tersebut. Dalam praktikum kali ini, kita akan mengerjakan contoh soal berikut ini. Contoh ini merupakan soal Contoh 1.2.10 dalam BMP.

Diberikan matriks  $A = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}$  dan  $B = \begin{bmatrix} 2 & 1 \\ -1 & -2 \end{bmatrix}$ . Tentukan:

- a.  $A^{-1}$
- b.  $B^{-1}$
- c.  $B^{-1}A^{-1}$
- d.  $(AB)^{-1}$

**Program:**

```
public class MatriksInvers {
    public static void main(String[] args) {
        // Matriks A
```

```

double[][] A = {
    {1, 2},
    {2, -1}
};

// Matriks B
double[][] B = {
    {2, 1},
    {-1, -2}
};

// Hitung invers A dan B
double[][] A_inv = invertMatrix(A);
double[][] B_inv = invertMatrix(B);

System.out.println("a.");
System.out.println("Invers dari A:");
printMatrix(A_inv);
System.out.println(" ");

System.out.println("b.");
System.out.println("Invers dari B:");
printMatrix(B_inv);
System.out.println(" ");

// Hitung B^-1 A^-1
System.out.println("c.");
double[][] B_inv_A_inv = multiplyMatrices(B_inv,
A_inv);
System.out.println("B^-1 A^-1:");
printMatrix(B_inv_A_inv);
System.out.println(" ");

// Hitung (AB)^-1 = B^-1 A^-1
System.out.println("d.");
double[][] AB_inv = B_inv_A_inv; // Sama dengan B^-1
A^-1
System.out.println("(AB)^-1:");
printMatrix(AB_inv);
}

// Fungsi untuk menginvers matriks 2x2
public static double[][] invertMatrix(double[][] matrix) {
    double det = (matrix[0][0] * matrix[1][1]) -
(matrix[0][1] * matrix[1][0]);
    if (det == 0) {
        throw new IllegalArgumentException("Matriks tidak
dapat diinvers karena determinannya nol.");
    }
}

```

```
    }
    double[][] inverse = new double[2][2];
    inverse[0][0] = matrix[1][1] / det;
    inverse[0][1] = -matrix[0][1] / det;
    inverse[1][0] = -matrix[1][0] / det;
    inverse[1][1] = matrix[0][0] / det;
    return inverse;
}

// Fungsi untuk mengalikan dua matriks 2x2
public static double[][] multiplyMatrices(double[][]
firstMatrix, double[][] secondMatrix) {
    double[][] result = new double[2][2];
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++) {
            result[i][j] = 0;
            for (int k = 0; k < 2; k++) {
                result[i][j] += firstMatrix[i][k] *
secondMatrix[k][j];
            }
        }
    }
    return result;
}

// Fungsi untuk mencetak matriks
public static void printMatrix(double[][] matrix) {
    for (double[] row : matrix) {
        for (double column : row) {
            System.out.printf("%.2f ", column);
        }
        System.out.println();
    }
}
}
```

**Output:**

```
a.
Invers dari A:
0.20 0.40
0.40 -0.20

b.
Invers dari B:
0.67 0.33
-0.33 -0.67
```

c.  
 $B^{-1} A^{-1}$ :  
 0.27 0.20  
 -0.33 0.00

d.  
 $(AB)^{-1}$ :  
 0.27 0.20  
 -0.33 0.00

Sebagai latihan, silakan kerjakan soal berikut.

Diberikan  $A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 4 \\ 3 & 4 & 0 \end{bmatrix}$  dan  $B = \begin{bmatrix} -2 & 1 & 0 \\ 4 & -1 & 2 \\ 1 & 2 & 0 \end{bmatrix}$

1. Tentukan  $A^{-1}$  dan  $B^{-1}$
2. Apakah nilai  $AB^{-1}$  sama dengan  $A^{-1}B$ ? Tunjukkan dalam bentuk program Java dan outputnya.

# BAB III

Prosedur Pelaksanaan  
Praktikum  
Pemrograman  
Menggunakan Python

Praktikum pemrograman numerik dilakukan dalam tiga tahap yaitu persiapan, pelaksanaan, dan pelaporan dan penilaian.

### III.A. TAHAP PERSIAPAN

Untuk memulai praktikum pemrograman numerik, mahasiswa wajib melakukan hal-hal berikut:

- a. Menyediakan komputer (PC atau laptop) yang terkoneksi internet,
- b. Mengunduh installer Python,
- c. Melakukan instalasi dengan mengikuti panduan yang disampaikan dalam tahap pelaksanaan.

### III.B. TAHAP PELAKSANAAN

Pada tahap ini, kita akan mempelajari terlebih dahulu materi praktikum dan contoh output yang dihasilkan oleh Python. Seperti pada kegiatan praktikum komputasi, Anda diharapkan mempraktikkan materi dan contoh yang diberikan sehingga menghasilkan output seperti yang ditampilkan. Selanjutnya, kerjakan latihan yang diberikan dalam panduan ini secara mandiri. Pendampingan pelaksanaan praktikum akan dilakukan dalam praktikum online dan mahasiswa dapat memanfaatkan menu VPL (*Virtual Programming Library*) yang disediakan dalam kelas praktikum online.

#### III.B.1. Pengantar Python

Python adalah bahasa pemrograman yang dikembangkan oleh Guido van Rossum pada akhir 1980-an dan pertama kali dirilis pada tahun 1991. Nama "Python" terinspirasi dari acara komedi "Monty Python's Flying Circus," yang mencerminkan sifat ringan dan fleksibel dari bahasa ini. Tujuan utama dari pengembangan Python adalah untuk membuat bahasa yang mudah dibaca dan digunakan, sehingga pemrogram dapat fokus pada penyelesaian masalah, bukan detail teknis yang rumit. Python menggabungkan fitur-fitur dari berbagai bahasa, seperti modularitas dari Modula-3 dan penanganan kesalahan dari C, menjadikannya alat yang kuat untuk berbagai aplikasi.

Perkembangan Python tidak berhenti di versi pertama. Pada tahun 2000, Python 2.0 dirilis dengan perbaikan fitur signifikan, seperti *garbage collection* berbasis siklus dan *list comprehensions*, yang meningkatkan efisiensi penulisan kode. Meskipun banyak digunakan, Python 2 memiliki beberapa keterbatasan yang kemudian diatasi dalam versi selanjutnya. Python 2.7, yang dirilis pada 2010, adalah versi terakhir dari Python 2 dan akan terus menerima pembaruan hingga 2020, di mana dukungan resmi dihentikan. Pengembang kemudian diarahkan untuk bermigrasi ke Python 3, yang

membawa perubahan besar dan peningkatan pada sintaks serta manajemen memori.

Python 3.0 diluncurkan pada tahun 2008, sebagai perombakan besar untuk mengatasi kekurangan desain di Python 2. Namun, karena perbedaan besar antara kedua versi, transisi ke Python 3 berlangsung perlahan. Python 3 memperkenalkan banyak fitur baru yang mempermudah pengembangan aplikasi modern, seperti penanganan string Unicode secara default, penghapusan ketidakseragaman tipe data, dan peningkatan dukungan untuk pemrograman berorientasi objek. Python 3 juga menambahkan dukungan untuk fitur yang lebih relevan dengan komputasi modern, seperti concurrent programming dan pengolahan data dalam skala besar.

Pada tahun 2023, Python merilis versi terbarunya, Python 3.12. Versi ini menghadirkan berbagai peningkatan performa, seperti optimisasi bytecode dan perbaikan efisiensi memori. Python 3.12 juga menyertakan fitur baru yang mempermudah debugging dan meningkatkan keamanan kode, seperti analisis statis dan optimisasi runtime. Dengan fitur-fitur baru yang mendukung aplikasi pembelajaran mesin, analisis data besar, dan pengembangan web modern, Python terus menjadi pilihan utama bagi pengembang di berbagai industri, dari teknologi hingga akademisi.

### III.B.2. Instalasi Python

Untuk instalasi Python pada komputer, kita perlu mendapatkan sumber instalasi terlebih dahulu. Sumber tersebut dapat diakses secara online sehingga Anda harus mempunyai koneksi internet.

#### III.B.2.1. Instalasi Python pada desktop/laptop

» Download Installer Python

1. Kunjungi website Python [www.python.org/downloads/windows/](http://www.python.org/downloads/windows/)
2. Temukan rilis Python terbaru (Python 3) dan kemudian klik menu *Download*.



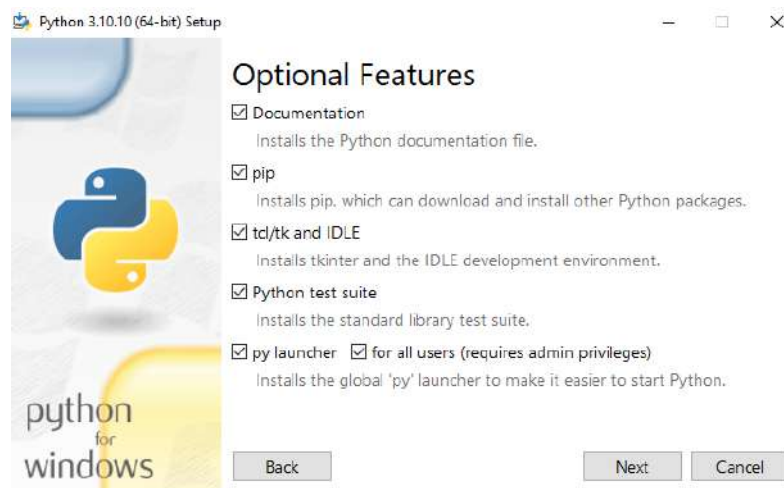
3. Klik tombol Download Python 3.xx.x (xx.x tergantung rilis terbaru)
4. Pada halaman Python rilis, silakan download installer sesuai dengan versi sistem operasi yang terpasang, apakah menggunakan Windows 32 bit atau 64 bit.

» Menjalankan Installer (Instal)

1. Setelah berhasil download, temukan file installer, dan klik 2x file python-3.xx.xx-mdxx-exe (xx adalah angka relase pythin dan versi OS yang tersedia).Python
2. Pilih opsi **Add python.exe to PATH**, agar pengguna bisa menjalankan python melalui *command line*.



3. Jika ingin menyimpan hasil instalasi dilain tempat, anda diberikan opsi **Customize installation**.
4. Pada window Optional Features bisa memilih fitur yang anda bisa sertakan ketika di instal.





- *Documentation*: direkomendasikan untuk kebutuhan dokumentasi python.
  - *pip*: diperlukan jika mau install paket-paket python dari luar (direkomendasikan).
  - *tcl/tk and IDLE*: menggunakan Tcl/Tk untuk membuat antarmuka pengguna grafisnya.
  - *Python test suite*: digunakan untuk testing dan pembelajaran.
  - *Py launcher dan for all users*: direkomendasikan jika mau menjalankan python dari command line.
5. Klik *Next* untuk masuk pada Window Advance Option (bairakn default).
  6. Terakhir klik tombol *install* dan tunggu sampai selesai.

» Ujicoba (verifikasi hasil instalasi) Python

1. Jalankan command prompt/terminal.
2. Dari prompt ketik: `python --version` untuk mengetahui python terinstal dan versi yang digunakan.
3. Selesai.

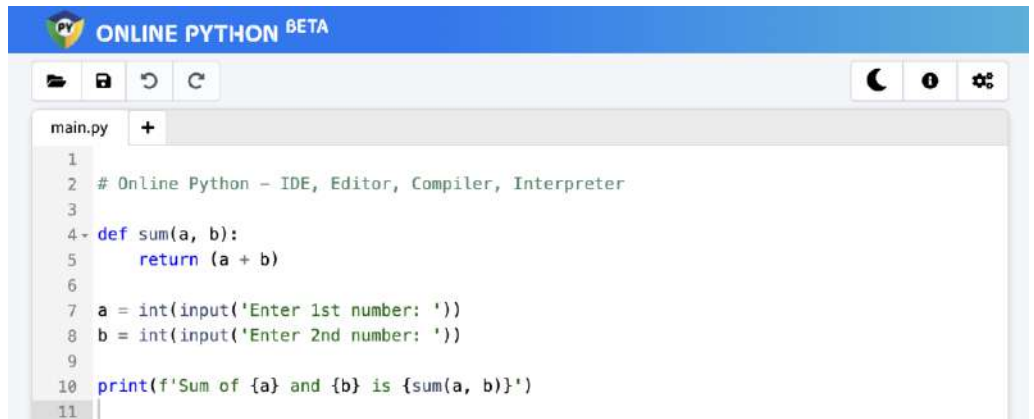
Anda dapat melihat video cara menginstal Python pada link berikut ini:



[https://www.youtube.com/watch?v=KzsvDQ\\_awH0](https://www.youtube.com/watch?v=KzsvDQ_awH0)

### III.B.2.2. Cara menjalankan Program Python secara online

1. Pastikan komputer Anda terhubung dengan internet
2. Jalankan/Buka Browser yang tersedia di komputer Anda.
3. Buka laman: <https://www.online-python.com>.



The screenshot shows the 'ONLINE PYTHON BETA' web interface. The code editor contains the following Python code:

```

1
2 # Online Python - IDE, Editor, Compiler, Interpreter
3
4 def sum(a, b):
5     return (a + b)
6
7 a = int(input('Enter 1st number: '))
8 b = int(input('Enter 2nd number: '))
9
10 print(f'Sum of {a} and {b} is {sum(a, b)}')
11

```


4. Untuk membuat file Python baru klik tombol (+).
5. Masukkan nama file, misalnya: *contoh1.py*.
6. Ketikkan kode program, contoh:

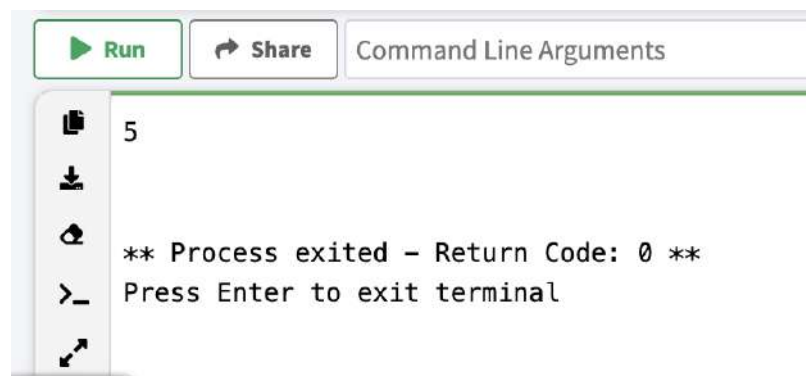
Contoh1.py

```

a = 2
b = 3
c = a + b
print(c)

```

7. Untuk menjalankan, klik tombol Run (  ).



The screenshot shows the execution interface of the Online Python IDE. It features a 'Run' button (a green play icon), a 'Share' button, and a 'Command Line Arguments' input field. Below these, a terminal window displays the output of the program:

```

5
** Process exited - Return Code: 0 **
>_ Press Enter to exit terminal

```

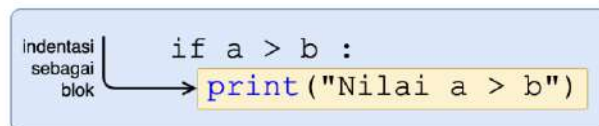
### III..B.3. Dasar-dasar Python

#### III.B.3.1. Penulisan Sintaks dan Indentasi

Sintaks Python adalah seperangkat aturan yang digunakan untuk membuat Program Python. Pada Python terdapat beberapa sub aturan penulisan, mulai dari struktur baris kode, statemen, komentar, penugasan, indentasi dan lainnya.

Secara umum penulisan pada Python memiliki sifat *case sensitive*, mulai dari keyword dan penamaan variabel, tidak menggunakan titik koma pada akhir sintaks, dan penggunaan indentasi yang ketat.

Indentasi digunakan sebagai pembentuk struktur dan blok dalam kode program Python. Begitu ketatnya indentasi pada Python sehingga baris program yang menggunakan spasi (menjorok ke dalam 1 spasi) dianggap sebagai bagian kode/blok program bagian atas.



```
if a > b :  
    print("Nilai a > b")
```

Sintaks `print("Nilai a > b")` adalah blok program dari sintaks `if (if a > b:)`.

Melihat pentingnya indentasi pada Python, penggunaan indentasi pada Python menjadi sebuah keharusan untuk diketahui dan dipahami dalam bahasa pemrograman Python.

#### III.B.3.2. Komentar

Komentar pada Python digunakan untuk menjelaskan kode Python. Penulisan komentar dapat diletakkan di atas atau samping kanan. Fungsi komentar pada bahasa pemrograman adalah sama, yang berbeda adalah sintaks atau kode yang digunakan yang menandakan suatu komentar. Komentar dalam bahasa pemrograman manapun tidak akan pernah dieksekusi oleh compiler. Dalam hal ini, komentar pada Python menggunakan tanda pagar (#).

Sintaks:
# komentar

**Program:**

```
# print("Bahasa Pemrograman Python")
```

```
print("Bahasa Pemrograman Python")
```

Penjelasan:

- Baris pertama tidak akan dieksekusi oleh kompiler.
- Baris kedua akan dieksekusi oleh compiler dengan mencetak tulisan ke layar.

**Output:**

```
Bahasa Pemrograman Python
```

### III.B.3.3. Import

Pernyataan import adalah salah satu fitur paling fundamental dalam bahasa pemrograman Python. Fitur ini memungkinkan Anda untuk memanfaatkan kode yang telah ditulis oleh pengembang lain dan diorganisasikan ke dalam modul-modul. Modul-modul ini berisi fungsi, kelas, dan variabel yang dapat Anda gunakan dalam program Anda sendiri. Dengan kata lain, import adalah cara untuk "meminjam" kode dari orang lain.

Import diperlukan karena beberapa hal berikut ini.

- *Efisiensi*: Tidak perlu menulis ulang kode yang sudah ada.
- *Modularitas*: Membagi kode menjadi modul-modul kecil membuat kode lebih terorganisir dan mudah dikelola.
- *Reusabilitas*: Modul dapat digunakan kembali dalam berbagai proyek.
- *Kolaborasi*: Modul memungkinkan pengembang berbagi kode dan bekerja sama.

**Sintaks:**

```
import nama_modul
```

**Program:**

```
import math
hasil = math.sqrt(16)
print(hasil)
```

Penjelasan:

- *import math*: Mengimport library math untuk digunakan.
- *math.sqrt(16)*: cara menggunakan fungsi *sqrt* dari library *math*.

**Output:**

```
4.0
```

### III.B.3.4. Variabel dan Tipe Data

Variabel adalah lokasi memori yang dicadangkan untuk menyimpan nilai-nilai. Ini berarti bahwa ketika Anda membuat sebuah variabel, Anda memesan beberapa ruang di memori. Variabel menyimpan data yang dilakukan selama program dieksekusi, yang nantinya isi dari variabel tersebut dapat diubah oleh operasi-operasi tertentu pada program yang menggunakan variabel.

Variabel dapat menyimpan berbagai macam tipe data. Di dalam pemrograman Python, variabel mempunyai sifat yang dinamis, artinya variabel Python tidak perlu dideklarasikan tipe data tertentu dan variabel Python dapat diubah saat program dijalankan.

Penulisan variabel Python sendiri juga memiliki aturan tertentu, yaitu:

1. Karakter pertama harus berupa huruf atau garis bawah/*underscore* ( `_` )
2. Karakter selanjutnya dapat berupa huruf, garis bawah/*underscore* ( `_` ) atau angka
3. Karakter pada nama variabel bersifat sensitif (*case-sensitif*) atau huruf kecil dan huruf besar dibedakan, contoh: *namaDepan* dan *namadepan* adalah variabel yang berbeda.

Tipe data adalah suatu media atau memori pada komputer yang digunakan untuk menampung informasi. Python memiliki tipe data yang cukup unik bila dibandingkan dengan bahasa pemrograman yang lain.

Berikut adalah tipe data dari bahasa pemrograman Python :

Tipe Data	Penjelasan	Contoh
Boolean	Menyatakan benar ( <i>True</i> ) yang bernilai 1, atau salah ( <i>False</i> ) yang bernilai 0	<i>True</i> atau <i>False</i>
String	Menyatakan karakter/kalimat bisa berupa huruf angka, dll (diapit tanda " atau ')	"True", "1", "a", "Sitta", "Hans", "Kani"
Integer	Menyatakan bilangan bulat	1 10 3 20
Float	Menyatakan bilangan yang mempunyai koma	0.1 1.2 10.5
Hexadecimal	Menyatakan bilangan dalam format heksa (bilangan berbasis 16)	9a 1d3
Complex	Menyatakan pasangan angka real dan imajiner	1 + 5j

List	Data untaian yang menyimpan berbagai tipe data dan isinya bisa diubah-ubah	["xyz", 786, 2.23]
Tuple	Data untaian yang menyimpan berbagai tipe data tapi isinya tidak bisa diubah	("xyz", 768, 2.23)
Dictionary	Data untaian yang menyimpan berbagai tipe data berupa pasangan penunjuk dan nilai	{"nama": "adi", "id": 2}

Untuk mulai membuat variabel dan nilai cukup menggunakan tanda **sama dengan** ( = ) diikuti dengan nilai data yang ingin dimasukkan, perhatikan sintaks berikut:

<b>Sintaks:</b>
#Inisiasi(proses memasukkan nilai data ke dalam variabel Nama_Variabel = Nilai_Data

Perlu diingat bahwa pada pemrograman Python, tipe data otomatis langsung menyesuaikan dari nilai data-nya, tidak perlu perlu dideklarasikan terlebih dahulu.

#### Program:

```
nama = "Budi"
print(nama); #proses mencetak nilai data dari variabel
```

#### Penjelasan:

- nama = "Budi" adalah nama variabel adalah nama, dan tipe datanya adalah string.

#### Output:

```
Budi
```

Contoh berikut dengan beberapa variabel dengan tipe data yang beragam.

#### Program:

```
# variable dengan tipe data string
a = "18 bulan"
print(a)
type(a) # mengecek tipe data dari variabel usia

# variable dengan tipe data integer
a = 18
```

```
print(a)
type(a) #mengecek tipe data dari variabel usia

# variable dengan tipe data float
A = 1.5
print(A)
type(A) #mengecek tipe data dari variabel Usia
```

Penjelasan:

- Variabel **a** diisi dengan data *string* (`a = "18 bulan"`).
- Variabel **a** kemudian ditimpa dengan data *integer* (`a = 18`).
- Variabel **A** diisi dengan data *float* (`A = 1.5`).

**Output:**

```
18 bulan
18
1.5
float
```

### III.B.3.5. Operator

Operator adalah simbol yang digunakan untuk melakukan operasi pada suatu nilai atau variabel, terdapat beberapa jenis operator dalam Python, yaitu:

#### 1. Operator Aritmatika

Operator aritmatika merupakan operator yang digunakan untuk melakukan operasi matematika dasar.

Operator	Penjelasan
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
//	Pembagian bulat
%	Modulus (sisa pembagian)
**	Pangkat

Berikut merupakan contoh penggunaan operator aritmatika dalam Python:

**Program:**

```
# asumsikan a bernilai 11 dan b bernilai 2
a = 11
b = 2

#proses mencetak hasil operasi aritmatika
print(a + b)
print(a - b)
print(a * b)
print(a / b)
print(a // b)
print(a % b)
print(a ** b)
```

**Penjelasan:**

- a dan b merupakan variabel yang akan dioperasikan.
- +, -, \*, dan seterusnya adalah operator aritmatika.

**Output:**

```
13
9
22
5.5
5
1
121
```

**2. Operator Relasional**

Operator relasional adalah operator yang digunakan untuk membandingkan dua nilai, baik berupa integer, float, string, atau boolean. Menghasilkan nilai kebenaran True atau False.

Operator	Penjelasan
==	Sama dengan
!=	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil sama dengan
>=	Lebih besar sama dengan



Berikut adalah contoh penggunaan operator relasional dalam Python:

**Program:**

```
x = 6
y = 7

#penggunaan operator relational
print(x == y)
print(x != y)
print(x < y)
print(x > y)
print(x <= y)
print(x >= y)
```

**Output:**

```
False
True
True
False
True
False
```

### 3. Operator Logika

Operator logika adalah operator yang digunakan untuk menggabungkan beberapa ekspresi logika dengan kedua operannya bertipe boolean dan mengembalikan hasil logika dengan nilai bertipe boolean (*True* atau *False*).

Operator	Penjelasan
<i>and</i>	Keduanya benar
<i>or</i>	Salah satu benar
<i>not</i>	Membalikkan nilai boolean

**Tabel kebenaran:**

<i>p</i>	<i>q</i>	<i>and</i>	<i>or</i>
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>

Berikut merupakan beberapa contoh penggunaan operator logika:

**Program:**

```
a = True
b = False
c = True
d = False

print(a and c)
print(a and b)
print(b or c)
print(b or b)
print(not a)
print(not b)
```

**Output:**

```
True
False
True
False
False
True
```

#### 4. Operator Assignment

Operator assignment adalah operator yang digunakan untuk memberi nilai kepada variabel dengan nilai tetap.

Operator	Penjelasan	Contoh
=	Penugasan sederhana	a = 2
+=	Penugasan dengan penjumlahan	a += 2 @ a = a + 2
-=	Penugasan dengan pengurangan	a -= 2 @ a = a - 2
*=	Penugasan dengan perkalian	a *= 2 @ a = a * 2
/=	Penugasan dengan pembagian	a /= 2 @ a = a / 2
//=	Penugasan dengan pembagian bulat	a //= 2 @ a = a // 2
%=	Penugasan dengan modulus (sisa hasil pembagian)	a %= 2 @ a = a % 2
**=	Penugasan dengan pangkat	a **= 2 @ a = a ** 2

Contoh penggunaan operator assignment pada Python:

**Program:**

```
# penugasan sederhana
a = 4
b = a
print(b)

# penugasan dengan penjumlahan
a = 15
a += 5
print(a)

# penugasan dengan pengurangan
a = 10
a -= 7
print(a)

# penugasan dengan perkalian
a = 5
a *= 3

print(a)
```

**Output:**

```
4
20
3
15
```

**III.B.3.6. Perulangan**

Perulangan pada pemrograman adalah sebuah konstruksi pemrograman yang terus-menerus diulang sampai memenuhi kondisi tertentu, dengan kata lain, perulangan memungkinkan sebuah program untuk dijalankan berkali-kali sampai kondisi tertentu terpenuhi. perulangan berguna untuk melakukan perintah yang dijalankan berulang kali. Pada pemrograman Python, terdapat dua jenis perulangan, yaitu: *for* dan *while*.

Perulangan *for* digunakan untuk mengulangi eksekusi blok kode dalam sebuah urutan.

**Sintaks**

```
for variabel dari objek iterasi:  
    # kode yang ingin dieksekusi pada setiap iterasi
```

Penjelasan:

- Variabel menyimpan nilai elemen saat ini dari *sequence*.
- *Sequence* adalah koleksi data yang akan diiterasi, misalnya *list*, *tuple*, atau *range*.
- aksi blok kode yang akan dieksekusi dalam setiap iterasi.

**Program:**

```
# Penggunaan perulangan for  
for n in range(5):  
    print(n)
```

Penjelasan:

- `range(5)` menghasilkan angka dari 0 hingga 4 (identasi pertama dimulai dari nol)
- `n` merupakan variabel setiap nilai dalam urutan tersebut, satu per satu. `print(n)` menampilkan nilai `n` dari setiap iterasi yang telah dilakukan.

**Output:**

```
0  
1  
2  
3  
4
```

## 2. Perulangan *while*

Perulangan *while* adalah menjalankan sebuah blok kode selama kondisi yang ditentukan bernilai *True*.

**Sintaks**

```
while kondisi:  
    aksi
```

Penjelasan:

- kondisi yang akan dievaluasi sebelum setiap iterasi selagi kondisi bernilai *True*. Namun akan berhenti berulang ketika kondisi bernilai *False*.
- aksi blok kode yang akan diindentasi di bawah pernyataan perulangan *while* yang akan dijalankan berulang kali selama kondisi bernilai *True*.

Program:

```
# Penggunaan while loop
a = 0
while a <= 5:
    print(a)
    a += 1
```

Penjelasan:

- `a = 0` merupakan inisiasi awal dengan memberikan nilai nol kepada variable `a`.
- `while a <= 5` mengindikasikan loop akan terus berjalan selama nilai `a` kurang dari atau sama dengan 5.
- Blok kode di dalam `while`
  - `print(a)` mencetak nilai `a` saat ini.
  - `a+=1` menambahkan nilai 1 ke nilai `a` (`a = a + 1`).
- Proses perulangan dimulai dari `a = 0`, kemudian program *while* akan menambahkan mengembalikan nilai `a` (`a += 1`). Kemudian program akan mengulangi proses ini sampai dengan `a` berjumlah kurang dari sama dengan 5.

Output:

```
0
1
2
3
4
5
```

### 3. Perulangan Nested

Perulangan Nested merupakan sebuah perulangan yang ditempatkan di dalam sebuah perulangan lainnya (perulangan dalam perulangan).

Perulangan *nested* sering digunakan untuk menangani struktur data yang lebih kompleks.

Agar dapat memahami perulangan *nested* dengan baik, perhatikan contoh berikut

**Program:**

```
for i in range(1, 5):  
    for j in range(2, 4):  
        print(i * j)  
    print()
```

Penjelasan:

- `for i in range(1, 5)` merupakan perulangan luar.
- `for j in range(2, 4)` merupakan perulangan dalam perulangan.

**Output:**

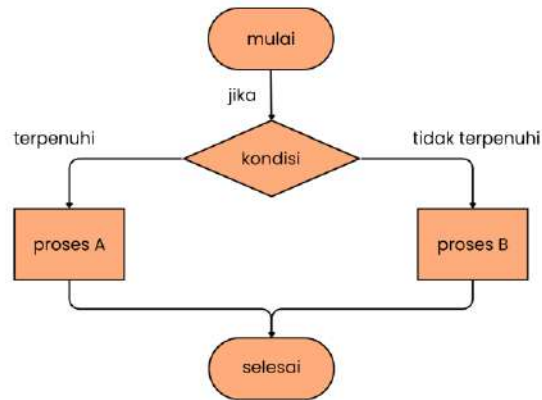
```
2  
3  
  
4  
6  
  
6  
9  
  
8  
12
```

### III.B.3.7. Seleksi

Salah satu kelebihan suatu bahasa pemrograman ialah pada kemampuannya untuk membuat struktur logika yang dapat dimanfaatkan untuk membuat percabangan alur program atau yang biasa dikenal sebagai *flowchart*. Pada *flowchart* tersebut, seringkali menggunakan struktur *If*.

Struktur *If* atau *If statement* adalah struktur kontrol pada python yang digunakan untuk membuat keputusan berdasarkan suatu kondisi. Ini memungkinkan program untuk mengambil tindakan yang berbeda berdasarkan apakah kondisi tertentu terpenuhi atau tidak.

Flowchart struktur If dapat digambarkan sebagai berikut:



Pada bahasa *Python* sendiri, terdapat beberapa logika kondisi diantaranya:

Equals	<code>a == b</code>
Not Equals	<code>a != b</code>
Less than	<code>a &lt; b</code>
Less than or equal to	<code>a &lt;= b</code>
Greater than	<code>a &gt; b</code>
Greater than or equal to	<code>a &gt;= b</code>

Selain itu, ada pula beberapa operator yang biasa digunakan pada If statement ini, diantaranya:

Operator	Fungsi
<code>&lt;</code>	Kurang dari
<code>&lt;=</code>	Kurang dari atau sama dengan
<code>&gt;</code>	Lebih dari
<code>&gt;=</code>	Lebih dari atau sama dengan
<code>==</code>	Sama dengan
<code>!=</code>	Tidak sama dengan

### Struktur Dasar If

Dalam bahasa pemrograman Python, struktur dasar If sangatlah sederhana. Bentuk struktur If dicontohkan seperti berikut:

**If kondisi:**

- Kondisi adalah bagian yang harus dipenuhi agar bagian kode yang ada di bawahnya dapat dijalankan oleh program python
- Setelah kondisi, akhiri dengan tanda titik dua ( : )
- Tulis kode pemrograman yang akan dijalankan apabila kondisi bernilai TRUE

**Program:**

```
usia = 15
if usia >= 14:
    print("Kamu sudah remaja")
```

**Penjelasan:**

Ketika sintaks di atas dijalankan, maka akan muncul tulisan “Kamu sudah remaja” yang artinya pada bagian program `print("Kamu sudah remaja")` kondisi yang dibuat menggunakan If yang bernilai TRUE.

**Output:**

```
Kamu sudah remaja
```

Lalu bagaimana jika usia = 12? Kita lihat pada contoh berikut:

**Program:**

```
usia = 12
if usia >= 14:
    print("Kamu sudah remaja")
```

Maka output yang keluar ketika sintaks tersebut di-run, tidak terdapat apa-apa atau dapat disebut FALSE. Mengapa demikian? Karena pada sintaks `usia = 12` tidak sama dengan sintaks `if usia == 14`. Sedangkan pada sintaks tersebut menggunakan logika `==` (equals) yang artinya sama dengan. Jadi variabel pada usia haruslah sama dengan variabel pada sintaks if tersebut.

**Else**

Pada contoh sebelumnya, kita telah menemukan FALSE pada sintaks kita. Maka dari itu, bahasa pemrograman Python memiliki tools lain untuk menghindari output FALSE tersebut, yaitu Else.

Else digunakan untuk membuat bagian program yang akan dijalankan apabila kondisi menghasilkan nilai FALSE. Cara penggunaan Else ini cukup mudah, yaitu cukup menuliskan kata kunci (*keyword*) dan diakhiri dengan titik dua.

**Program:**

```
n = int(input("Angka? "))
```



```
if n < 0:  
    print("Nilai absolut dari ", n, "adalah", -n)  
else:  
    print("Nilai absolut dari ", n, "adalah", n)
```

Penjelasan:

- Pada contoh di atas, pertama-tama sintaks tersebut meminta user memasukan angka yang disimpan pada variabel 'n'.
- Selanjutnya, program akan memeriksa apakah nilai n yang dimasukan kurang atau lebih dari 0. Jika kurang dari 0 maka akan bernilai TRUE, dan jika lebih dari 0 maka akan bernilai FALSE.

**Output:**

Ada 2 input, yang pertama akan diminta untuk memasukkan angka berapapun

Angka?

Kemudian, program akan memunculkan hasil nilai absolut dari angka yang dimasukkan (dalam contoh ini, kita akan mencoba untuk memasukan nilai -25)

```
Angka? -25  
Nilai absolut dari -25 adalah 25
```

**Elif**

Selain If dan Else, python juga memiliki tools yang dapat membantu untuk menyelesaikan masalah dengan percabangan yang lebih rumit dan lebih banyak dari contoh-contoh sebelumnya. Tools tersebut adalah Elif.

Namun, pada Elif ini, tools Else masih tetap digunakan sebagai alur percabangan akhir yang akan dijalankan oleh program.

**Program:**

```
a = 0  
while a < 10:  
    a = a + 1  
    if a > 5:  
        print(a, ">", 5)  
    elif a <= 3:  
        print(a, "<=", 3)  
    else:
```

```
print("Tidak ada yang benar")
```

**Output:**

```
1 <= 3
2 <= 3
3 <= 3
Tidak ada yang benar
Tidak ada yang benar
6 > 5
7 > 5
8 > 5
9 > 5
10 > 5
```

**III.B.3.8. Fungsi**

Fungsi adalah blok kode terorganisir yang digunakan untuk melakukan tugas tertentu. Fungsi membantu dalam memecah kode program menjadi modul-modul yang lebih kecil dan terstruktur, yang memudahkan dalam pemeliharaan dan pemahaman kode.

*Definisi Fungsi dalam Python*

Untuk mendefinisikan fungsi dalam Python, gunakan kata kunci *def* diikuti dengan nama fungsi, parameter dalam kurung, dan titik dua. Blok kode fungsi harus menggunakan indentasi.

**Sintaks:**

```
def nama_fungsi(parameter1, parameter2, ...):
    # blok kode
    return nilai_kembali
```

**Program:**

```
def sapa_nama(nama):
    print(f"Halo, {nama}!")
sapa_nama("Andi")
```

Penjelasan:

- `def sapa_nama(nama):` mendefinisikan fungsi `sapa_nama` yang menerima satu parameter `nama`.
- `print(f"Halo, {nama}!")` mencetak sapaan dengan `nama` yang diberikan.
- `sapa_nama("Andi")` memanggil fungsi dengan argumen "Andi".

**Output:**

```
Halo, Andi!
```

### Fungsi dengan Nilai Kembali

Fungsi dapat mengembalikan nilai menggunakan kata kunci *return*.

**Program:**

```
def tambah(a, b):  
    return a + b  
hasil = tambah(3, 5)  
print(hasil)
```

Penjelasan:

- `def tambah(a, b):` mendefinisikan fungsi `tambah` yang menerima dua parameter `a` dan `b`.
- `return a + b` mengembalikan hasil penjumlahan `a` dan `b`.
- `hasil = tambah(3, 5)` memanggil fungsi dan menyimpan hasilnya dalam variabel `hasil`.
- `print(hasil)` mencetak hasil penjumlahan.

**Output:**

```
8
```

### Parameter Default

Fungsi dapat memiliki parameter dengan nilai default. Jika tidak ada argumen yang diberikan untuk parameter tersebut, nilai akan digunakan.

**Program:**

```
def sapa(nama="Teman"):  
    print(f"Halo, {nama}!")  
sapa()  
sapa("Budi")
```

Penjelasan:

- `def sapa(nama="Teman"):` mendefinisikan fungsi `sapa` dengan parameter `nama` yang memiliki nilai default "Teman".
- `sapa()` memanggil fungsi tanpa argumen sehingga menggunakan nilai *default*.
- `sapa("Budi")` memanggil fungsi dengan argumen "Budi".

**Output:**

```
Halo, Teman!  
Halo, Budi!
```

### Fungsi dengan Argumen Variabel

Python memungkinkan penggunaan fungsi yang menerima jumlah argument yang tidak ditentukan menggunakan *\*args* dan *\*\*kwargs*.

**Program:**

```
def total(*angka):  
    return sum(angka)  
print(total(1, 2, 3, 4))  
print(total(10, 20))
```

Penjelasan:

- `def total(*angka):` mendefinisikan fungsi `total` yang dapat menerima jumlah argumen yang tidak terbatas.
- `return sum(angka)` mengembalikan jumlah dari semua argumen yang diberikan.
- `print(total(1, 2, 3, 4))` mencetak hasil penjumlahan dari argumen 1, 2, 3, dan 4.
- `print(total(10, 20))` mencetak hasil penjumlahan dari argumen 10 dan 20.

**Output:**

```
10  
30
```

### Fungsi Rekursif

Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri untuk menyelesaikan masalah yang lebih kecil dari masalah asli.

**Program:**

```
def faktorial(n):
```

```
if n == 1:
    return 1
else:
    return n * faktorial(n-1)
print(faktorial(5))
```

Penjelasan:

- `def faktorial(n)`: mendefinisikan fungsi faktorial yang menerima satu parameter `n`.
- `if n == 1`: kondisi dasar untuk menghentikan rekursi.
- `return n * faktorial(n-1)` menghitung faktorial dengan memanggil fungsi faktorial secara rekursif.
- `print(faktorial(5))` mencetak hasil faktorial dari 5.

**Output:**

```
120
```

### III.B.3.9. List, Tuple, Dictionary, dan Set

#### List

List adalah tipe struktur yang paling dasar dalam Bahasa pemrograman Python. *List* data menyimpanan elemen-elemen atau koleksi data, urutan data disebut indeks dan indeks pada Python selalu dimulai dari nol. Jadi ketika dalam *list* terdapat 4 anggota, maka indeks tertingginya adalah 3 (0, 1, 2, 3). Nilai sebuah List dapat diubah ketika sudah dideklarasikan dan diinisiasi.

`var_list_a = [ 30, 10, 25, 60 ]`

Indeks:    **0**        **2**        **3**        **4**

#### Sintaks:

```
nama_variabel_list = [ nilai_1, nilai_2, nilai_n, ... ]
```

Penjelasan:

- `Nama_variabel_list`: nama variabel yang akan terisi dengan elemen-elemen list

- Nilai-nilai list atau elemen dalam list akan selalu diapit oleh kurung siku [...].

**Program:**

```
# Deklarasi dan inisiasi variabel list
var_list_a = [ 30, 10, 25, 60 ]

# Cetak nilai nilai variabel list
print(var_list_a[0])
print(var_list_a[2])
# Memodifikasi nilai varaibel list
var_list_a[2] = 26
print(var_list_a[2])
```

**Penjelasan:**

- Variabel `var_list_a` diisi dengan nilai 30, 10, 25, dan 60
- `var_list_a[0]` adalah mengakses nilai list pada indeks 0
- `var_list_a[2]` adalah mengkases nilai list pada indeks 2.
- Mengubah nilai `var_list_a[2]` dari nilai 25 ke 26.
- Mengakses dan mencetak kembali list `var_list_a[2]`.

**Output:**

```
30
25
26
```

Akses dan cetak nilai List dengan perulangan dengan contoh berikut:

**Program:**

```
var_list_a = [ 30, 10, 25, 60 ]
for i in range(4):
    print(var_list_a[i])
```

**Output:**

```
30
10
25
60
```

**Tuple**

Tuple adalah salah satu tipe data struktur yang dapat menyimpan koleksi data, akses Tuple sama seperti akses List yaitu dengan indeks, Tuple

dicirikan dengan tanda kurung (...), Nilai Tuple tidak dapat diubah ketika sudah didelkarsi dan diinisiasi.

`var_tuple_a = (30, 10, 25, 60)`

Indeks: 0 2 3 4

**Sintaks:**

```
nama_variabel_tuple = ( nilai_1, nilai_2, nilai_n, ... )
```

Penjelasan:

- `nama_variabel_tuple`: nama variabel yang akan terisi dengan elemen-elemen tuple.
- Nilai-nilai tuple atau elemen dalam tuple akan selalu diapit oleh kurung (...).

**Program:**

```
var_tuple_a = ( 30, 10, 25, 60 )  
print(var_tuple_a[0])  
print(var_tuple_a[2])
```

Penjelasan:

- Variabel `var_tuple_a` diisi dengan nilai 30, 10, 25, dan 60
- `var_tuple_a[0]` adalah mengakses nilai tuple pada indeks 0
- `var_tuple_a[2]` adalah mengakses nilai tuple pada indeks 2.

**Output:**

```
30  
25
```

Akses dan cetak nilai Tuple dengan perulangan dengan contoh berikut:

**Program:**

```
var_tuple_a = [ 30, 10, 25, 60 ]  
for i in range(4):  
    print(var_tuple_a[i])
```

**Output:**

```
30
10
25
60
```

### Dictionary

Dictionary digunakan untuk menyimpan nilai yang disertai dengan kunci. Nilai dictionary bisa diubah tapi tidak bisa duplikasi, jika terjadi duplikasi, maka nilai duplikasi akan jadi nilai.

#### Sintaks:

```
nama_variabel_dict = ( kunci_1: nilai_1,
                       kunci_2: nilai_2,
                       kunci_n: nilai_n,
                       ....: ... )
```

Penjelasan:

- nama\_variabel\_dict: nama variabel yang akan terisi dengan elemen-elemen *dictionary*.
- Nilai-nilai dictionary diapit oleh kurung kurawal {...}.

#### Program:

```
var_dict_a = {
    "mtk": "Praktikum ABCD",
    "sks": 2,
    "nilai": 90
}
print(var_dict_a)
```

#### Output:

```
{'mtk': 'Praktikum ABCD', 'sks': 2, 'nilai': 90}
```

### III.B.3.10. Array

**Array** adalah struktur data yang digunakan untuk menyimpan kumpulan data dengan tipe data yang sama dan dapat diakses menggunakan indeks tertentu. Array membantu dalam mengorganisir data dengan cara yang sistematis sehingga mudah untuk diakses dan dimanipulasi.

#### Program:

```
# membuat array
```



```
angka = [2, 4, 6, 8, 10]
# mengakses elemen pertama
print('elemen pertama =', angka[0])
# mengakses elemen kedua
print('elemen kedua =', angka[1])
# mengakses elemen terakhir
print('elemen terakhir =', angka[-1])
# mengubah elemen dalam array
angka[0] = 17
print(angka)
```

**Output:**

```
elemen pertama = 2
elemen kedua = 4
elemen terakhir = 10
[17, 4, 6, 8, 10]
```

### III.B.4. Akar Persamaan dan Sistem Persamaan Linear

Materi ini telah dipelajari dalam mata kuliah Metode Numerik. Bahan ajar yang digunakan adalah MATA4213 Metode Numerik. Ada dua topik yang akan dibahas dalam Praktikum Komputasi dan Pemrograman ini.

Topik bahasan tentang Akar Persamaan yang akan dipelajari dalam praktikum ini adalah:

1. Metode Tertutup, meliputi metode Biseksi dan metode Regula Falsi.
2. Metode Terbuka, meliputi Iterasi Fixed-Point, metode Newton-Raphson, dan metode Secant.

Adapun topik tentang Sistem Persamaan Linear yang dipelajari adalah:

1. Metode Jacobi,
2. Metode Gauss-Seidel,
3. Metode Dekomposisi LU Doolittle,
4. Metode Dekomposisi LU Crout.

#### Praktikum III.B.4.1. Metode Biseksi

Pada Modul 2 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Biseksi. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 2.1 dalam BMP.

Dengan menggunakan metode Biseksi, tentukan akar real dari  $f(x) = -2x^5 - 1.6x^3 + 12x + 1$  di dalam selang  $[1,2]$  dengan  $\varepsilon = 10^{-5}$ .

**Program:**

```
def f(_x):
```

```

# Fungsi yang akan dicari akarnya
return (-2 * _x ** 5) - (1.6* _x ** 3) + (12 * _x) + 1

def bisection(_a, _b, _toleransi):
    # Metode biseksi untuk mencari akar
    if f(_a) * f(_b) >= 0:
        print("Tidak ada akar dalam interval yang diberikan.")
        return

    while (_b - _a) / 2 > _toleransi:
        _c = (_a + _b) / 2
        if f(_c) == 0:
            return _c
        elif f(_a) * f(_c) < 0:
            _b = _c
        else:
            _a = _c

    return (_a + _b) / 2

# Nilai awal interval dan toleransi
_a = 1
_b = 2
_toleransi = 1e-5

# Panggil fungsi biseksi
_akar = bisection(_a, _b, _toleransi)

print("Akar Real:", _akar)

```

**Output:**

```
Akar Real: 1.4661941528320312
```

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode Biseksi, tentukan akar real dari  $x^{3.5} = 80$  di dalam selang  $[2.0, 5.0]$  dengan  $\varepsilon = 10^{-5}$ .

**Praktikum III.B.4.2. Metode Regula Falsi**

Pada Modul 2 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Regula Falsi. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 2.3 dalam BMP.

Dengan menggunakan metode Regula Falsi, tentukan akar real dari  $f(x) = -2x^5 - 1.6x^3 + 12x + 1$  di dalam selang  $[1,2]$  dengan  $\varepsilon = 10^{-5}$ .

Program:

```
def f(_x):
    # Fungsi yang akan dicari akarnya
    return (-2 * _x ** 5) - (1.6* _x ** 3) + (12 * _x) + 1

def regula_falsi(_a, _b, _toleransi):
    # Metode Regula Falsi untuk mencari akar
    if f(_a) * f(_b) >= 0:
        print("Tidak ada akar dalam interval yang diberikan.")
        return

    while abs(_b - _a) > _toleransi:
        _c = (_a * f(_b) - _b * f(_a)) / (f(_b) - f(_a))
        if f(_c) == 0:
            return _c
        elif f(_a) * f(_c) < 0:
            _b = _c
        else:
            _a = _c

    return _c

# Nilai awal interval dan toleransi
_a = 1
_b = 2
_toleransi = 1e-5

# Panggil fungsi Regula Falsi
_akar = regula_falsi(_a, _b, _toleransi)

print("Akar Real:", _akar)
```

Output:

Akar Real: 1.4661888771239682

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode Regula-Falsi, tentukan akar real dari  $x^{3.5} = 80$  di dalam selang  $[2.0,5.0]$  dengan  $\varepsilon = 10^{-5}$ .

**Praktikum III.B.4.3. Metode Iterasi Fixed-Point**

Pada Modul 2 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Iterasi Fixed-Point. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 2.5 dalam BMP.

Dengan menggunakan metode iterasi *fixed-point*, tentukan akar real dari  $f(x) = -2x^5 - 1.6x^3 + 12x + 1 = 0$  dengan titik awal  $x_0 = 1.0$ .

**Program:**

```
import math

# Fungsi f(x) (tidak digunakan langsung dalam iterasi, hanya
# sebagai referensi)
def f(x):
    return -2 * x**2 - 1.6 * x**3 + 12 * x + 1

# Fungsi g(x) yang merupakan bentuk terbalik dari f(x) untuk
# fixed-point iteration
def g(x):
    return (-2 * x**2 - 1.6 * x**3 + 1) / -12

# Metode fixed-point iteration
def fixed_point_iteration(x0, tolerance, max_iterations):
    x_lama = x0
    iterasi = 0

    # Tampilkan nilai x awal
    print(f"Iterasi x{iterasi} = {x_lama:.6f}")

    for iterasi in range(1, max_iterations + 1):
        x_baru = g(x_lama)

        # Tampilkan nilai x setiap iterasi
        print(f"Iterasi x{iterasi} = {x_baru:.6f}")

        # Periksa apakah sudah mencapai toleransi
        if abs(x_baru - x_lama) < tolerance:
            break

        # Update nilai x lama dengan yang baru
        x_lama = x_baru

    # Hasil akhir
```

```
print(f"Akar mendekati: x = {x_lama:.6f} setelah {iterasi} iterasi")

# Contoh penggunaan
if __name__ == "__main__":
    x0 = 1.0          # Titik awal
    toleransi = 1e-6 # Toleransi
    maks_iterasi = 100 # Jumlah iterasi maksimum

    fixed_point_iteration(x0, toleransi, maks_iterasi)
```

### Output:

```
Iterasi x0 = 1.000000
Iterasi x1 = 0.216667
Iterasi x2 = -0.074153
Iterasi x3 = -0.082471
Iterasi x4 = -0.082275
Iterasi x5 = -0.082279
Iterasi x6 = -0.082279
Akar mendekati: x = -0.082279 setelah 6 iterasi
```

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode iterasi *fixed-point*, carilah akar real dari persamaan  $3x - e^x = 0$  dengan titik awal  $x_0 = 1.0$  dengan keakurasian sampai dengan  $10^{-5}$ .

### Praktikum III.B.4.4. Metode Newton-Raphson

Pada Modul 2 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Newton-Raphson. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 2.6 dalam BMP.

Dengan menggunakan metode Newton-Raphson, tentukan akar real dari  $f(x) = -2x^5 - 1.6x^3 + 12x + 1 = 0$  dengan titik awal  $x_0 = 1.0$ .

### Program:

```
import math

# Fungsi f(x)
def f(x):
```

```

    return -2 * x**5 - 1.6 * x**3 + 12 * x + 1

# Turunan dari f(x), yaitu f'(x)
def f_prime(x):
    return -10 * x**4 - 4.8 * x**2 + 12

# Metode Newton-Raphson
def newton_raphson(x0, tolerance, max_iterations):
    x_lama = x0
    iterasi = 0

    # Tampilkan nilai awal x0
    print(f"Iterasi {iterasi}: x = {x_lama:.6f}")

    for iterasi in range(1, max_iterations + 1):
        x_baru = x_lama - f(x_lama) / f_prime(x_lama)

        # Tampilkan nilai setiap iterasi
        print(f"Iterasi {iterasi}: x = {x_baru:.6f}")

        # Periksa apakah sudah mencapai toleransi
        if abs(x_baru - x_lama) < tolerance:
            break

        # Update nilai x lama
        x_lama = x_baru

    # Tampilkan akar yang ditemukan
    print(f"Akar mendekati: x = {x_lama:.6f} setelah {iterasi}
iterasi")

# Contoh penggunaan
if __name__ == "__main__":
    x0 = 1.0          # Titik awal
    toleransi = 1e-6 # Toleransi
    maks_iterasi = 100 # Jumlah iterasi maksimum

    newton_raphson(x0, toleransi, maks_iterasi)

```

**Output:**

```

Iterasi 0: x = 1.000000
Iterasi 1: x = 4.357143
Iterasi 2: x = 3.482969
Iterasi 3: x = 2.791247

```

```
Iterasi 4: x = 2.254943
Iterasi 5: x = 1.860455
Iterasi 6: x = 1.606826
Iterasi 7: x = 1.490822
Iterasi 8: x = 1.467102
Iterasi 9: x = 1.466190
Iterasi 10: x = 1.466189
Iterasi 11: x = 1.466189
Akar mendekati: x = 1.466189 setelah 11 iterasi
```

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode Newton-Raphson, carilah akar real dari persamaan  $3x - e^x = 0$  dengan titik awal  $x_0 = 1.0$  dengan keakurasian sampai dengan  $10^{-5}$ .

### Praktikum III.B.4.5. Metode Secant

Pada Modul 2 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Secant. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 2.7 dalam BMP.

Dengan menggunakan metode Secant, tentukan akar real dari  $f(x) = -2x^5 - 1.6x^3 + 12x + 1 = 0$  dengan titik awal  $x_0 = 1.0$  dan  $x_1 = 2$  dengan  $\varepsilon = 10^{-5}$ .

#### Program:

```
# Fungsi f(x) = -2x^5 - 1.6x^3 + 12x + 1
def f(x):
    return -2*x**5 - 1.6*x**3 + 12*x + 1

# Metode Secant
def secant_method(x0, x1, tol, max_iter=100):
    for i in range(max_iter):
        f_x0 = f(x0)
        f_x1 = f(x1)

        # Hindari pembagian dengan nol
        if abs(f_x1 - f_x0) < 1e-12:
            return None, i

        # Iterasi metode Secant
        x2 = x1 - f_x1 * (x1 - x0) / (f_x1 - f_x0)
```

```

# Jika akar ditemukan dengan toleransi yang diinginkan
if abs(f(x2)) < tol:
    return x2, i

# Update nilai x0 dan x1 untuk iterasi berikutnya
x0, x1 = x1, x2

return None, max_iter # Tidak konvergen

# Titik awal dan toleransi
x0 = 1.0
x1 = 2.0
tolerance = 1e-5

# Cari akar
root, iterations = secant_method(x0, x1, tolerance)
print(f"Akar: {root}, Ditemukan dalam {iterations} iterasi")

```

**Output:**

```
Akar: 1.466188815013839, Ditemukan dalam 7 iterasi
```

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode Secant, carilah akar real dari persamaan  $3x - e^x = 0$  dengan titik awal  $x_0 = 1.0$  dengan keakurasian sampai dengan  $10^{-5}$ .

**Praktikum III.B.4.6. Metode Jacobi**

Pada Modul 4 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Jacobi. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 4.2 dalam BMP.

Diberikan sistem persamaan linear  $Ax = b$  sebagai berikut:

$$\begin{aligned} 10x_1 - x_2 + 2x_3 &= 6 \\ -x_1 + 11x_2 - x_3 + 3x_4 &= 25 \\ 2x_1 - x_2 + 10x_3 - x_4 &= -11 \\ 3x_2 - x_3 + 8x_4 &= 15 \end{aligned}$$

Selesaikan sistem persamaan linear tersebut dengan menggunakan iterasi Jacobi yaitu  $\mathbf{x}^{(k)} = T\mathbf{x}^{(k-1)} + \mathbf{c}$ .

**Program:**

```
import numpy as np
```



```
def jacobi(A, b, tolerance=1, max_iterations=10000):  
    # membuat matriks 0 dengan ukuran matriks yang sama dengan  
    matriks b  
    x = np.zeros_like(b, dtype=np.double())  
  
    # np.diagonal(A) mengambil elemen diagonal dari matriks A  
    # np.diag() membuat matriks diagonal  
    T = A - np.diag(np.diagonal(A))  
  
    # loop iterasi jacobi  
    for k in range(max_iterations):  
  
        x_old = x.copy()  
  
        # np.dot() perkalian dot dari kedua matrix  
        x[0:len(b)] = (b - np.dot(T, x)) / np.diagonal(A)  
  
        # menghitung error  
        sum1 = np.linalg.norm(x - x_old, ord=np.inf)  
        sum2 = np.linalg.norm(x, ord=np.inf)  
  
        if sum1 / sum2 < tolerance:  
            print(f'Konvergen setelah iterasi ke-{k+1}')  
            break  
  
        if sum1 / sum2 > tolerance:  
            print(f'Belum konvergen setelah {k+1} kali iterasi')  
  
    print(f"Hasilnya setelah iterasi ke-{k+1}: {x}")  
    return  
  
jacobi(np.array([[10, -1, 2, 6],  
                [-1, 11, -1, 3],  
                [2, -1, 10, -1],  
                [0, 3, -1, 8]]),  
        np.array([6, 25, -11, 15]))
```

**Output:**

Konvergen setelah iterasi ke-2

Hasilnya setelah iterasi ke-2: [ 1.03018182 2.03693802 -  
1.0144562 0.98434122]

Sekarang perhatikan apabila toleransinya diperkecil

```
import numpy as np

def jacobi(A, b, tolerance=1e-10, max_iterations=10000):

    # membuat matriks 0 dengan ukuran matriks yang sama dengan
    # matriks b
    x = np.zeros_like(b, dtype=np.double())

    # np.diagonal(A) mengambil elemen diagonal dari matriks A
    # np.diag() membuat matriks diagonal
    T = A - np.diag(np.diagonal(A))

    # loop iterasi jacobi
    for k in range(max_iterations):

        x_old = x.copy()

        # np.dot() perkalian dot dari kedua matrix
        x[0:len(b)] = (b - np.dot(T, x)) / np.diagonal(A)

        # menghitung error
        sum1 = np.linalg.norm(x - x_old, ord=np.inf)
        sum2 = np.linalg.norm(x, ord=np.inf)

        if sum1 / sum2 < tolerance:
            print(f'Konvergen setelah iterasi ke-{k+1}')
            break

        if sum1 / sum2 > tolerance:
            print(f'Belum konvergen setelah {k+1} kali iterasi')

    print(f"Hasilnya setelah iterasi ke-{k+1}: {x}")
    return

jacobi(np.array([[10, -1, 2, 6],
                 [-1, 11, -1, 3],
                 [2, -1, 10, -1],
```

```
[0, 3, -1, 8]],  
np.array([6, 25, -11, 15]))
```

**Output:**

Konvergen setelah iterasi ke-28

Hasilnya setelah iterasi ke-28: [ 1. 2. -1. 1.]

Dari hasil yang diberikan, apa yang dapat Anda simpulkan?

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode Jacobi, tentukan nilai  $x_1$ ,  $x_2$ , dan  $x_3$  dari sistem persamaan linear berikut, dengan  $\mathbf{x}^{(0)} = \mathbf{0}$ .

$$3x_1 - x_2 + x_3 = 4$$

$$3x_1 + 6x_2 + 2x_3 = 21$$

$$3x_1 + 3x_2 + 7x_3 = 30$$

dengan toleransi kurang dari  $10^{-3}$ .

**Praktikum III.B.4.7. Metode Gauss-Seidel**

Pada Modul 4 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Gauss-Seidel. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 4.3 dalam BMP.

Diberikan sistem persamaan linear linear  $Ax = b$  sebagai berikut:

$$10x_1 - x_2 + 2x_3 = 6$$

$$-x_1 + 11x_2 - x_3 + 3x_4 = 25$$

$$2x_1 - x_2 + 10x_3 - x_4 = -11$$

$$3x_2 - x_3 + 8x_4 = 15$$

Selesaikan sistem persamaan linear tersebut menggunakan metode Gauss-Seidel dengan nilai pendekatan awal  $x^{(0)} = (0,0,0,0)^t$ .

**Program:**

```
import numpy as np  
  
def gauss_seidel(A, b, tolerance=1e-0, max_iterations=10000):  
    # membuat matriks 0 dengan ukuran matriks yang sama dengan  
    matriks b  
    x = np.zeros_like(b, dtype=np.double)  
  
    #loop iterasi gauss-seidel
```

```

for k in range(max_iterations):

    x_old = x.copy()

    # perkalian dot
    for i in range(A.shape[0]):
        x[i] = (b[i] - np.dot(A[i,:i], x[:i]) -
np.dot(A[i,(i+1):], x_old[(i+1):])) / A[i,i]

    #Stop condition
    if np.linalg.norm(x - x_old, ord=np.inf) /
np.linalg.norm(x, ord=np.inf) < tolerance:
        print(f'Konvergen setelah iterasi ke-{k+1}')
        break

    if np.linalg.norm(x - x_old, ord=np.inf) /
np.linalg.norm(x, ord=np.inf) >= tolerance:
        print(f'belum konvergen setelah {k+1} kali iterasi')

    print(f"Hasilnya setelah iterasi ke-{k+1}: {x}")
    return

gauss_seidel(np.array([[10, -1, 2, 0],
                      [-1, 11, -1, 3],
                      [2, -1, 10, -1],
                      [0, 3, -1, 8]]),
             np.array([6, 25, -11, 15]))

```

**Output:**

```

Konvergen setelah iterasi ke-2
Hasilnya setelah iterasi ke-2: [ 1.03018182  2.03693802 -
1.0144562  0.98434122]

```

Sekarang perhatikan apabila toleransinya diperkecil.

```

import numpy as np

def gauss_seidel(A, b, tolerance=1e-10, max_iterations=10000):

    # membuat matriks 0 dengan ukuran matriks yang sama dengan
    matriks b
    x = np.zeros_like(b, dtype=np.double)

```

```
#loop iterasi gauss-seidel
for k in range(max_iterations):

    x_old = x.copy()

    # perkalian dot
    for i in range(A.shape[0]):
        x[i] = (b[i] - np.dot(A[i,:i], x[:i]) -
np.dot(A[i,(i+1):], x_old[(i+1):])) / A[i ,i]

    #Stop condition
    if np.linalg.norm(x - x_old, ord=np.inf) /
np.linalg.norm(x, ord=np.inf) < tolerance:
        print(f'Konvergen setelah iterasi ke-{{k+1}}')
        break

    if np.linalg.norm(x - x_old, ord=np.inf) /
np.linalg.norm(x, ord=np.inf) >= tolerance:
        print(f'belum konvergen setelah {{k+1}} kali iterasi')

    print(f"Hasilnya setelah iterasi ke-{{k+1}}: {x}")
    return

gauss_seidel(np.array([[10, -1, 2, 0],
    [-1, 11, -1, 3],
    [2, -1, 10, -1],
    [0, 3, -1, 8]]),
    np.array([6, 25, -11, 15]))
```

**Output:**

Konvergen setelah iterasi ke-11

Hasilnya setelah iterasi ke-11: [ 1. 2. -1. 1.]

Dari hasil yang diberikan, apa yang dapat Anda simpulkan?

Sebagai latihan, silakan kerjakan soal berikut ini.

Dengan menggunakan metode metode Gauss-Seidel, tentukan nilai  $x_1$ ,  $x_2$ , dan  $x_3$  dari sistem persamaan linear berikut, dengan  $\mathbf{x}^{(0)} = \mathbf{0}$ .

$$3x_1 - x_2 + x_3 = 4$$

$$3x_1 + 6x_2 + 2x_3 = 21$$

$$3x_1 + 3x_2 + 7x_3 = 30$$

dengan toleransi kurang dari  $10^{-3}$ .

**Praktikum III.B.4.8. Metode Dekomposisi LU Doolittle**

Pada Modul 5 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Dekomposisi LU Doolittle. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 5.1 dalam BMP.

Diketahui sistem persamaan linear berikut:

$$-3x_1 + 2x_2 - x_3 = -1$$

$$6x_1 - 6x_2 + 7x_3 = -7$$

$$3x_1 - 4x_2 + 4x_3 = -6$$

Selesaikan sistem persamaan linear tersebut dengan menggunakan metode Dekomposisi LU Doolittle.

**Program:**

```
import numpy as np

# Fungsi untuk dekomposisi LU menggunakan metode Doolittle
def doolittle_decomposition(A):
    n = len(A)
    L = np.zeros((n, n))
    U = np.zeros((n, n))

    for i in range(n):
        # Mengisi matriks U
        for j in range(i, n):
            U[i, j] = A[i, j] - sum(L[i, k] * U[k, j] for k in
range(i))

        # Mengisi matriks L
        L[i, i] = 1 # Diagonal L harus 1 pada metode Doolittle
        for j in range(i + 1, n):
            L[j, i] = (A[j, i] - sum(L[j, k] * U[k, i] for k in
range(i))) / U[i, i]

    return L, U

# Definisikan matriks koefisien A dan vektor konstanta B
A = np.array([[ -3,  2, -1],
              [  6, -6,  7],
              [  3, -4,  4]])

B = np.array([-1, -7, -6])
```

```
# Lakukan dekomposisi LU dengan metode Doolittle
L, U = doolittle_decomposition(A)

# Langkah 1: Menyelesaikan L * y = B
y = np.linalg.solve(L, B)

# Langkah 2: Menyelesaikan U * x = y
x = np.linalg.solve(U, y)

# Menampilkan hasil akhir
print("Matriks L:")
print(L)
print("\nMatriks U:")
print(U)
print("\nSolusi x:")
print(x)
```

**Output:**

```
Matriks L:
[[ 1.  0.  0.]
 [-2.  1.  0.]
 [-1.  1.  1.]]

Matriks U:
[[-3.  2. -1.]
 [ 0. -2.  5.]
 [ 0.  0. -2.]]

Solusi x:
[ 2.  2. -1.]
```

Sebagai latihan, silakan kerjakan soal berikut.

Dengan menggunakan metode Dekomposisi LU Doolittle, tentukan nilai  $x_1$ ,  $x_2$ , dan  $x_3$  dari sistem persamaan linear berikut:

$$\begin{aligned}3x_1 - x_2 + x_3 &= 4 \\3x_1 + 6x_2 + 2x_3 &= 21 \\3x_1 + 3x_2 + 7x_3 &= 30\end{aligned}$$

**Praktikum III.B.4.9. Metode Dekomposisi LU Crout**

Pada Modul 5 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Dekomposisi LU Crout. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 5.4 dalam BMP.

Diketahui sistem persamaan linear berikut:

$$-3x_1 + 2x_2 - x_3 = -1$$

$$6x_1 - 6x_2 + 7x_3 = -7$$

$$3x_1 - 4x_2 + 4x_3 = -6$$

Selesaikan sistem persamaan linear tersebut dengan menggunakan metode Dekomposisi LU Crout.

**Program:**

```
import numpy as np

# Fungsi untuk melakukan dekomposisi LU dengan metode Crout
def crout_decomposition(A):
    n = len(A)
    L = np.zeros((n, n))
    U = np.eye(n) # Matriks U dengan elemen diagonal 1

    for j in range(n):
        for i in range(j, n):
            L[i, j] = A[i, j] - sum(L[i, k] * U[k, j] for k in
range(j))
        for i in range(j + 1, n):
            U[j, i] = (A[j, i] - sum(L[j, k] * U[k, i] for k in
range(j))) / L[j, j]

    return L, U

# Definisikan matriks koefisien A dan vektor konstanta B
A = np.array([[ -3,  2, -1],
              [  6, -6,  7],
              [  3, -4,  4]])

B = np.array([-1, -7, -6])

# Lakukan dekomposisi LU dengan metode Crout
L, U = crout_decomposition(A)

# Langkah 1: Menyelesaikan L * y = B
y = np.linalg.solve(L, B)

# Langkah 2: Menyelesaikan U * x = y
x = np.linalg.solve(U, y)

# Menampilkan hasil akhir x, L, dan U
print("Matriks L:")
```



```
print(L)
print("\nMatriks U:")
print(U)
print("\nSolusi x:")
print(x)
```

### Output:

```
Matriks L:
[[-3.  0.  0.]
 [ 6. -2.  0.]
 [ 3. -2. -2.]]

Matriks U:
[[ 1.         -0.66666667  0.33333333]
 [ 0.          1.         -2.5       ]
 [ 0.          0.          1.         ]]

Solusi x:
[ 2.  2. -1.]
```

Sebagai latihan, silakan kerjakan soal berikut.

Dengan menggunakan metode Dekomposisi LU Crout , tentukan nilai  $x_1$ ,  $x_2$ , dan  $x_3$  dari sistem persamaan linear berikut:

$$\begin{aligned}3x_1 - x_2 + x_3 &= 4 \\3x_1 + 6x_2 + 2x_3 &= 21 \\3x_1 + 3x_2 + 7x_3 &= 30\end{aligned}$$

### III.B.5. Interpolasi Polinomial dan Masalah Nilai Awal

Materi ini juga telah dipelajari dalam mata kuliah Metode Numerik. Topik bahasan tentang Interpolasi Polinomial yang akan dipelajari dalam praktikum ini adalah:

1. Metode Lagrange.
2. Metode Newton Beda Terbagi.

Adapun topik tentang Masalah Nilai Awal yang dipelajari adalah Metode Runge-Kutta.

#### Praktikum III.B.5.1. Metode Lagrange

Pada Modul 6 BMP MATA4213 Metode Numerik, telah dibahas tentang Metode Lagrange. Silakan Anda baca kembali materi tersebut.

Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 6.1 dalam BMP.

Diberikan tiga data suhu, yaitu (0,3.85), (20,0.8), dan (40,0.212). Gunakan interpolasi polinomial Lagrange order pertama dan order kedua untuk menentukan nilai pada saat suhu 15°C.

**Program:**

```
import numpy as np

# Data suhu (derajat Celsius, suhu)
x = np.array([0, 20, 40])
y = np.array([3.85, 0.8, 0.212])

# Titik yang ingin diinterpolasi
x_interp = 15

# Fungsi interpolasi Lagrange
def lagrange_interpolation(x, y, x_interp):
    n = len(x)
    y_interp = 0
    for i in range(n):
        # Menghitung nilai L_i(x)
        L_i = 1
        for j in range(n):
            if i != j:
                L_i *= (x_interp - x[j]) / (x[i] - x[j])
        y_interp += y[i] * L_i
    return y_interp

# Interpolasi orde 1 (menggunakan 2 titik terdekat)
# Kita akan menggunakan titik (0, 3.85) dan (20, 0.8) karena 15
# lebih dekat ke interval ini
x1 = np.array([0, 20])
y1 = np.array([3.85, 0.8])
y_interp_orde1 = lagrange_interpolation(x1, y1, x_interp)

# Interpolasi orde 2 (menggunakan semua titik data)
y_interp_orde2 = lagrange_interpolation(x, y, x_interp)

print("Nilai interpolasi orde 1 pada suhu 15 derajat Celsius:",
      y_interp_orde1)
```

```
print("Nilai interpolasi orde 2 pada suhu 15 derajat Celsius:",  
y_interp_orde2)
```

**Output:**

```
Nilai interpolasi orde 1 pada suhu 15 derajat Celsius: 1.5625  
Nilai interpolasi orde 2 pada suhu 15 derajat Celsius:  
1.3316875
```

Sebagai latihan, silakan kerjakan soal berikut ini.

Diketahui empat pasangan titik, yaitu (1,3), (2,6), (3,19), dan (5,99).  
Tentukan interpolasi Lagrange order 3.

**Praktikum III.B.5.2. Metode Newton Beda Terbagi**

Pada Modul 6 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Newton Beda Terbagi. Silakan Anda baca kembali materi tersebut. Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 5.4 dalam BMP.

Diketahui kurva  $y = \ln(x)$  dengan dua titik yaitu titik  $x = 1$  dan  $x = 6$ .  
Temukan nilai pendekatan untuk  $x = 2$  dengan menggunakan metode Newton Beda Terbagi.

**Program:**

```
# Fungsi untuk menghitung beda terbagi pertama  
def divided_difference(x0, x1, f_x0, f_x1):  
    return (f_x1 - f_x0) / (x1 - x0)  
  
# Fungsi untuk polinomial Newton dengan dua titik  
def newton_interpolation(x, x0, x1, f_x0, f_x1):  
    f_01 = divided_difference(x0, x1, f_x0, f_x1)  
    return f_x0 + f_01 * (x - x0)  
  
# Titik-titik yang diberikan  
x0 = 1  
x1 = 6  
f_x0 = 0          # ln(1)  
f_x1 = 1.79176   # ln(6)  
x_value = 2      # Nilai x yang ingin kita aproksimasi  
  
# Menghitung nilai pendekatan untuk x = 2  
result = newton_interpolation(x_value, x0, x1, f_x0, f_x1)  
print(f"Pendekatan ln(2) menggunakan metode Newton Beda  
Terbagi: {result}")
```

**Output:**

Pendekatan  $\ln(2)$  menggunakan metode Newton Beda Terbagi:  
0.358352

Sebagai latihan, silakan kerjakan soal berikut.

Diketahui  $y = f(x) = \frac{1}{x}$  dan tiga titik yaitu  $x_0 = 2$ ,  $x_1 = 2.75$ , dan  $x_2 = 4$ . Tentukan interpolasi dengan metode beda terbagi untuk tiga titik tersebut.

**Praktikum III.B.5.3. Metode Runge Kutta**

Pada Modul 6 BMP MATA4213 Metode Numerik, telah dibahas tentang Algoritma Runge Kutta. Silakan Anda baca kembali materi tersebut.

Selanjutnya, silakan Anda perhatikan contoh berikut. Contoh ini merupakan soal Contoh 10.5 dalam BMP.

Dengan menggunakan metode Runge-Kutta Order Keempat, hitunglah nilai  $y$  dari

$$y' = 4e^{0.8t} - 0.5y$$

dari  $t = 0$  hingga 4 dengan nilai langkah  $h = 1$ . Kondisi awal pada saat  $t = 0$  adalah  $y = 2$ . Perlu untuk diketahui bahwa solusi analitik diperoleh sebagai berikut.

$$y = \frac{4}{1.3}(e^{0.8t} - e^{-0.5t}) + 2e^{-0.5t}$$

**Program:**

```
import math

# Fungsi turunan y' = f(t, y)
def f(t, y):
    return 4 * math.exp(0.8 * t) - 0.5 * y

# Solusi analitik yang diberikan
def analytical_solution(t):
    return (4 / 1.3) * (math.exp(0.8 * t) - math.exp(-0.5 * t)) + 2 * math.exp(-0.5 * t)

# Implementasi Metode Runge-Kutta Orde 4 (RK4) dengan pelacakan error
def runge_kutta_4_with_error_tracking(t0, y0, t, h):
```

```

n = int((t - t0) / h)
y = y0
results = []
for i in range(n):
    k1 = h * f(t0, y)
    k2 = h * f(t0 + 0.5 * h, y + 0.5 * k1)
    k3 = h * f(t0 + 0.5 * h, y + 0.5 * k2)
    k4 = h * f(t0 + h, y + k3)
    yrk4 = y + (1 / 6) * (k1 + 2 * k2 + 2 * k3 + k4)

    # Hitung solusi analitik dan kesalahan relatif
    exact_y = analytical_solution(t0 + h)
    relative_error = abs((yrk4 - exact_y) / exact_y) * 100
# dalam persen

    results.append([i + 1, k1, k2, k3, k4, yrk4,
relative_error])

    y = yrk4
    t0 = t0 + h
return results

# Inisialisasi nilai
t0 = 0 # Nilai awal t
y0 = 2 # Kondisi awal y(t=0)
t = 4 # Titik di mana kita ingin mencari solusi y
h = 1 # Ukuran langkah

# Dapatkan hasil per iterasi
error_tracking_results =
runge_kutta_4_with_error_tracking(t0, y0, t, h)

# Tampilkan hasil dalam bentuk tabel sederhana
print(f"{'Iterasi':<10}{'k1':<15}{'k2':<15}{'k3':<15}{'k4':<1
5}{'yrk4':<15}{'error (%)':<20}")
print("-" * 90)
for row in error_tracking_results:

print(f"{'row[0]:<10}{'row[1]:<15.6f}{'row[2]:<15.6f}{'row[3]:<15
.6f}{'row[4]:<15.6f}{'row[5]:<15.6f}{'row[6]:<20.6f}")

```

**Output:**

Iterasi (%)	k1	k2	k3	k4	yrk4	error
1	3.000000	4.217299	3.912974	5.945677	6.201037	0.103407

2	5.801645	8.729538	7.997565	12.712829	14.862484	0.125046
3	12.380888	19.029761	17.367542	27.977693	33.721348	0.131176
4	27.232032	42.109905	38.390437	62.074228	75.439172	0.133011

Sebagai latihan, silakan kerjakan soal berikut.

Carilah solusi pendekatan untuk persamaan diferensial dibawah ini.

$$y' = y - x^2 + 1, 0 \leq x \leq 2, y(0) = 0.5$$

yang mempunyai penyelesaian eksak adalah

$$y(x) = (x + 1)^2 - 0.5e^x$$

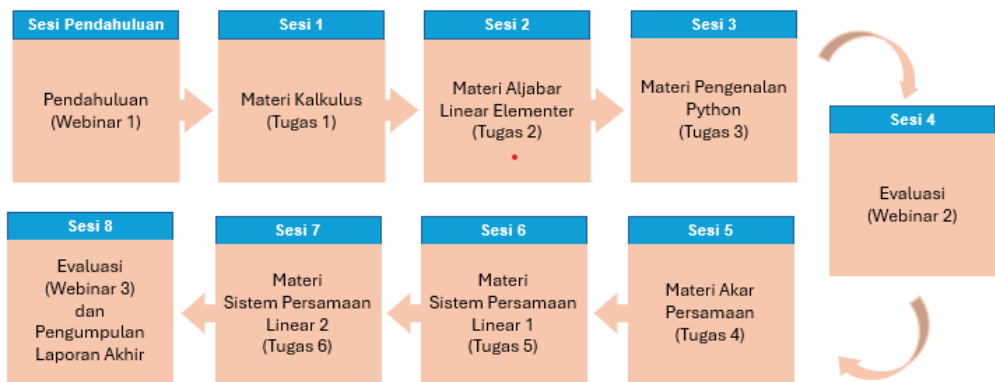
dengan menggunakan metode Runge Kutta Order Keempat.

# BAB IV

## Praktikum Online

Pada tahap ini, Anda akan mengikuti proses praktikum melalui <https://elearning.ut.ac.id>. Dalam kegiatan praktikum online ini, Anda akan melakukan praktikum secara mandiri (asinkronus) dan melalui webinar (sinkronus). Kegiatan praktikum online ini dilaksanakan selama 8 (delapan) minggu.

Dalam praktikum online, aktivitas belajar yang dilaksanakan dapat dilihat dalam skema berikut ini:



Gambar 4.1. Skema aktivitas belajar dalam praktikum online

Secara detail, aktivitas belajar yang dilakukan dalam setiap sesi adalah sebagai berikut.

Tabel 1. Aktivitas belajar tutor dan mahasiswa dalam praktikum online

Sesi	Mahasiswa/ Tutor	Aktivitas	Keterangan
Pendahuluan	Mahasiswa	<ul style="list-style-type: none"> <li>- Mempelajari Panduan Praktikum Komputasi dan Pemrograman</li> <li>- Mengikuti Webinar 1</li> <li>- Mengunduh format laporan akhir praktikum</li> </ul>	Dilaksanakan sebelum dimulainya pelaksanaan tutorial/praktikum online
	Tutor	<ul style="list-style-type: none"> <li>- Membuat dan mengumumkan link webinar di Sesi Pendahuluan</li> <li>- Mempresentasikan materi tentang proses praktikum, bentuk pelaporan,</li> </ul>	



Sesi	Mahasiswa/ Tutor	Aktivitas	Keterangan
		dan skema penilaian dalam webinar.	
1	Mahasiswa	<ul style="list-style-type: none"> <li>- Mengerjakan Tugas 1.</li> <li>- Memindahkan hasil praktikum atau tugas ke format laporan.</li> </ul>	Tugas dikumpulkan pada setiap sesi
	Tutor	<ul style="list-style-type: none"> <li>- Mengingatkan mahasiswa untuk mengerjakan tugas tepat pada waktunya</li> <li>- Memberikan umpan balik dari Tugas 1 yang dikumpulkan mahasiswa jika diperlukan</li> <li>- Merespon pertanyaan/komentar dalam forum diskusi</li> </ul>	
2	Mahasiswa	<ul style="list-style-type: none"> <li>- Mengerjakan Tugas 2</li> <li>- Memindahkan hasil praktikum atau tugas ke format laporan</li> </ul>	Tugas dikumpulkan pada setiap sesi
	Tutor	<ul style="list-style-type: none"> <li>- Mengingatkan mahasiswa untuk mengerjakan tugas tepat pada waktunya</li> <li>- Memberikan umpan balik dari Tugas 2 yang dikumpulkan mahasiswa jika diperlukan</li> <li>- Merespon pertanyaan/komentar dalam forum diskusi</li> </ul>	
3	Mahasiswa	<ul style="list-style-type: none"> <li>- Mengerjakan Tugas 3</li> </ul>	Tugas dikumpulkan pada setiap sesi

Sesi	Mahasiswa/ Tutor	Aktivitas	Keterangan
		- Memindahkan hasil praktikum atau tugas ke format laporan	
	Tutor	- Mengingatn mahasiswa untuk mengerjakan tugas tepat pada waktunya - Memberikan umpan balik dari Tugas 3 yang dikumpulkan mahasiswa jika diperlukan - Merespon pertanyaan/ komentar dalam forum diskusi	
4	Mahasiswa	Mempresentasikan Tugas 1, 2, dan 3.	
	Tutor	- Membuat dan mengumumkan link webinar di Sesi 4 - Merespon hasil presentasi - Menilai Tugas 1, 2, dan 3 berdasarkan hasil presentasi mahasiswa -	
5	Mahasiswa	- Mengerjakan Tugas 4 - Memindahkan hasil praktikum atau tugas ke format laporan	Tugas dikumpulkan pada setiap sesi
	Tutor	- Mengingatn mahasiswa untuk mengerjakan tugas tepat pada waktunya - Memberikan umpan balik dari Tugas 4 yang dikumpulkan mahasiswa jika diperlukan	

Sesi	Mahasiswa/ Tutor	Aktivitas	Keterangan
		<ul style="list-style-type: none"> <li>- Merespon pertanyaan/ komentar dalam forum diskusi</li> </ul>	
6	Mahasiswa	<ul style="list-style-type: none"> <li>- Mengerjakan Tugas 5</li> <li>- Memindahkan hasil praktikum atau tugas ke format laporan</li> </ul>	Tugas dikumpulkan pada setiap sesi
	Tutor	<ul style="list-style-type: none"> <li>- Mengingatkan mahasiswa untuk mengerjakan tugas tepat pada waktunya</li> <li>- Memberikan umpan balik dari Tugas 5 yang dikumpulkan mahasiswa jika diperlukan</li> <li>- Merespon pertanyaan/ komentar dalam forum diskusi</li> </ul>	
7	Mahasiswa	<ul style="list-style-type: none"> <li>- Mengerjakan Tugas 6</li> <li>- Memindahkan hasil praktikum atau tugas ke format laporan</li> </ul>	Tugas dikumpulkan pada setiap sesi
	Tutor	<ul style="list-style-type: none"> <li>- Mengingatkan mahasiswa untuk mengerjakan tugas tepat pada waktunya</li> <li>- Memberikan umpan balik dari Tugas 6 yang dikumpulkan mahasiswa jika diperlukan</li> <li>- Merespon pertanyaan/ komentar dalam forum diskusi</li> </ul>	

Sesi	Mahasiswa/ Tutor	Aktivitas	Keterangan
8	Mahasiswa	Mempresentasikan Tugas 4, 5, dan 6.	
	Tutor	<ul style="list-style-type: none"> <li>- Membuat dan mengumumkan link webinar di Sesi 8</li> <li>- Merespon hasil presentasi</li> <li>- Menilai Tugas 4, 5, dan 6 berdasarkan hasil presentasi mahasiswa</li> </ul>	

### PENILAIAN HASIL PRAKTIKUM

Nilai akhir mata kuliah STMA4224 Praktikum Komputasi dan Pemrograman berasal dari 100% nilai akhir praktikum online. Dalam hal ini, penilaian akhir praktikum online STMA4224 Praktikum Komputasi dan Pemrograman diberikan oleh tutor pada sistem LMS-UT (<https://elearning.ut.ac.id>). Tutor menilai dengan menggunakan Format Penilaian yang dapat dilihat pada lampiran. Adapun komponen penilaian akhir mata kuliah adalah sebagai berikut.

Tabel 2. Komponen Penilaian Akhir

No.	Komponen	Persentase
1	Tugas 1	10
2	Tugas 2	10
3	Tugas 3	10
4	Presentasi dalam Webinar 2	15
5	Tugas 4	10
6	Tugas 5	10
7	Tugas 6	10
8	Presentasi dalam Webinar 3	15
9	Laporan akhir	10
<b>Total</b>		<b>100</b>

Keberhasilan mahasiswa dalam mengikuti mata kuliah ini ditunjukkan dengan keaktifan mahasiswa dalam mengikuti proses praktikum online. Aktivitas mahasiswa dapat terlihat dari tingkat kehadiran, partisipasi dalam

forum diskusi, pengumpulan tugas, dan presentasi saat webinar. Untuk itu, siapkan diri Anda dalam mengikuti praktikum online dan pastikan ketersediaan sumber daya (komputer/laptop) dan jaringan internet di lingkungan Anda.



# Lampiran

## 1. FORMAT LAPORAN AKHIR

Laporan akhir ditulis dalam bentuk file doc. atau pdf., dan berukuran A4. Adapun isi laporan terdiri dari:

- Cover laporan, menampilkan nama praktikum, masa registrasi, nama dan NIM mahasiswa, dan nama tutor/laboran yang mendampingi dalam praktikum online.
- Pendahuluan, memaparkan latar belakang dan tujuan kegiatan praktikum.
- Pelaksanaan, menampilkan jawaban tugas 1, 2, 3, 4, 5, dan 6.
- Kesimpulan dan saran.

Format laporan akhir dapat diakses di kelas praktikum online atau website Prodi Matematika FST UT.

## 2. FORM PENILAIAN PRAKTIKUM

### a. Komponen Penilaian Tugas

No.	Komponen
1	Kesesuaian jawaban dengan soal yang diberikan
2	Program benar dan dapat dijalankan
3	Output program benar dan sesuai dengan soal yang diberikan
4	Tidak plagiasi
5	Ketepatan waktu pengumpulan tugas

### b. Komponen Penilaian Presentasi

No.	Komponen
1	Dapat menjelaskan program dan output yang dikerjakan
2	Dapat menunjukkan program dan output yang diperoleh adalah hasil pekerjaan sendiri
3	Bersikap sopan dan menghargai pendapat orang lain

### c. Komponen Penilaian Laporan Akhir

No.	Komponen
1	Lengkap, berisikan jawaban Tugas 1, 2, 3, 4, 5, dan 6

No.	Komponen
2	Mengikuti format laporan akhir yang diberikan
3	Ketepatan waktu pengumpulan laporan



## Daftar Pustaka

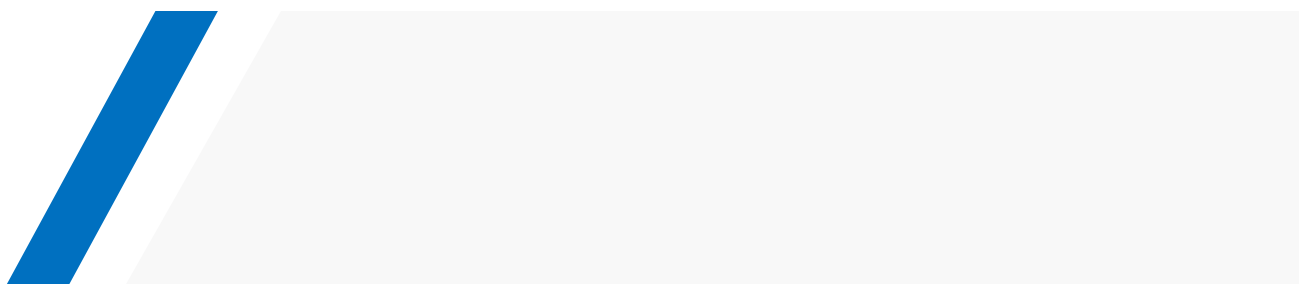
Chairul Imron. (2023). BMP MATA4213 Metode Numerik. Tangerang Selatan: Universitas Terbuka.

Kani. (2022). BMP MSIM4203 Algoritma dan Pemrograman. Tangerang Selatan: Universitas Terbuka.

Pamuntjak, R.J. dan Warsito (2022). BMP MATA4112 Aljabar Linear Elementer I. Tangerang Selatan: Universitas Terbuka.

Warsito. (2016). BMP MATA4110 Kalkulus I. Tangerang Selatan: Universitas Terbuka.





Kover dalam

Halaman  
Prancis

## » Kata Pengantar

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa yang telah memberikan wawasan dan ilmu pengetahuan serta pemahaman untuk mempelajarinya.

Mulai tahun akademik 2025/2026, Program Studi S1 Matematika FST UT menawarkan mata kuliah STMA4325 Praktikum Visualisasi dan Optimasi sebanyak 2 (dua) sks. Untuk menempuh mata kuliah tersebut, mahasiswa wajib mempunyai Panduan Praktikum Visualisasi dan Optimasi ini. Karena dalam panduan, mahasiswa akan mendapatkan penjelasan tentang konsep, prosedur, dan petunjuk pelaksanaan praktikum mulai persiapan, pelaksanaan, dan asesmen praktikum.

Panduan Praktikum Visualisasi dan Optimasi ini dapat terwujud atas kerjasama tim penulis dengan beberapa mahasiswa. Kami mengucapkan terima kasih dan penghargaan setinggi-tingginya atas waktu, ilmu, dan kerja cerdasnya sehingga panduan ini dapat selesai tepat pada waktunya. Kami menyadari bahwa panduan ini masih banyak kekurangan. Untuk itu, saran dan koreksi dari para pengguna sangat berguna untuk menyempurnakan materi dalam panduan ini. Apabila ada hal-hal yang akan disampaikan, silakan mengirimkan email ke [prodimatematikafst@ecampus.ut.ac.id](mailto:prodimatematikafst@ecampus.ut.ac.id).

Tangerang Selatan, Desember 2024

Salam,

Tim Penulis:

1. Darsih Idayani
2. Asmara Iriani Tarigan
3. Siti Umamah Naili Muna

Mahasiswa:

Ahmad Hairul Bahri

Annisa

Siti Sopiiah Sopiyan

# » Daftar Isi

<b>Kata Pengantar</b>	iii
<b>Daftar isi</b>	iv
<b>BAB 1 Pendahuluan</b>	2
A. Latar Belakang	2
B. Tujuan	10
C. Ruang Lingkup	15
D. Persyaratan Peserta	16
<b>BAB 2 Prosedur Pelaksanaan Praktikum Menggunakan GeoGebra</b>	28
A. Persiapan	54
B. Praktikum Visualisasi Fungsi	55
1. Materi	68
2. Pelaksanaan	72
C. Praktikum VisualisasiGraf	55
1. Materi	68
2. Pelaksanaan	72
D. Praktikum Pemrograman Linear dan Tak Linear	55
1. Materi	68
2. Pelaksanaan	72

E. Pelaporan dan Penilaian	72
<b>BAB 3 Prosedur Pelaksanaan Praktikum Menggunakan Excel</b>	<b>28</b>
A. Persiapan	54
B. Praktikum Visualisasi Data	55
1. Materi	68
2. Pelaksanaan	72
C. Pelaporan dan Penilaian	55
<b>BAB 4 Prosedur Pelaksanaan Praktikum Menggunakan Solver</b>	<b>28</b>
A. Persiapan	54
B. Praktikum Pemrograman Linear	55
1. Materi	68
2. Pelaksanaan	72
C. Praktikum Masalah Transportasi	55
1. Materi	68
2. Pelaksanaan	72
D. Praktikum Masalah Penugasan	55
1. Materi	68
2. Pelaksanaan	72
E. Praktikum Pemrograman Tak Linear Satu Variabel	55

1. Materi	68
2. Pelaksanaan	72
<hr/>	
<hr/>	
F. Praktikum Pemrograman Tak Linear Banyak Variabel	55
1. Materi	68
2. Pelaksanaan	72
G. Pelaporan dan Penilaian	72
<b>Lampiran</b>	<b>79</b>
<b>Daftar Pustaka</b>	<b>80</b>

# BAB 1

## Pendahuluan



# »» Pendahuluan

## A. LATAR BELAKANG

Pada era digital ini, kebutuhan akan lulusan yang terampil dalam menggunakan teknologi informasi dan komunikasi semakin besar. Oleh karena itu, Program Studi S1 Matematika FST UT melakukan perubahan kurikulum menjadi berorientasi luaran atau Outcome Based Education. Salah satu bentuk perubahan tersebut adalah dengan memberikan mata kuliah STMA4325 Praktikum Visualisasi dan Optimasi.

Mata kuliah Praktikum Visualisasi dan Optimasi mulai ditawarkan kepada mahasiswa pada 2025/2026 Ganjil. Mata kuliah ini merupakan mata kuliah yang wajib ditempuh mahasiswa. Selama mempelajari mata kuliah ini, mahasiswa akan mengikuti proses pembelajaran yang memberikan pengalaman dan keterampilan visualisasi dan optimasi. Oleh karena itu, mahasiswa harus menggunakan komputer dan aplikasi terkait visualisasi dan optimasi secara langsung. Agar proses tersebut berjalan dengan baik, pendampingan praktikum akan dilaksanakan melalui tutorial online sehingga mahasiswa wajib mengikuti tutorial online.

## B. TUJUAN

Setelah mempelajari mata kuliah STMA4325 Praktikum Visualisasi dan Optimasi, mahasiswa diharapkan mampu menggunakan software atau bahasa pemrograman untuk menentukan solusi suatu masalah matematika. Adapun capaian pembelajaran lulusan yang dibebankan pada mata kuliah ini adalah mahasiswa mampu (1) menguasai konsep teoritis matematika meliputi logika matematika, matematika diskrit, aljabar, analisis, dan geometri, serta teori peluang dan statistika, (2) mampu mengkomunikasikan dan mengembangkan pemikiran matematis, yang diawali dari penguasaan prosedural/komputasi hingga penguasaan yang luas meliputi eksplorasi, penalaran logis, generalisasi, abstraksi, dan bukti formal, (3) mampu mengamati, mengidentifikasi, merumuskan, dan memecahkan masalah melalui pendekatan matematis dengan melibatkan bantuan teknologi, (4) mampu merekonstruksi/modifikasi, menganalisis/berpikir secara terstruktur terhadap model matematika dari suatu sistem/masalah nyata, serta mengkaji keakuratan dan menginterpretasikan hasil, dan (5) mampu mengambil keputusan yang tepat dengan menggunakan berbagai alternatif pemecahan masalah matematis yang telah tersedia.

### C. RUANG LINGKUP

Panduan Praktikum Visualisasi dan Optimasi ini menjelaskan tentang tahapan pelaksanaan praktikum yang terdiri atas persiapan, pelaksanaan, dan evaluasi.

1. Pada tahapan persiapan dijelaskan tentang proses instalasi aplikasi pemrograman dan materi pendukung yang diperlukan.
2. Pada tahapan pelaksanaan diberikan materi dan contoh praktikum yang dapat dipraktikkan secara mandiri. Selain itu, diberikan pula soal-soal latihan untuk dikerjakan secara mandiri. Adapun verifikasi program dan hasil output yang diperoleh akan dibahas dalam praktikum online.
3. Pada tahapan praktikum online, akan dijelaskan aktivitas pembelajaran, pelaporan, dan penilaian praktikum online. Selain itu, akan disampaikan pula format pelaporan dan skema penilaian mata kuliah ini. Sebagai bukti proses praktikum, Anda akan diminta membuat laporan akhir praktikum dengan menggunakan format yang terdapat dalam lampiran.

### D. STRUKTUR PANDUAN

Secara garis besar, Panduan Praktikum Visualisasi dan Optimasi ini terdiri atas tiga topik, yaitu:

1. Topik 1 membahas tentang Prosedur Pelaksanaan Praktikum Visualisasi Menggunakan GeoGebra.
2. Topik 2 membahas tentang Prosedur Pelaksanaan Praktikum Visualisasi data Menggunakan Excel.
3. Topik 3 membahas tentang Praktikum Optimasi Menggunakan Solver.

Dalam panduan praktikum ini, materi yang digunakan merujuk pada BMP MATA4110 Kalkulus I Edisi 3, BMP MATA4443 Analisis Jaringan Edisi 2, BMP MATA4222 Pemrograman Linear Edisi 2, dan BMP MATA4303 Riset Operasi Edisi 1. Mahasiswa dapat menggunakan bahan ajar yang sesuai untuk mendukung proses pembelajaran dalam praktikum ini.

### E. PERSYARATAN PESERTA

Mata kuliah STMA4325 Praktikum Visualisasi dan Optimasi merupakan mata kuliah wajib bagi mahasiswa Program Studi S1 Matematika FST UT. Dengan demikian, peserta mata kuliah ini adalah mahasiswa yang telah melakukan registrasi mata kuliah STMA4325 Praktikum Visualisasi dan Optimasi. Sebagai prasyarat mempelajari mata kuliah ini, mahasiswa sebaiknya telah menempuh mata kuliah Kalkulus I/Kalkulus Diferensial, Analisis Jaringan/Pengantar Analisis Jaringan, Pemrograman Linear, dan Riset Operasional I/Riset Operasi.

# BAB 2

## Prosedur Pelaksanaan Praktikum Menggunakan GeoGebra

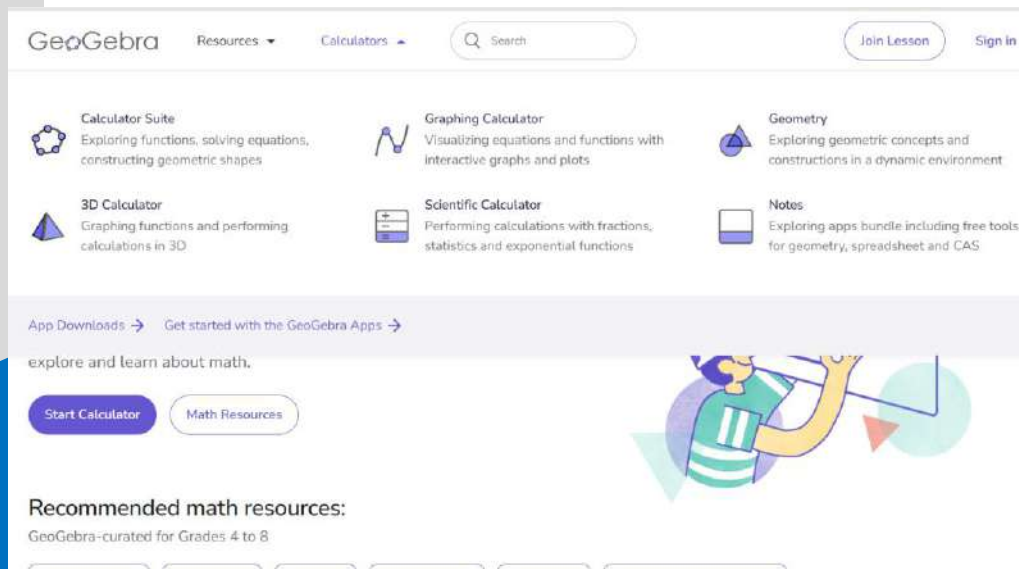
# »» Prosedur Pelaksanaan Praktikum Menggunakan GeoGebra

## A. PERSIAPAN

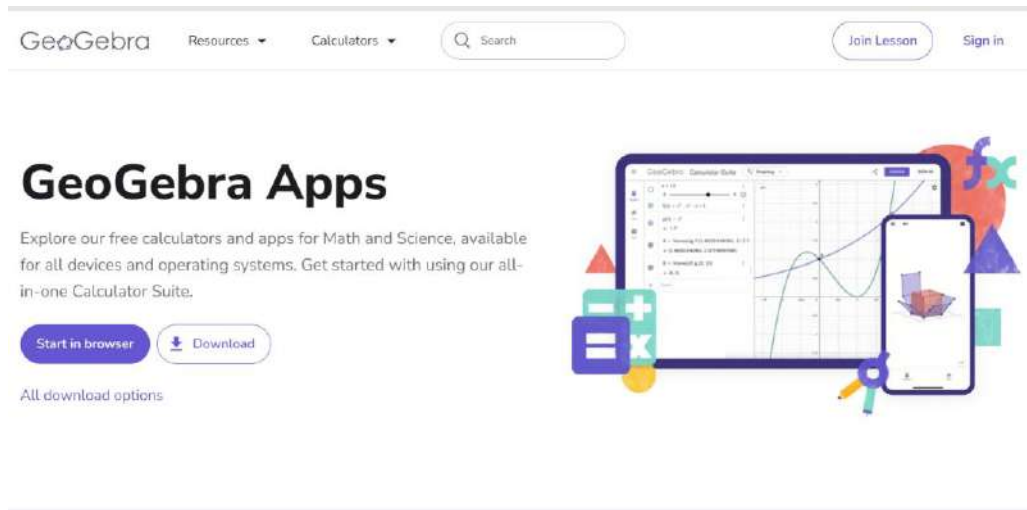
### 1. Mengunduh dan Menginstal GeoGebra

GeoGebra dapat digunakan di ponsel pintar, tablet, laptop, atau komputer dengan berbagai sistem seperti Linux, Unix, Mac OS X, Windows, dan platform lainnya yang dapat menjalankan Java. GeoGebra dapat digunakan secara daring atau diunduh untuk diinstal di komputer dengan mengunjungi situs web <http://www.geogebra.org>.

Untuk menggunakan secara daring, klik “Calculators” di bagian atas dan klik “Calculator Suite” (lihat Gambar 2.1). Selain menggunakan cara tersebut, dapat langsung mengunjungi situs web <https://www.geogebra.org/calculator>. Masih di situs web <http://www.geogebra.org>, scroll ke bawah sampai menemukan bagian “What we offer” kemudian klik “Explore All →” di “Math Calculators & Apps”. Setelah ke laman GeoGebra Apps, klik “Start in browser” untuk menuju ke laman Calculator Suite dan menggunakannya secara daring atau klik “Download” untuk mengunduh secara gratis (lihat Gambar 1.2). Selain menggunakan cara tersebut, untuk menuju laman GeoGebra Apss dapat mengunjungi situs web <https://www.geogebra.org/download>. Geogebra dapat diinstal di komputer dengan klik dua kali berkas hasil unduhan yang bertipe application. Setelah diinstal, dapat langsung dibuka dan digunakan.



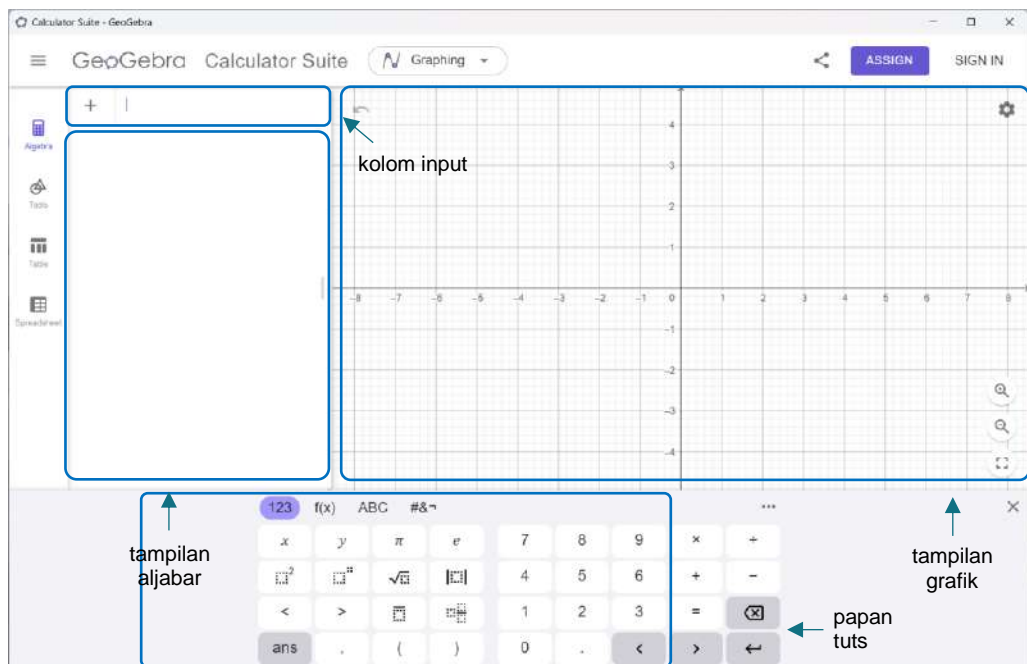
Gambar 2.1 Laman <http://www.geogebra.org>



**Gambar 2.2** Laman <https://www.geogebra.org/download>


## 2. Komponen GeoGebra

GeoGebra memiliki tampilan awal seperti yang ditunjukkan pada Gambar 2.3 dengan beberapa fitur utama, yaitu menu (Algebra, Tools, Table, dan Spreadsheet), kolom input, tampilan aljabar, tampilan grafik, dan papan tuts. Selain itu, di bagian kiri atas terdapat Main Menu dengan ikon tiga garis.



**Gambar 2.3** Tampilan awal GeoGebra




### 3. Menyimpan file GeoGebra

Gambar yang dihasilkan GeoGebra dapat disimpan dalam berbagai jenis file, yaitu Geogebra file (.ggb), PNG image (.png), SVG image (.svg), PDF document (.pdf), dan 3D print (.stl) dengan klik ikon  "Main Menu" pada bagian kiri atas, kemudian klik "Download as" dan pilih jenis file yang diinginkan. Selain itu, dapat menggunakan pilihan "Save to your computer" untuk menyimpan langsung dalam bentuk GeoGebra file dan "Export image" jika ingin menyimpan dalam bentuk PNG file.

### 4. Bekerja dengan GeoGebra

Fitur-fitur yang ada dalam GeoGebra yang telah dijelaskan sebelumnya dapat digunakan untuk membuat titik, menggambar garis, sudut, bangun datar, grafik, dan bangun ruang. Untuk membuat titik, menggambar garis, sudut, bangun datar dan grafik dapat menggunakan GeoGebra Calculator Suite dengan memilih Graphing di bagian atas, di sebelah tulisan Calculator Suite (lihat Gambar 2.3). Sedangkan untuk menggambar bangun ruang, GeoGebra Calculator Suite dengan memilih 3D Calculator di tempat yang sama.



#### a. Membuat Titik

Untuk membuat titik dapat dilakukan dengan klik menu Tools  dan klik Point  yang ada pada bagian Basic Tools kemudian pilih posisi titik yang akan dibuat di tampilan grafik. Point  juga ada pada bagian Points.

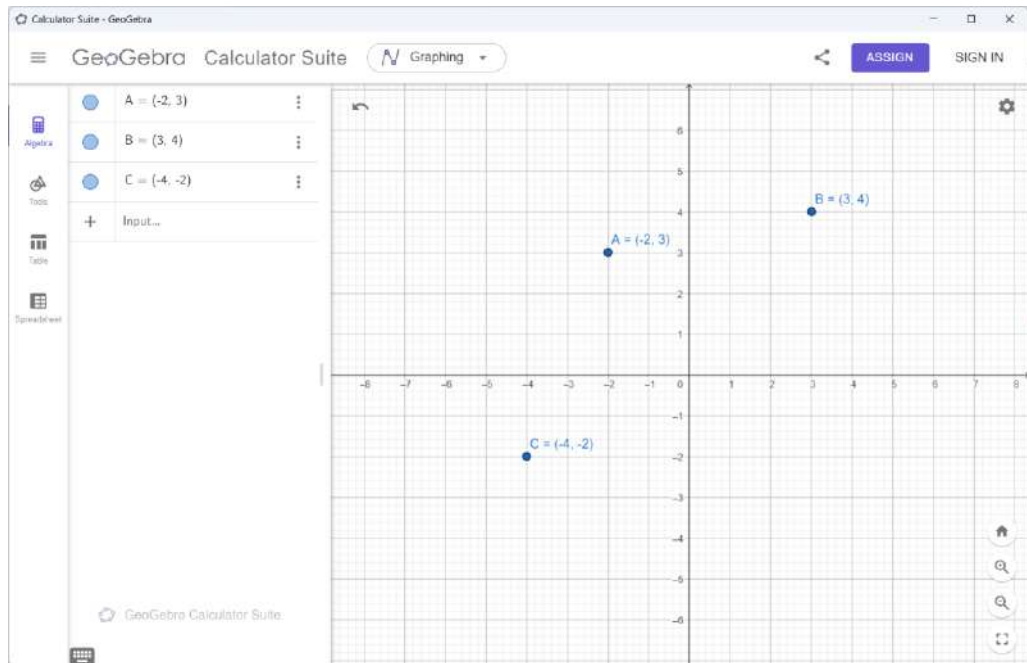
#### Praktikum 1.1

Gambarlah titik  $A(-2,3)$ ,  $B(3,4)$ , dan  $C(-4,-2)$ !

Berikut ini adalah langkah-langkah untuk menggambar titik  $A$ ,  $B$ , dan  $C$  dengan cara menggambar langsung di bagian tampilan grafik:


- Klik Tools 
- Klik Point  pada bagian Basic Tools
- Letakkan kursor di tampilan grafik pada  $x = -2$  dan  $y = 3$
- Letakkan kursor di tampilan grafik pada  $x = 3$  dan  $y = 4$
- Letakkan kursor di tampilan grafik pada  $x = -4$  dan  $y = -2$

Cara lain dapat dilakukan dengan menyetik  $A(-2,3)$ ,  $B(3,4)$ , dan  $C(-4,-2)$  di kolom input satu persatu. Untuk menampilkan nilai  $x$  dan  $y$  pada titik  $A$ , klik kanan pada titiknya, klik Settings, centang pada Show Label, kemudian pilih Name & Value. Tampilan aljabar dan tampilan grafik dapat dilihat pada Gambar 2.4.



**Gambar 2.4** Titik  $A(-2,3)$ ,  $B(3,4)$ , dan  $C(-4,-2)$




### b. Membuat Garis



Untuk membuat garis dapat dilakukan dengan klik menu Tools  pilih garis yang akan dibuat pada bagian “Lines”. Terdapat beberapa pilihan, Segment, Line, Ray, dan Vector. Selain itu, terdapat bagian “Construct” dengan beberapa pilihan antara lain Perpendicular Line untuk membuat garis tegak lurus, Parallel Line untuk membuat garis sejajar, dll.

#### Praktikum 1.2

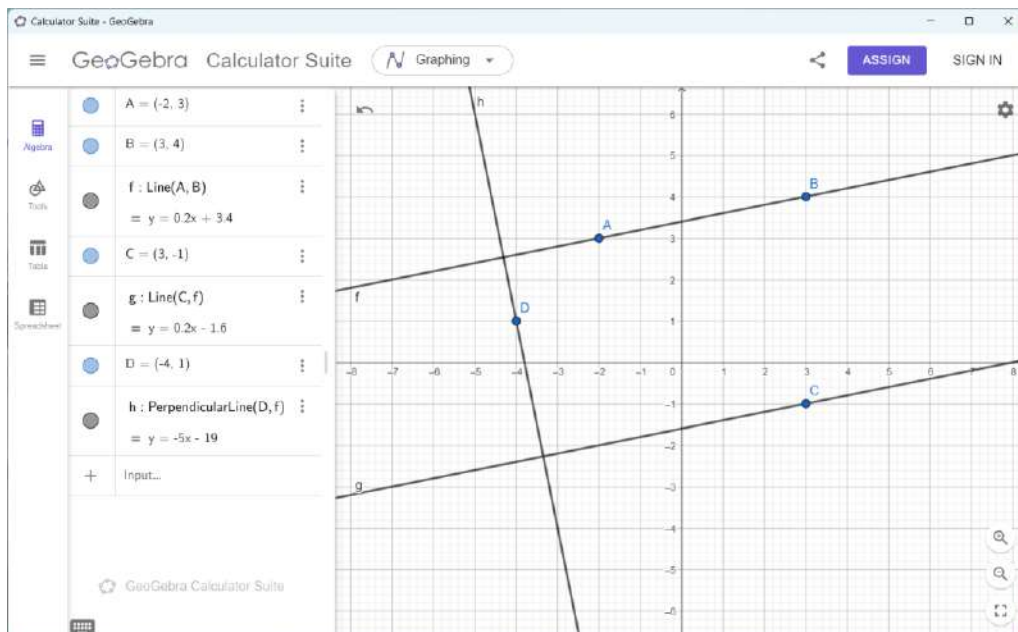
Gambarlah garis yang melalui dua titik, yaitu titik  $A(-2,3)$  dan  $B(3,4)$ . Kemudian gambar garis sejajar dan garis tegak lurus dengan garis pertama!

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.2:

- Klik Tools 
- Pilih Line  pada bagian Lines
- Letakkan kursor di tampilan grafik pada  $x = -2$  dan  $y = 3$
- Letakkan kursor di tampilan grafik pada  $x = 3$  dan  $y = 4$
- Pilih Parallel Line  pada bagian Construct
- Klik garis pertama yang dibuat dari titik  $A(-2,3)$  dan  $B(3,4)$


- Tarik kursor ke bawah atau ke atas (tergantung akan ditempatkan dimana garis sejajarnya)
- Klik sembarang tempat di tampilan grafik
- Klik Tools 
- Pilih Perpendicular Line  pada bagian Construct
- Klik garis pertama yang dibuat dari titik titik  $A(-2,3)$  dan  $B(3,4)$
- Tarik kursor ke kiri atau kanan (tergantung akan ditempatkan dimana garis tegak lurusnya)
- Klik sembarang tempat di tampilan grafik

Tampilan aljabar dan tampilan grafik dari langkah-langkah di atas dapat dilihat pada Gambar 2.5. Pada tampilan grafik, garis  $f$  adalah garis yang dibentuk dari dua titik, garis  $g$  adalah garis sejajar dengan garis  $f$ , dan garis  $h$  adalah garis yang tegak lurus dengan garis  $f$ .



**Gambar 2.5** Garis melalui dua titik, garis sejajar, dan tegak lurus

### c. Membuat Sudut



Untuk membuat sudut dapat dilakukan dengan klik menu Tools  pilih sudut yang akan dibuat pada bagian “Measure”. Terdapat dua pilihan untuk membuat sudut, yaitu Angle dan Angle with Given Size.



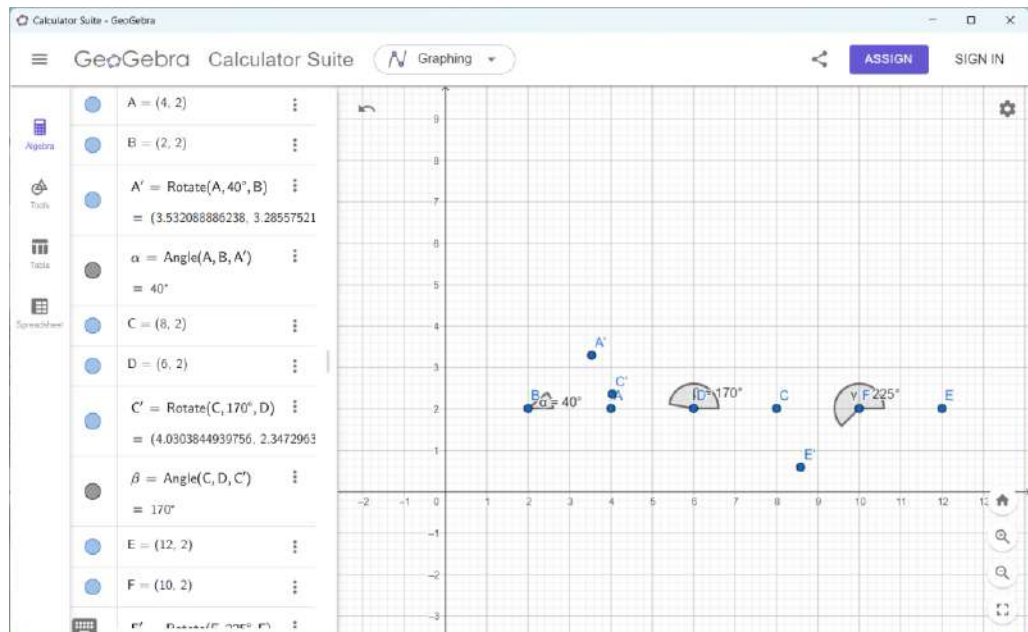
### Praktikum 1.3

Gambarlah sudut  $40^\circ$ ,  $170^\circ$ , dan  $255^\circ$ !

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.3:


- Klik Tools 
- Pilih Angle with Given Size  pada bagian Measure
- Klik sembarang tempat di tampilan grafik
- Klik sekali lagi sembarang tempat di tampilan grafik. Akan muncul sebuah titik yang akan menjadi verteks dari sudut yang akan dibuat
- Isi besar sudut  $40^\circ$
- Pilih “counterclockwise”
- Klik “OK”

Ulangi langkah-langkah di atas untuk menggambar sudut  $170^\circ$  dan  $255^\circ$ . Tampilan aljabar dan tampilan grafik dari langkah-langkah di atas dapat dilihat pada Gambar 2.6.



**Gambar 2.6** Sudut  $40^\circ$ ,  $170^\circ$ , dan  $255^\circ$

#### d. Membuat Bidang Datar




Untuk membuat bidang datar dapat dilakukan dengan klik menu Tools  pilih garis yang akan dibuat pada bagian "Lines". Terdapat beberapa pilihan, Segment, Line, Ray, dan Vector. Selain itu, terdapat bagian "Construct" dengan beberapa pilihan antara lain Perpendicular Line untuk membuat garis tegak lurus, Parallel Line untuk membuat garis sejajar, dll.

Bidang datar yang berupa segitiga, segiempat, dan segi- $n$  beraturan dapat dibuat dengan memilih Polygon atau Regular Polygon di bagian Polygons. Sedangkan bidang datar yang berupa lingkaran dapat dibuat dengan memilih Circle with Center atau Circle: Center & Radius pada bagian Circles. Sebagai tambahan, untuk menggambar elips dapat menggunakan Ellipse yang ada pada bagian Conics.

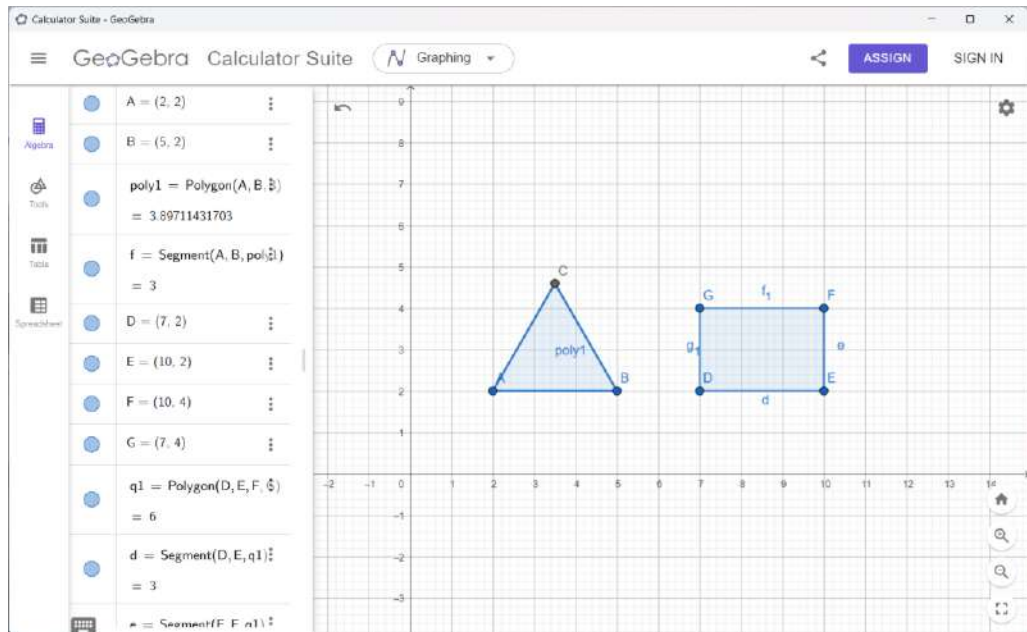
##### Praktikum 1.4

Gambarlah segitiga sama sisi dengan panjang sisinya 3 satuan panjang dan persegi panjang dengan panjang 3 dan lebar 2 satuan panjang!

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.4:

- Klik Tools 
- Pilih Regular Polygon  pada bagian Polygons
- Klik sembarang tempat di tampilan grafik
- Klik sekali lagi di tampilan grafik dengan jarak 3 satuan panjang dari titik pertama.
- Isi banyaknya Vertices (sisi) dengan 3
- Klik "OK"
- Pilih Polygon  pada bagian Polygons
- Klik sembarang tempat di tampilan grafik
- Klik lagi di tampilan grafik dengan jarak 3 satuan panjang dari titik pertama
- Klik lagi di tampilan grafik dengan jarak 2 satuan panjang dari titik kedua dan posisi dengan dua sudut pertama membentuk sudut 90 derajat
- Klik sekali lagi di titik pertama



Tampilan aljabar dan tampilan grafik dari langkah-langkah di atas dapat dilihat pada Gambar 2.7.



**Gambar 2.7** Segitiga sama sisi dan persegi panjang

### e. Membuat Bangun Ruang



Sebelum membuat bangun ruang pastikan memilih 3D Calculator terlebih dahulu di bagian atas, di sebelah tulisan Calculator Suite, hingga tampilan grafik berubah menjadi 3D. Selanjutnya, untuk membuat bangun ruang dapat

dilakukan dengan klik menu Tools . Terdapat beberapa pilihan yang akan muncul di bagian Solids. Jika tidak muncul bagian Solids, klik More di sebelah kiri bawah. Beberapa pilihan di bagian Solid yaitu Pyramid, Prism, Tetrahedron, Cube, Sphere: Center & Point, Sphere: Center, Cone, dan Cylinder. Untuk membuat jaring-jaring bangun ruang, terdapat pilihan Net di bagian Solids. Untuk memodifikasi tampilan grafik 3D agar muncul garis pada bidang  $x$  sehingga memudahkan dalam menentukan posisi titik ketika akan membuat bangun ruang, klik setting  di sebelah kanan atas dan pilih Show Grid.


#### Praktikum 1.5

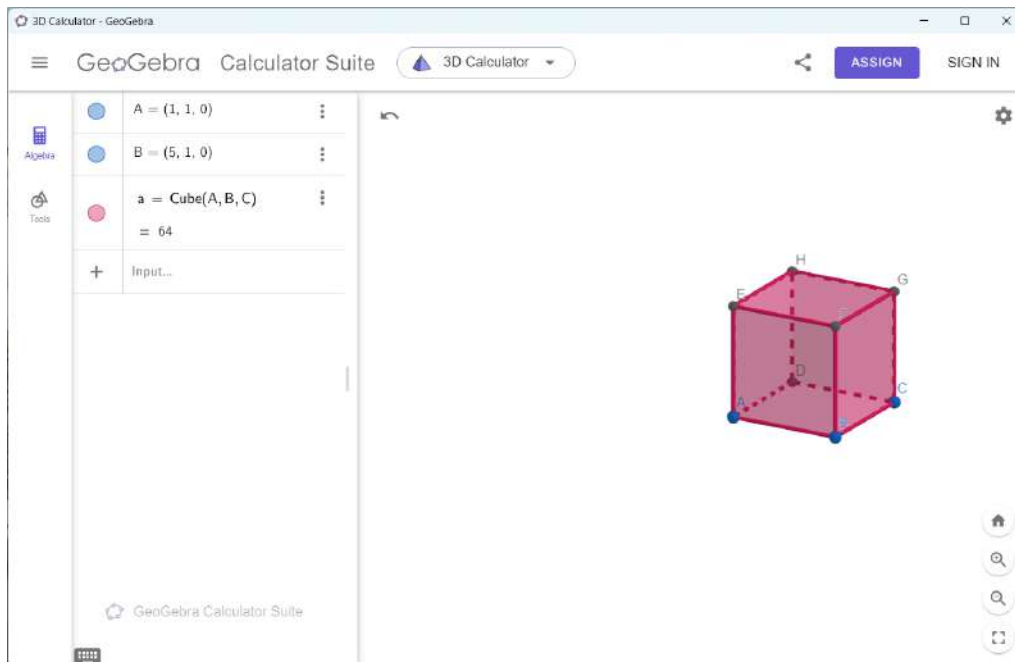
Gambarlah kubus dengan panjang sisi 4 satuan panjang dan jaring-jaringnya!

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.4:

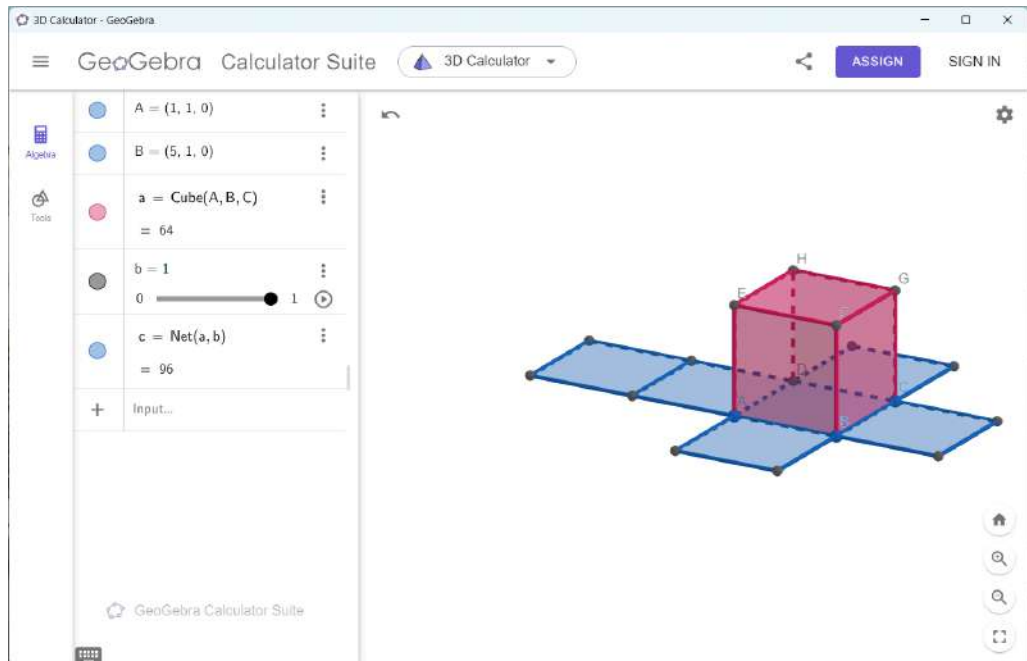
- Klik Tools 
- Pilih Cube  pada bagian Basic Tools atau Solids

- Klik sembarang tempat di tampilan grafik 3D
- Klik sekali lagi di tampilan grafik 3D dengan jarak 4 satuan panjang dari titik pertama
- Pilih "Net"
- Klik kubus yang telah dibuat sebelumnya

Untuk menghilangkan sumbu  $x$  dan  $y$ , bidang  $x$ , dan garis pada bidang  $x$ , klik setting  di sebelah kanan atas dan hapus centang pada Show Axes, Show Plane, dan Show Grid. Tampilan aljabar dan tampilan grafik dari langkah-langkah di atas dapat dilihat pada Gambar 2.8 dan Gambar 2.9.



**Gambar 2.8** Bangun ruang kubus



Gambar 2.9 Kubus dan jaring-jaringnya

## B. PRAKTIKUM VISUALISASI FUNGSI

### 1. Operasi Fungsi, Fungsi Komposisi, dan Fungsi Invers

Pada BMP MATA4110 Kalkulus I Modul 2 telah dibahas materi mengenai fungsi dan grafiknya. Sebelum menggunakan *software* GeoGebra untuk menentukan operasi fungsi, komposisi fungsi, dan fungsi invers diharapkan Anda telah mempelajari materi tersebut.

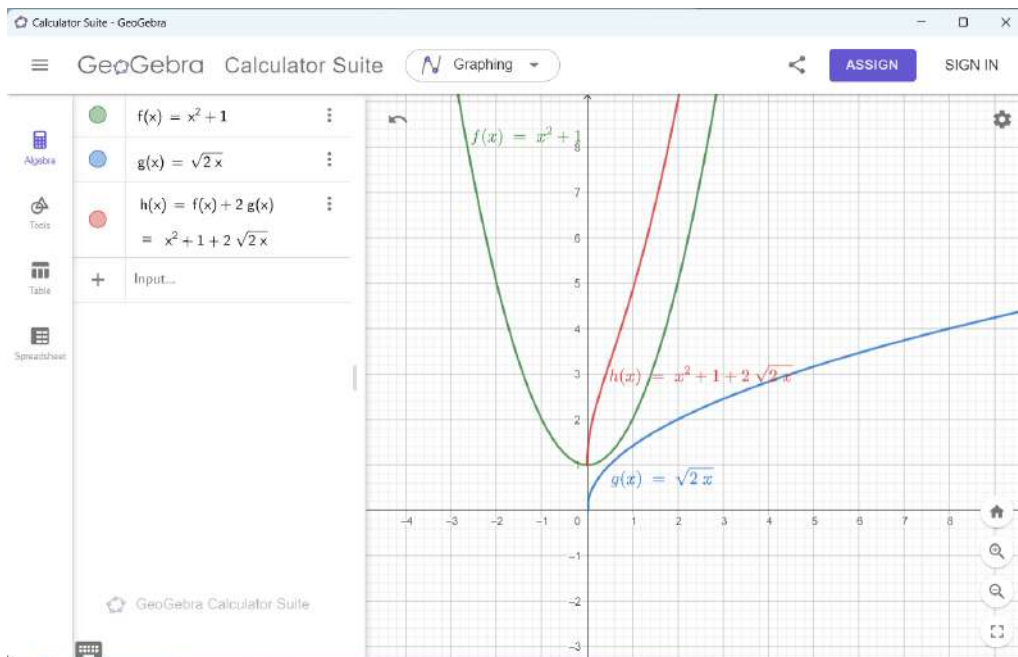
Secara umum, untuk menentukan operasi fungsi seperti penjumlahan, pengurangan, perkalian, pembagian, dan perpangkatan dapat diketik secara langsung di kolom input.

#### Praktikum 1.6

Diketahui  $f(x) = x^2 + 1$  dan  $g(x) = \sqrt{2x}$ . Tentukan  $f(x) + 2g(x)$ ,  $g(x) - 3f(x)$ ,  $\frac{1}{2}f(x)g(x)$ ,  $\frac{g(x)}{f(x)}$ ,  $f^2(x) + \frac{1}{g(x)}$  serta gambar grafik dari masing-masing operasi fungsi tersebut! **Apa perlu dituliskan  $x \geq 0$ ??**

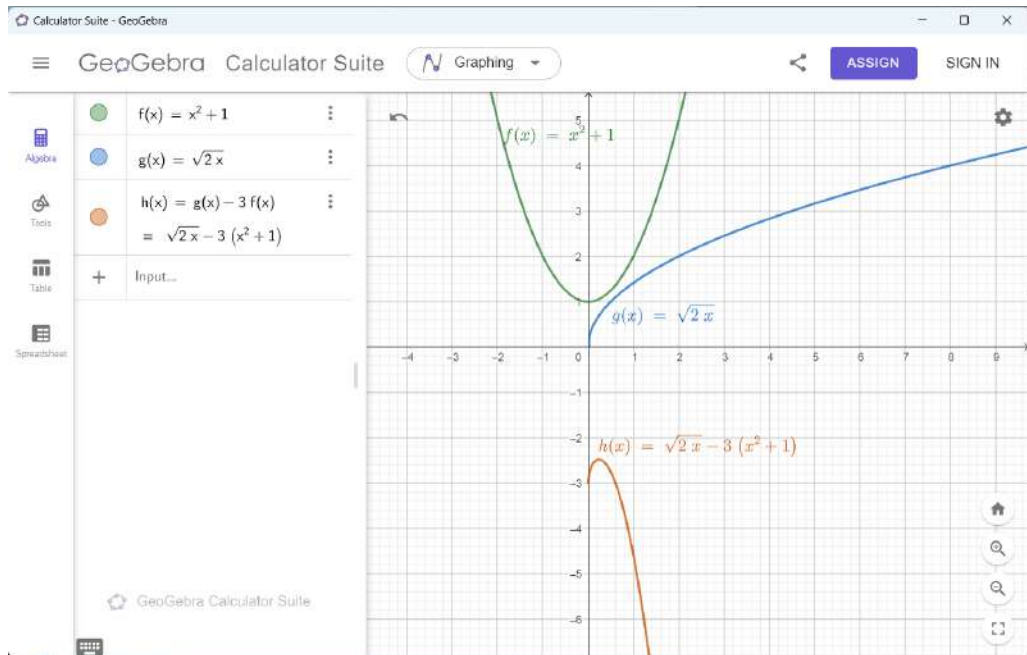
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.6:

- Ketikkan  $x^2 + 1$  dan  $\sqrt{2x}$  di kolom input
- Ketikkan  $f + 2g$  di kolom input sehingga diperoleh  $h(x) = x^2 + 1 + 2\sqrt{2x}$ .



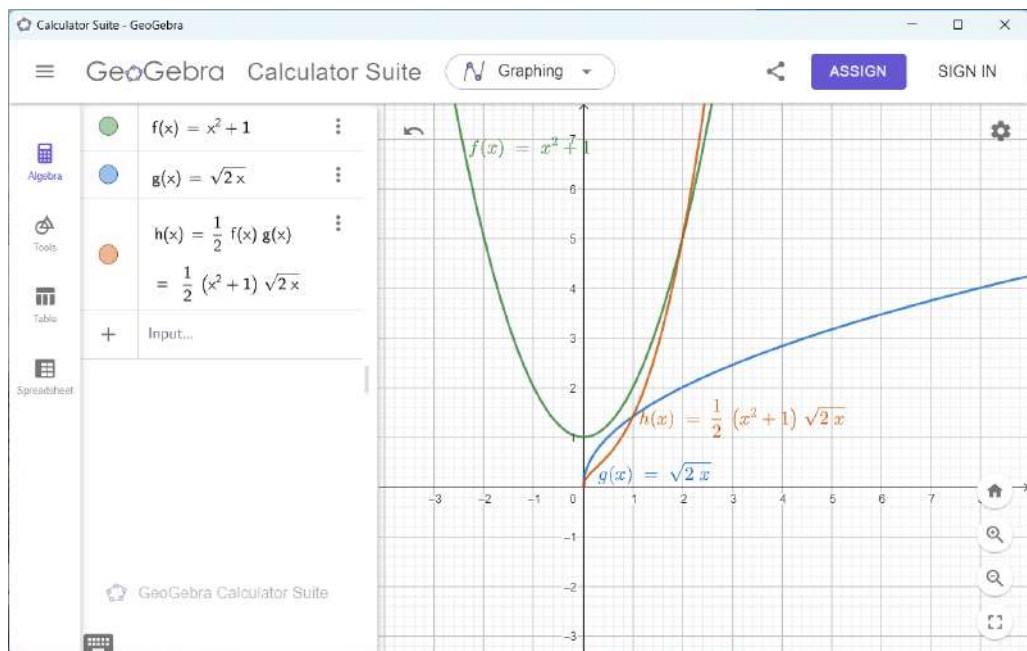
**Gambar 2.10** Grafik  $f(x)$ ,  $g(x)$ , dan  $f(x) + 2g(x)$ .

- Untuk mencari operasi fungsi selanjutnya, setelah disimpan, hapus  $h(x)$  dengan klik titik tiga di sebelah kanan  $h(x)$  pada tampilan aljabar.
- Ketikkan  $g - 3f$  di kolom input sehingga diperoleh  $h(x) = \sqrt{2}x - 3(x^2 + 1)$ .



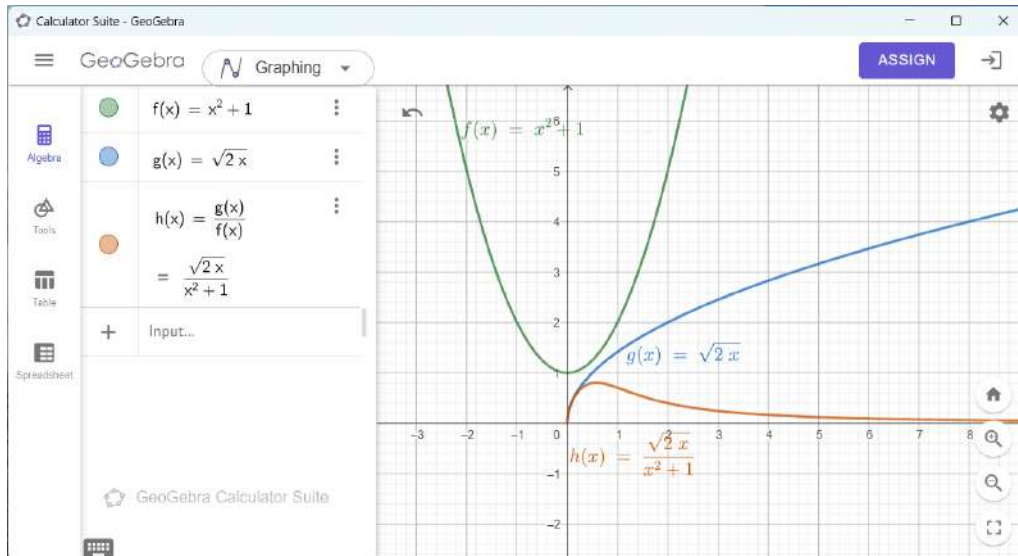
**Gambar 2.11** Grafik  $f(x)$ ,  $g(x)$ , dan  $g(x) - 3f(x)$ .

- Untuk mencari operasi fungsi selanjutnya, setelah disimpan, hapus  $h(x)$  dengan klik titik tiga di sebelah kanan  $h(x)$  pada tampilan aljabar.
- Ketikkan  $\frac{1}{2}fg$  di kolom input sehingga diperoleh  $h(x) = \frac{1}{2}(x^2 + 1)\sqrt{2x}$ .



**Gambar 2.12** Grafik  $f(x)$ ,  $g(x)$ , dan  $\frac{1}{2}f(x)g(x)$ .

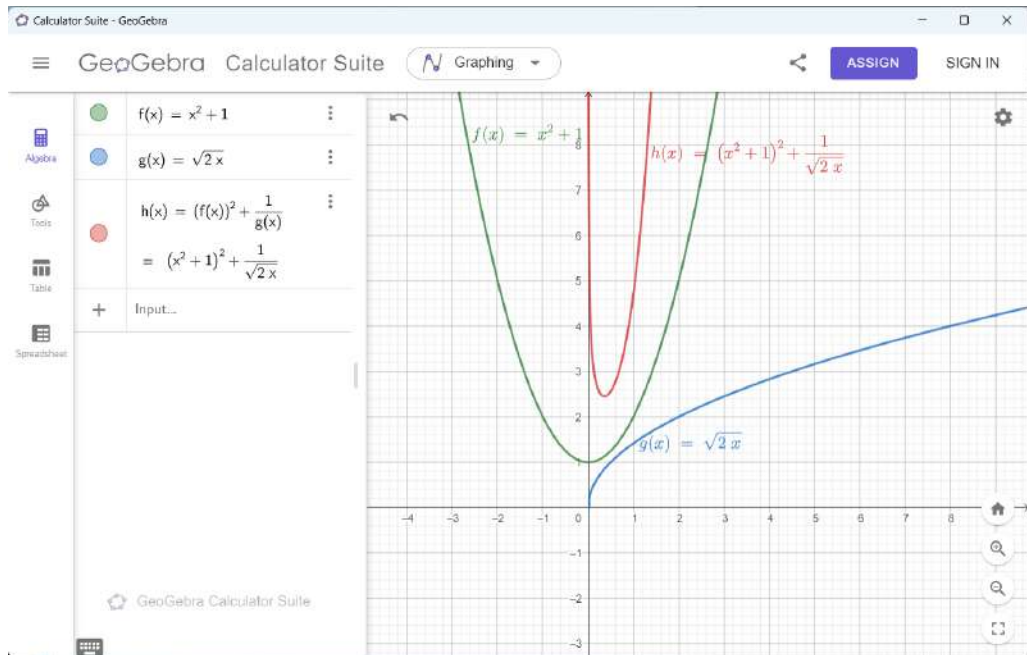
- Untuk mencari operasi fungsi selanjutnya, setelah disimpan, hapus  $h(x)$  dengan klik titik tiga di sebelah kanan  $h(x)$  pada tampilan aljabar.
- Ketikkan  $g/f$  di kolom input sehingga diperoleh  $\frac{\sqrt{2x}}{x^2+1}$



**Gambar 2.13** Grafik  $f(x)$ ,  $g(x)$ , dan  $\frac{g(x)}{f(x)}$ .

- Untuk mencari operasi fungsi selanjutnya, setelah disimpan, hapus  $h(x)$  dengan klik titik tiga di sebelah kanan  $h(x)$  pada tampilan aljabar.
- Ketikkan  $f^2 + 1/g$  di kolom input sehingga diperoleh  $h(x) = (x^2 + 1)^2 + \frac{1}{\sqrt{2x}}$ .





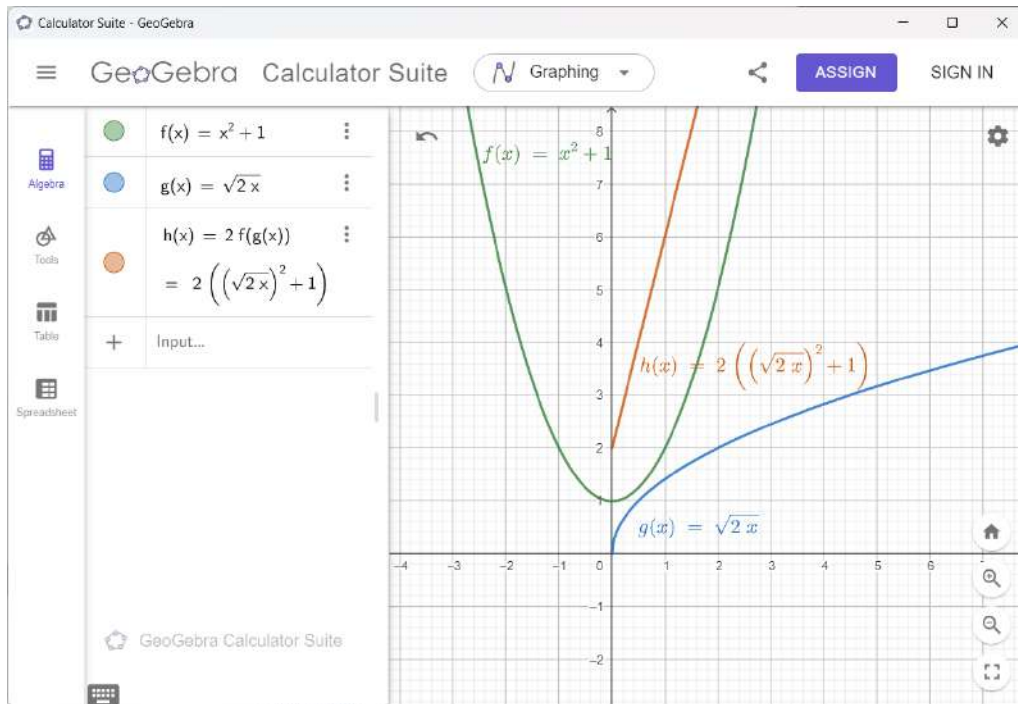
**Gambar 2.14** Grafik  $f(x)$ ,  $g(x)$ , dan  $f^2(x) + \frac{1}{g(x)}$ .

### Praktikum 1.7

Diketahui  $f(x) = x^2 + 1$  dan  $g(x) = \sqrt{2x}$ . Tentukan  $(2f \circ g)(x)$  dan  $(g \circ 3f)(x)$ . **Apa perlu dituliskan  $x \geq 0$ ??**

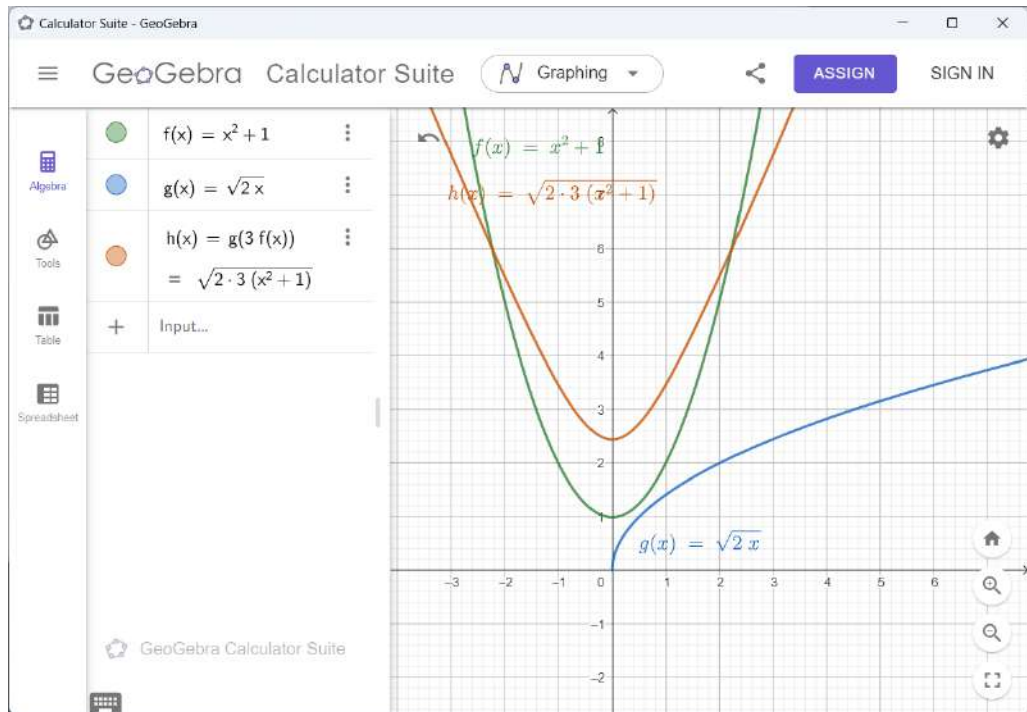
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.7:

- Ketikkan  $x^2 + 1$  dan  $\sqrt{2x}$  di kolom input
- Ketikkan  $2f(g(x))$  di kolom input sehingga diperoleh  $h(x) = 2((\sqrt{2x})^2 + 1)$  atau  $h(x) = 4x + 2$  dengan domain  $x \geq 0$ .



**Gambar 2.15** Grafik  $f(x)$ ,  $g(x)$ , dan  $(2f \circ g)(x)$ .

- Untuk mencari operasi fungsi selanjutnya, setelah disimpan, hapus  $h(x)$  dengan klik titik tiga di sebelah kanan  $h(x)$  pada tampilan aljabar.
- Ketikkan  $g(3f(x))$  di kolom input sehingga diperoleh  $h(x) = \sqrt{2 \cdot 3(x^2 + 1)}$  atau  $h(x) = \sqrt{6(x^2 + 1)}$ .

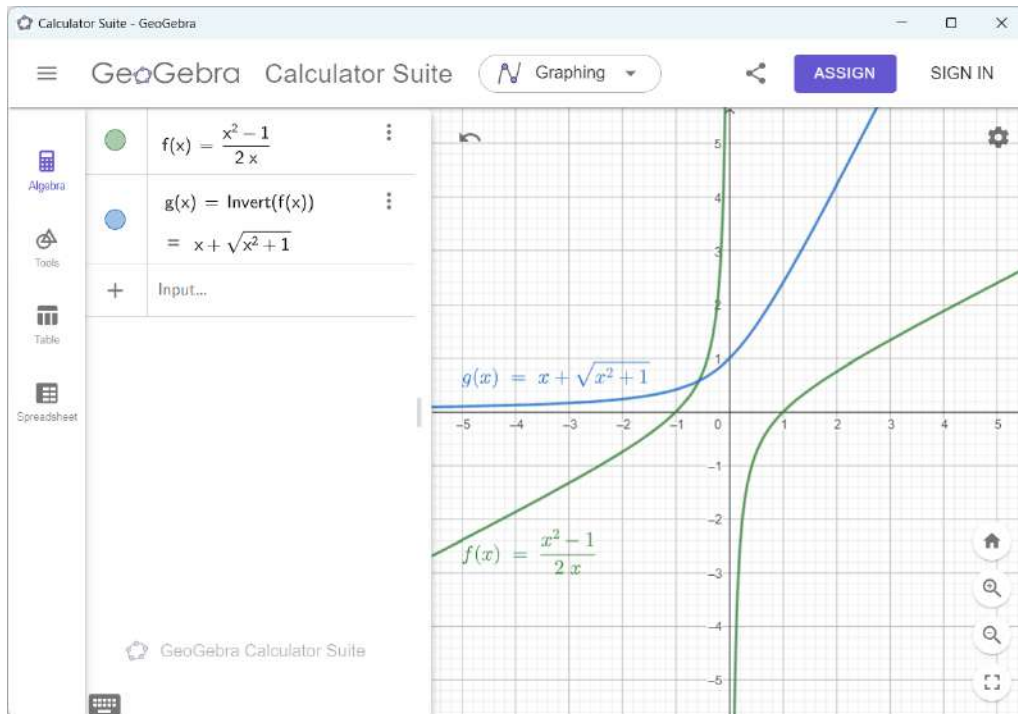


**Gambar 2.16** Grafik  $f(x)$ ,  $g(x)$ , dan  $(g \circ 3f)(x)$ .

### Praktikum 1.8

Diketahui  $f(x) = \frac{x^2-1}{2x}$ . Tentukan fungsi inversnya dan gambarlah grafiknya!

Untuk menentukan fungsi invers dan gambar grafik Praktikum 1.8 dapat dilakukan dengan mengetikkan  $\frac{x^2-1}{2x}$  di kolom input kemudian mengetikkan  $\text{Invert}(f(x))$  sehingga diperoleh  $g(x) = x + \sqrt{x^2 + 1}$ . Sebagai tambahan, dalam menentukan fungsi invers dapat juga dengan mengetikkan  $\text{Invert}\left(\frac{x^2-1}{2x}\right)$  tanpa menjabarkan fungsinya terlebih dahulu seperti cara pertama.



**Gambar 2.17** Grafik  $f(x)$  dan invers  $f(x)$  atau  $f^{-1}(x)$ .

## 2. Grafik Fungsi

Pada BMP MATA4110 Kalkulus I Modul 3 telah dibahas mengenai fungsi dan grafiknya. Sebelum menggunakan *software* GeoGebra untuk menggambar grafik fungsi diharapkan Anda telah mempelajari materi tersebut.

Secara umum, untuk menggambar grafik fungsi dapat dilakukan dengan cara mengetikkan fungsinya secara langsung di kolom input atau dengan klik kecuali fungsi dengan domain tertentu, fungsi sepotong-sepotong, dan fungsi parameter. Untuk menggambar fungsi dengan domain tertentu dapat menggunakan sintaksis sebagai berikut

$lf(\text{domain fungsi}, \text{fungsi})$

Sedangkan untuk menggambar grafik fungsi sepotong-sepotong dengan 2 fungsi dapat menggunakan sintaksis sebagai berikut

$lf(\text{domain fungsi 1}, \text{fungsi 1}, \text{if}(\text{domain fungsi 2}, \text{fungsi 2}))$

dan untuk menggambar grafik fungsi sepotong-sepotong dengan  $n$  fungsi dapat menggunakan sintaksis sebagai berikut

$lf(\text{domain fungsi 1}, \text{fungsi 1}, \text{if}(\text{domain fungsi 2}, \text{fungsi 2}, (\dots, \dots, \text{if}(\text{domain fungsi } n, \text{fungsi } n))))$

Sebagai tambahan, untuk membuat grafik fungsi parameter menggunakan sintaksi berikut

*Curve(fungsi parameter 1, fungsi parameter 2, parameter, batas bawah, batas atas)*

**Contoh:**

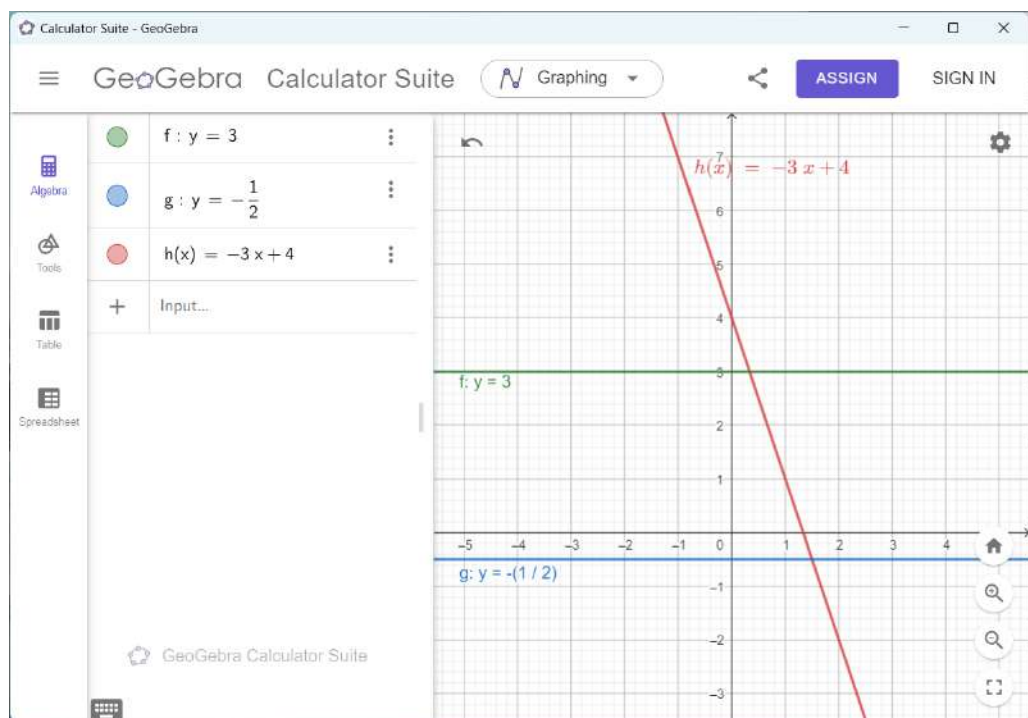
### a. Fungsi Aljabar

Fungsi aljabar meliputi fungsi rasional dan irasional. Fungsi rasional terdiri atas fungsi konstan, polinom, dan pecahan.

#### Praktikum 1.9

Gambarlah grafik fungsi  $f(x) = 3$ ,  $f(x) = -\frac{1}{2}$ , dan  $f(x) = -3x + 4$ .

Untuk menggambar grafik fungsi pada Praktikum 1.9, ketikkan  $y = 3$ ,  $y = -\frac{1}{2}$ , dan  $-3x + 4$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.18.



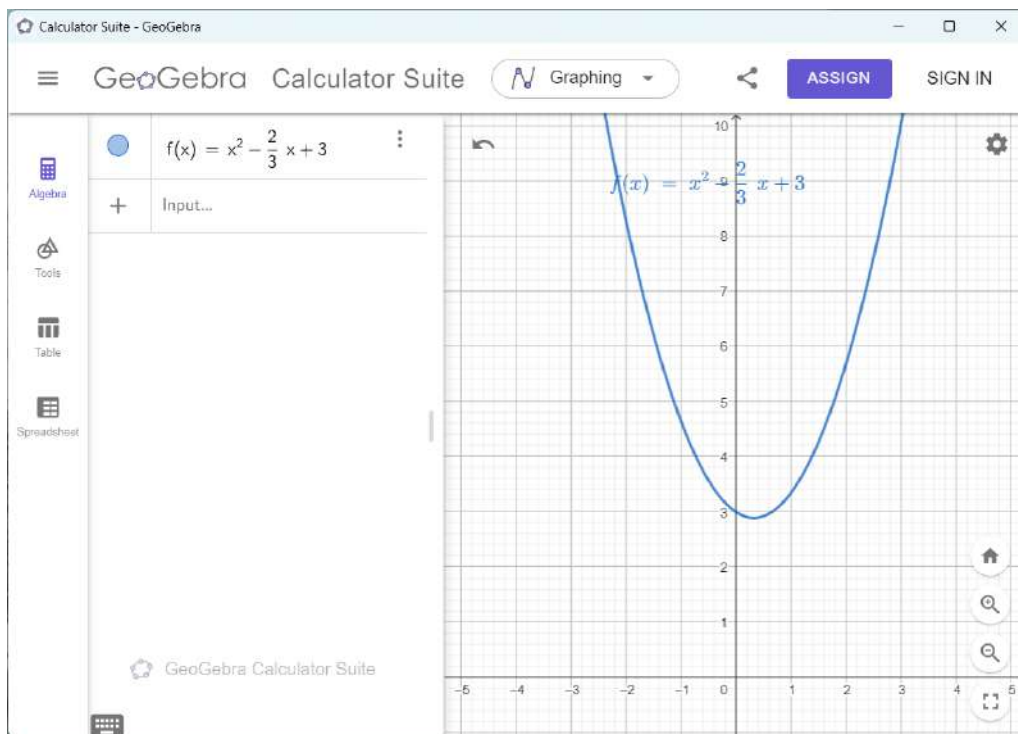
**Gambar 2.18** Grafik  $f(x) = 3$ ,  $f(x) = -\frac{1}{2}$ , dan  $f(x) = -3x + 4$ .

### Praktikum 1.10

Gambarlah grafik fungsi  $f(x) = x^2 - \frac{2}{3}x + 3$  dan  $f(x) = 2x^3 - 5x^2 + x + 1$ .

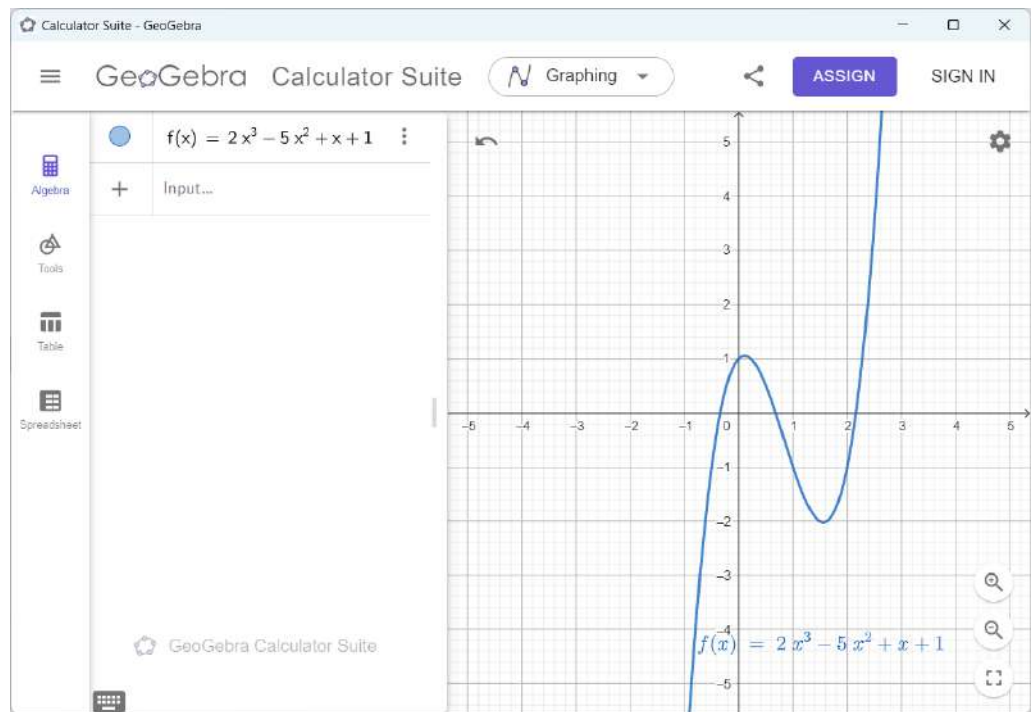
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.10:

- Ketikkan  $x^2 - \frac{2}{3}x + 3$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.19.



**Gambar 2.19** Grafik  $f(x) = x^2 - \frac{2}{3}x + 3$ .

- Untuk menggambar grafik fungsi selanjutnya, setelah disimpan, hapus  $f(x)$  dengan klik titik tiga di sebelah kanan  $f(x)$  pada tampilan aljabar atau klik "Clear All" pada bagian "Main Menu".
- Ketikkan  $2x^3 - 5x^2 + x + 1$  di kolom input diperoleh grafik seperti pada Gambar 2.20.



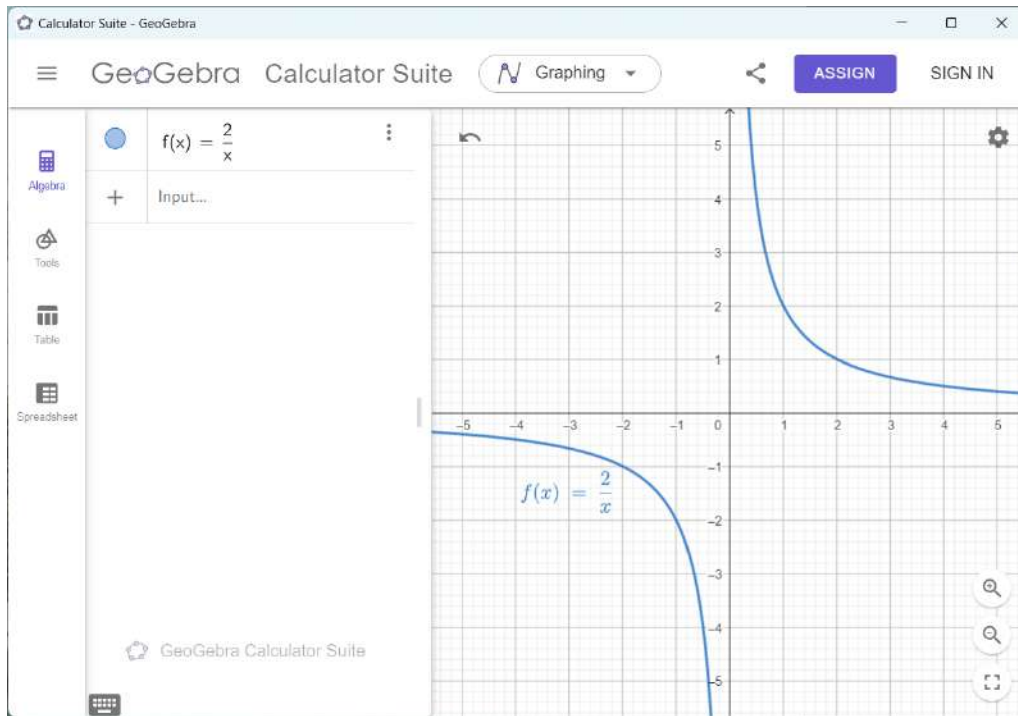
**Gambar 2.20** Grafik  $f(x) = 2x^3 - 5x^2 + x + 1$ .

### Praktikum 1.11

Gambarlah grafik fungsi  $f(x) = \frac{2}{x}$  dan  $f(x) = \frac{1}{3x^3}$ .

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.11:

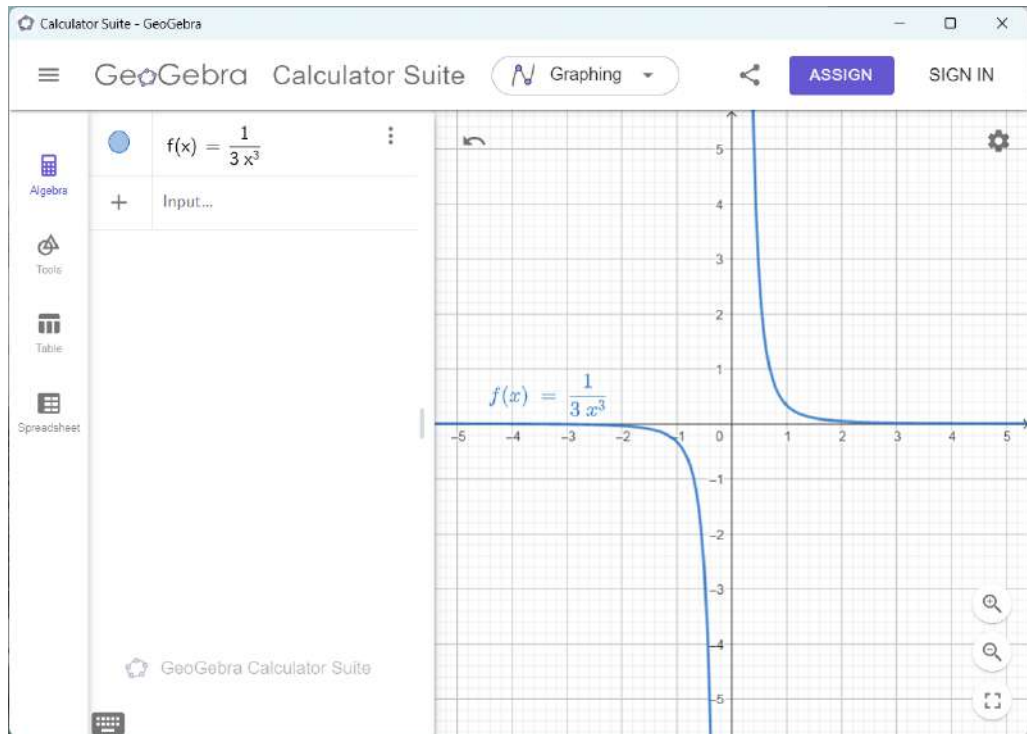
- Ketikkan  $\frac{2}{x}$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.21.



**Gambar 2.21** Grafik  $f(x) = \frac{2}{x}$ .

- Ketikkan  $\frac{1}{3x^3}$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.22.





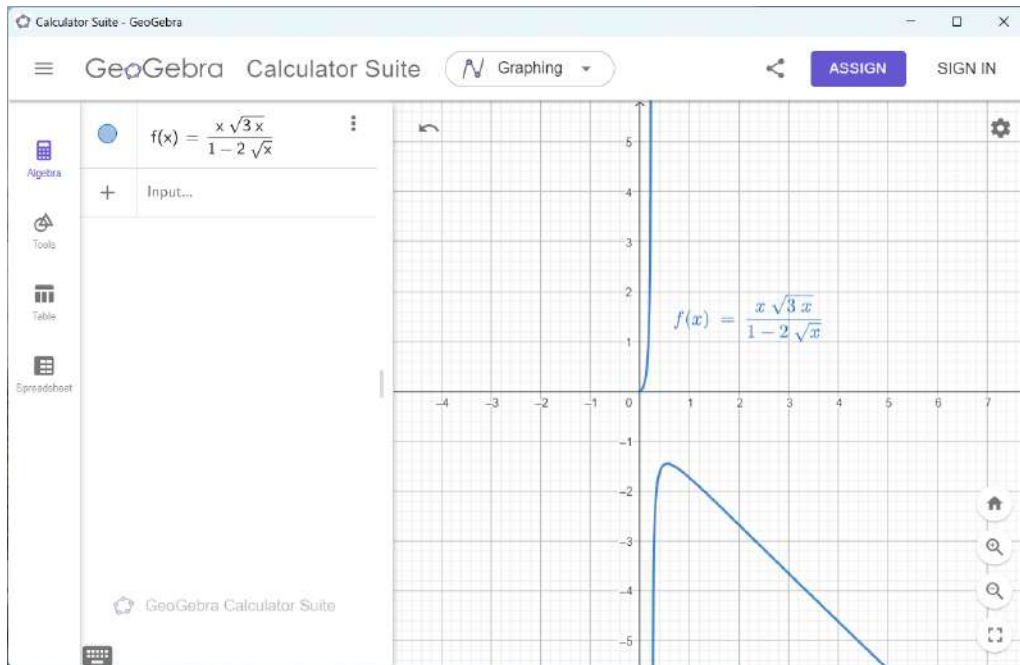
**Gambar 2.22** Grafik  $f(x) = \frac{1}{3x^3}$ .

### Praktikum 1.12

Gambarlah grafik fungsi  $f(x) = \frac{x\sqrt{3x}}{1-2\sqrt{x}}$  dan  $f(x) = \frac{\sqrt{2x}+\sqrt{x}}{2\sqrt{x}-1}$ .

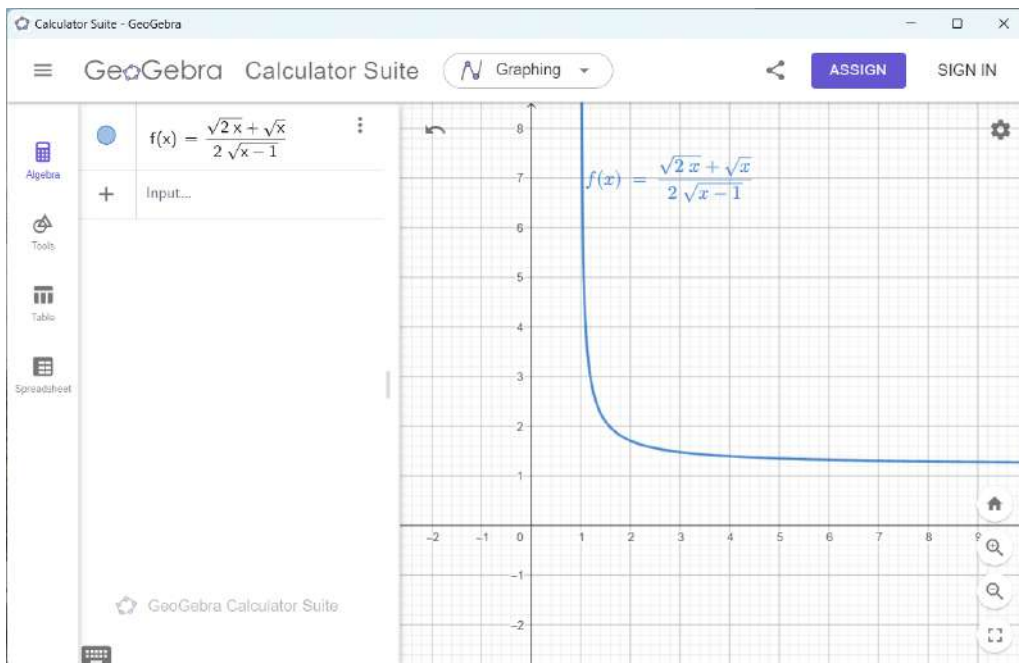
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.12:

- Ketikkan  $\frac{x\sqrt{3x}}{1-2\sqrt{x}}$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.23.



**Gambar 2.23** Grafik  $f(x) = \frac{x\sqrt{3x}}{1-2\sqrt{x}}$ .

- Ketikkan  $\frac{\sqrt{2x}+\sqrt{x}}{2\sqrt{x}-1}$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.24.

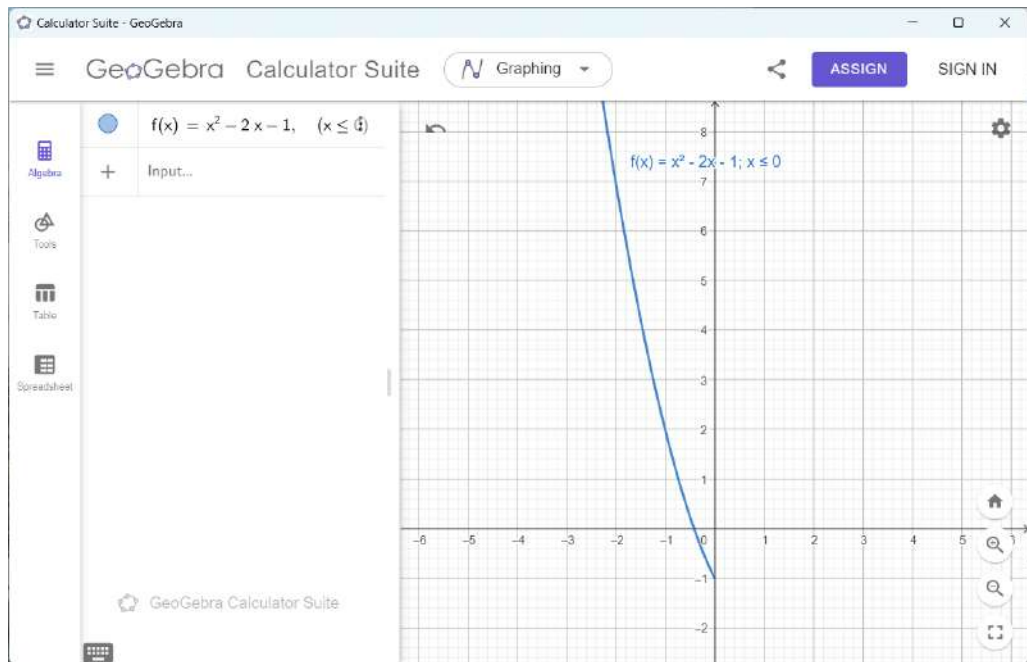


**Gambar 2.24** Grafik  $f(x) = \frac{\sqrt{2x} + \sqrt{x}}{2\sqrt{x-1}}$ .

*Praktikum 1.13*

Gambarlah grafik fungsi  $f(x) = x^2 - 2x - 1$ , dengan  $x \leq 0$ .

Praktikum 1.13 dapat diselesaikan dengan langsung mengetikkan  $y=lf(x\leq 0,x^2-2x-1)$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.25.

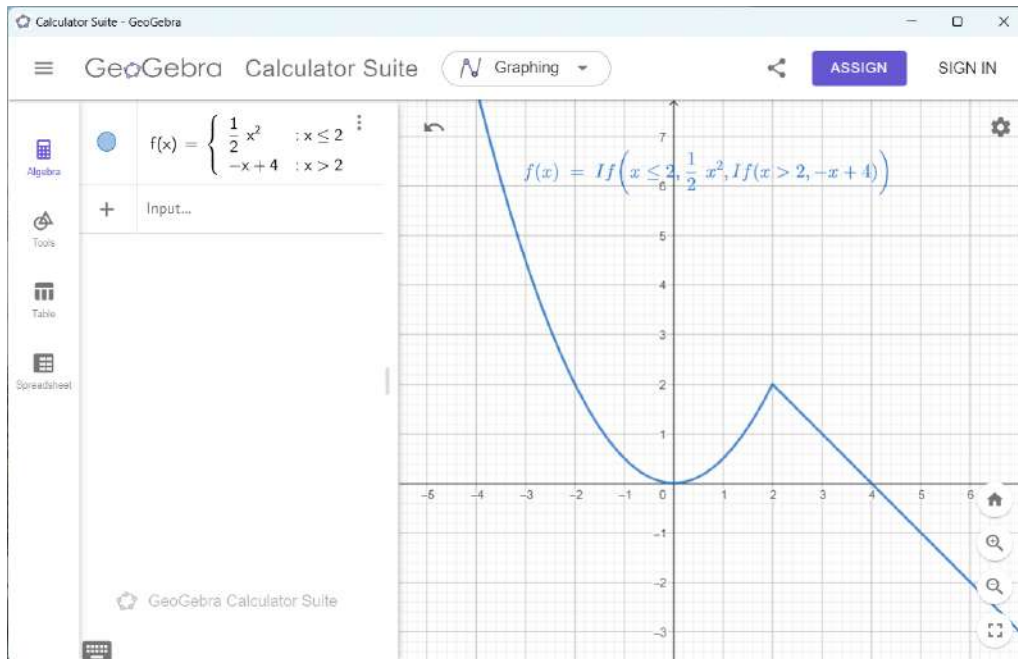


**Gambar 2.25** Grafik  $f(x) = x^2 - 2x - 1$ , dengan  $x \leq 0$ .

*Praktikum 1.14*

Gambarlah grafik fungsi  $f(x) = \begin{cases} \frac{1}{2}x^2, & x \leq 2 \\ -x + 4, & x > 2 \end{cases}$

Praktikum 1.14 dapat diselesaikan dengan langsung mengetikkan  $y=lf(x\leq 2,(1/2)*x^2,if(x>2,-x+4))$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.26.



**Gambar 2.26** Grafik  $f(x) = \begin{cases} \frac{1}{2}x^2, & x \leq 2 \\ -x + 4, & x > 2 \end{cases}$ .

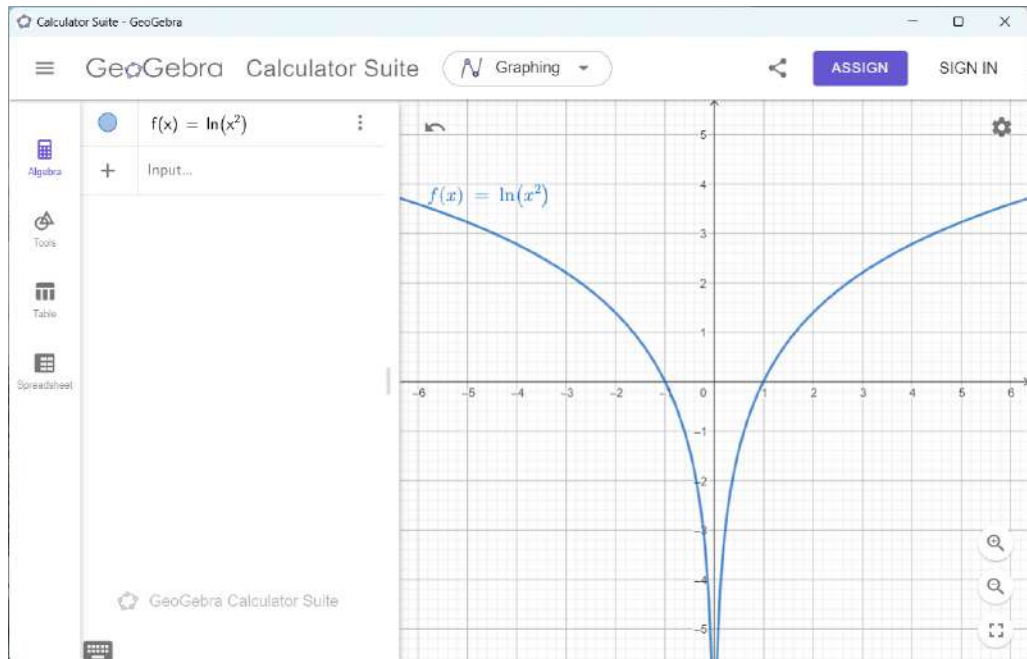
**b. Fungsi Transenden:** fungsi logaritma, Fungsi eksponensial, Fungsi trigonometri, Fungsi siklometri, Fungsi hiperbolik, Fungsi parameter.

### Praktikum 1.15

Gambarlah grafik fungsi  $f(x) = \ln x^2$  dan  $f(x) = e^{2x}$

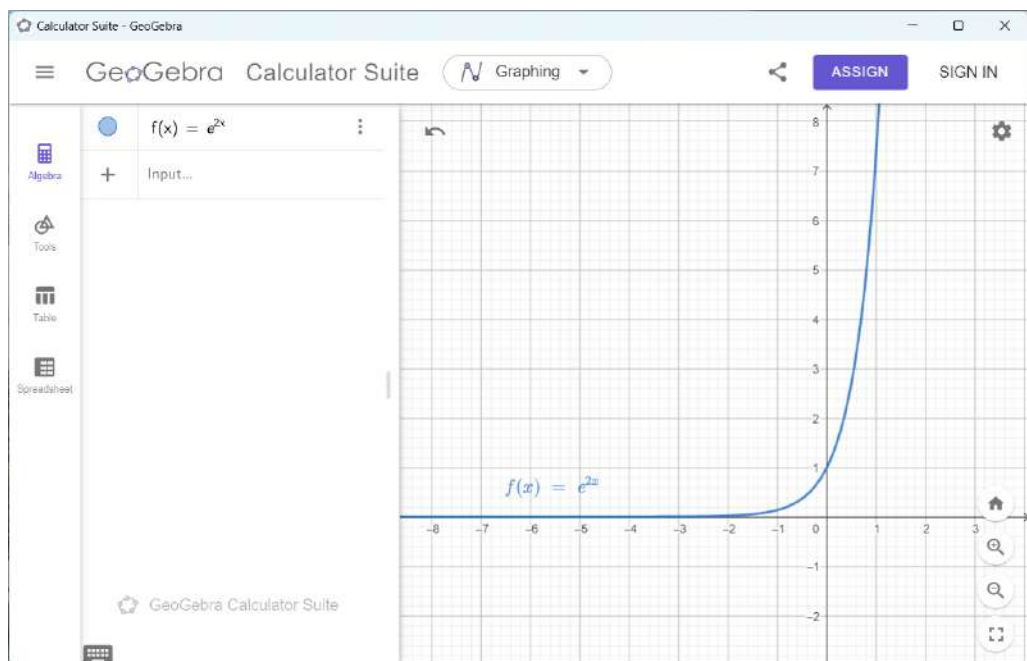
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.15:

- Ketikkan  $\ln x^2$  di kolom input sehingga sehingga diperoleh grafik seperti pada Gambar 2.27.



Gambar 2.27 Grafik  $f(x) = \ln x^2$ .

- Ketikkan  $e^{2x}$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.28.



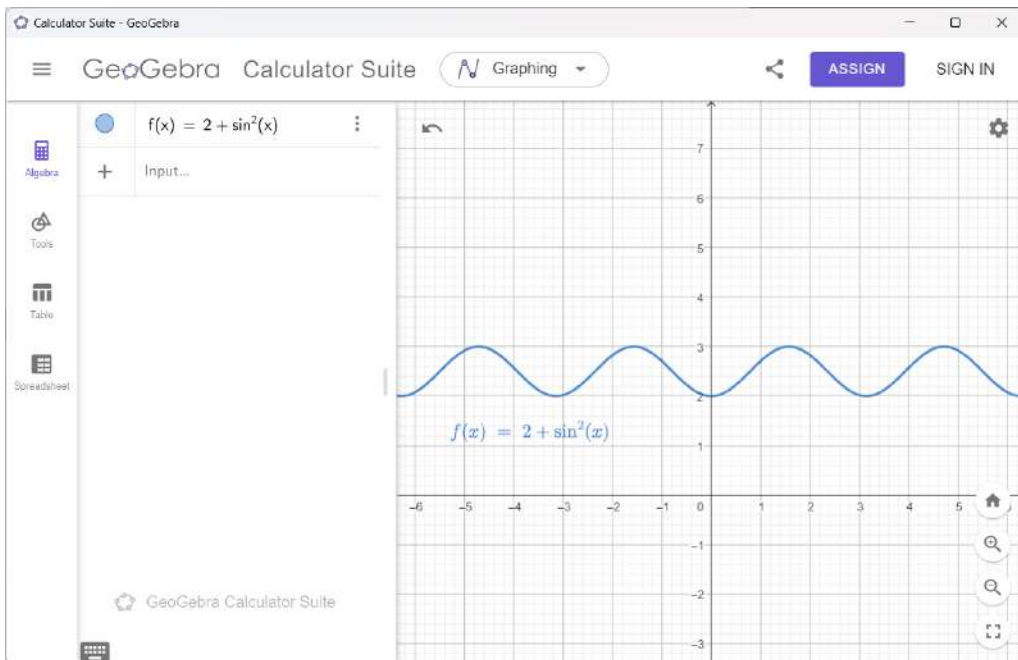
Gambar 2.28 Grafik  $f(x) = e^{2x}$ .

**Praktikum 1.16**

Gambarlah grafik fungsi  $f(x) = 2 + \sin^2 x$ ,  $f(x) = \frac{1}{2} + 2 \cos x$ , dan  $f(x) = \tan(2x + 1)$

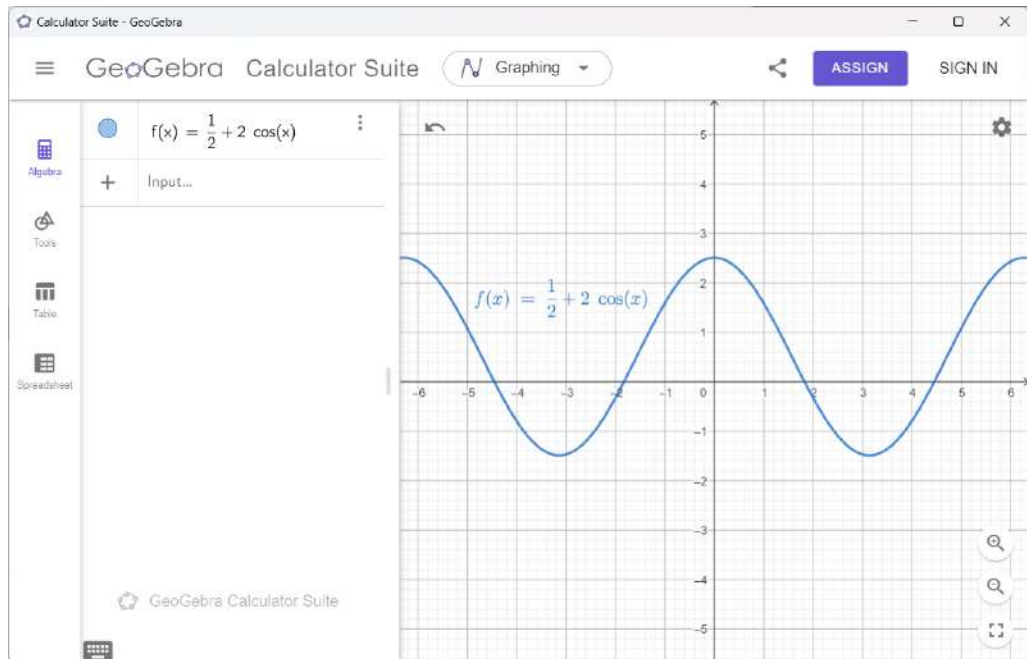
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.16:

- Ketikkan  $2 + \sin^2 x$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.29.



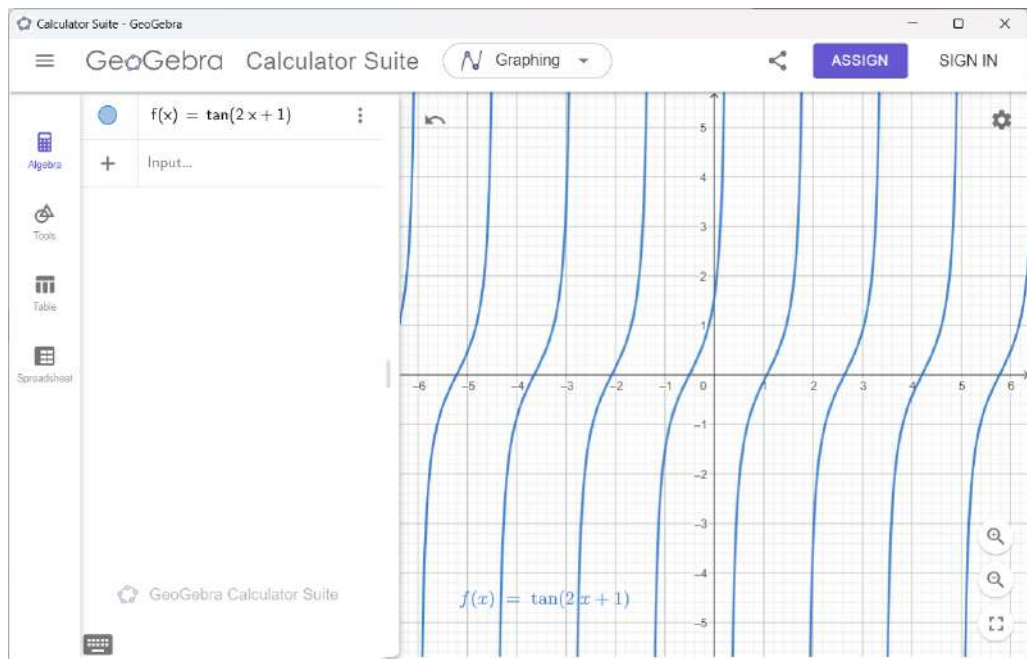
**Gambar 2.29** Grafik  $f(x) = 2 + \sin^2 x$ .

- Ketikkan  $\frac{1}{2} + 2 \cos x$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.30.



**Gambar 2.30** Grafik  $f(x) = \frac{1}{2} + 2 \cos x$ .

- Ketikkan  $\tan(2x + 1)$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.31.



**Gambar 2.31** Grafik  $f(x) = \tan(2x + 1)$ .

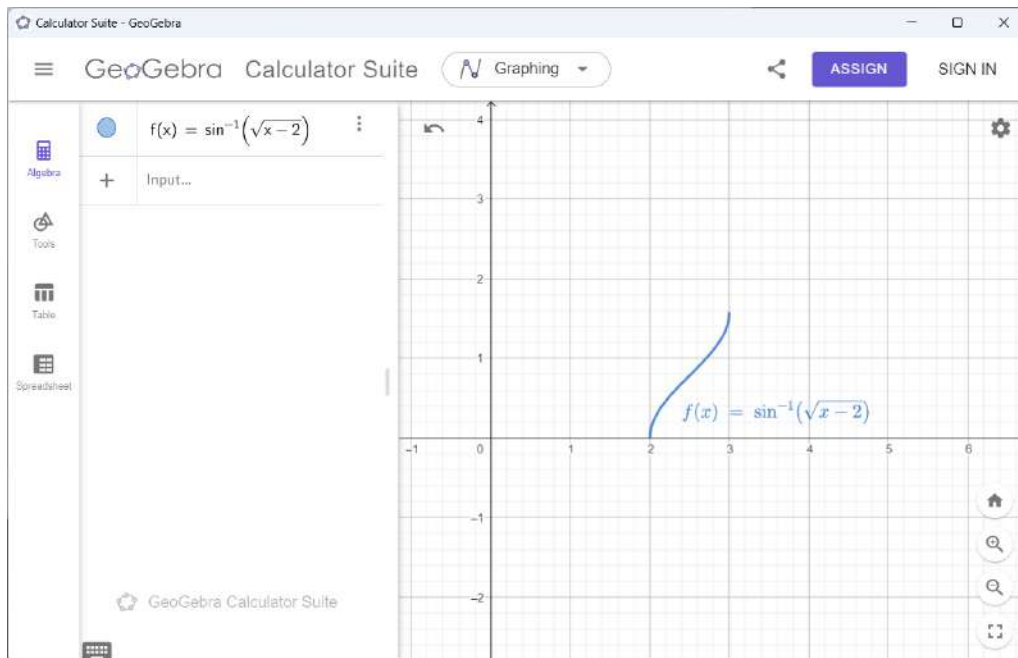


**Praktikum 1.17**

Gambarlah grafik fungsi siklometri (invers trigonometri)  $f(x) = \sin^{-1}\sqrt{x-2}$ ,  $f(x) = \cos^{-1}\left(\frac{1}{2}x - \frac{1}{4}\right)$ , dan  $f(x) = -\frac{1}{2} + \frac{1}{2}\tan^{-1}x$ .

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.17:

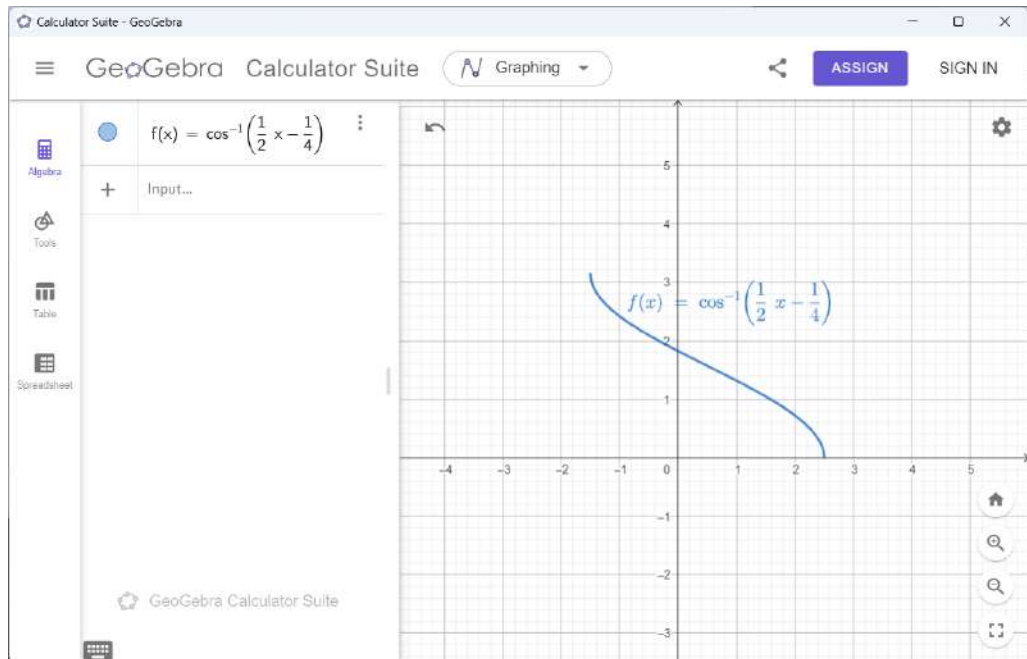
- Ketikkan  $\sin^{-1}\sqrt{x-2}$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.32.



**Gambar 2.32** Grafik  $f(x) = \sin^{-1}\sqrt{x-2}$ .

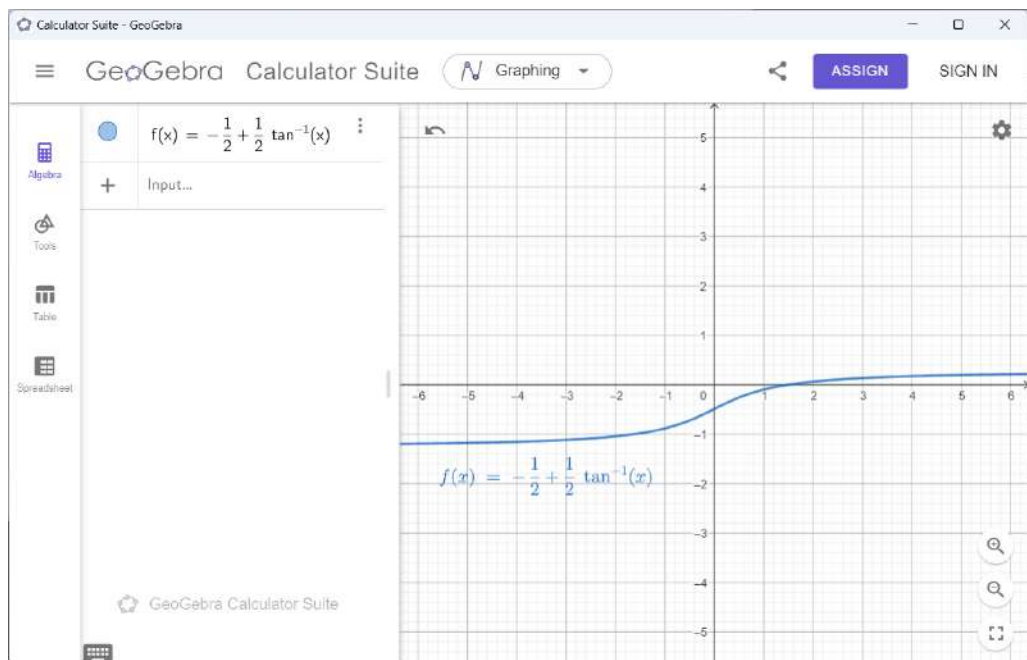
- Ketikkan  $\cos^{-1}\left(\frac{1}{2}x - \frac{1}{4}\right)$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.33.





**Gambar 2.33** Grafik  $f(x) = \cos^{-1}\left(\frac{1}{2}x - \frac{1}{4}\right)$ .

- Ketikkan  $-\frac{1}{2} + \frac{1}{2}\tan^{-1}x$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.34.



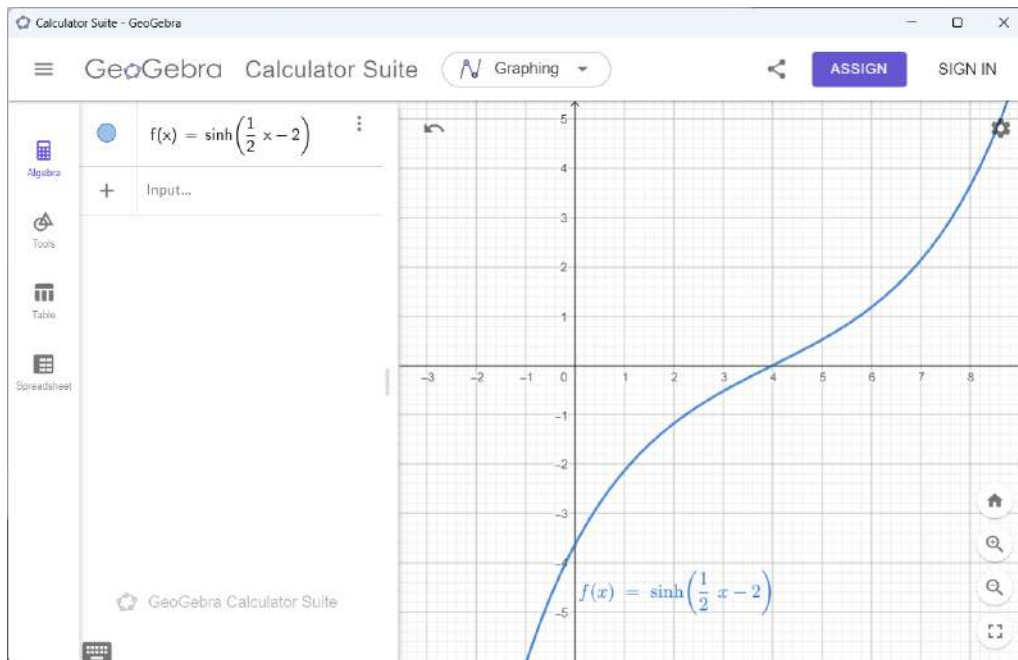
**Gambar 2.34** Grafik  $f(x) = -\frac{1}{2} + \frac{1}{2} \tan^{-1} x$ .

*Praktikum 1.18*

Gambarlah grafik fungsi hiperbolik  $f(x) = \sinh\left(\frac{1}{2}x - 2\right)$ ,  $f(x) = 2\cosh^2 x$ , dan  $f(x) = \frac{1}{2} + \tanh x^2$ .

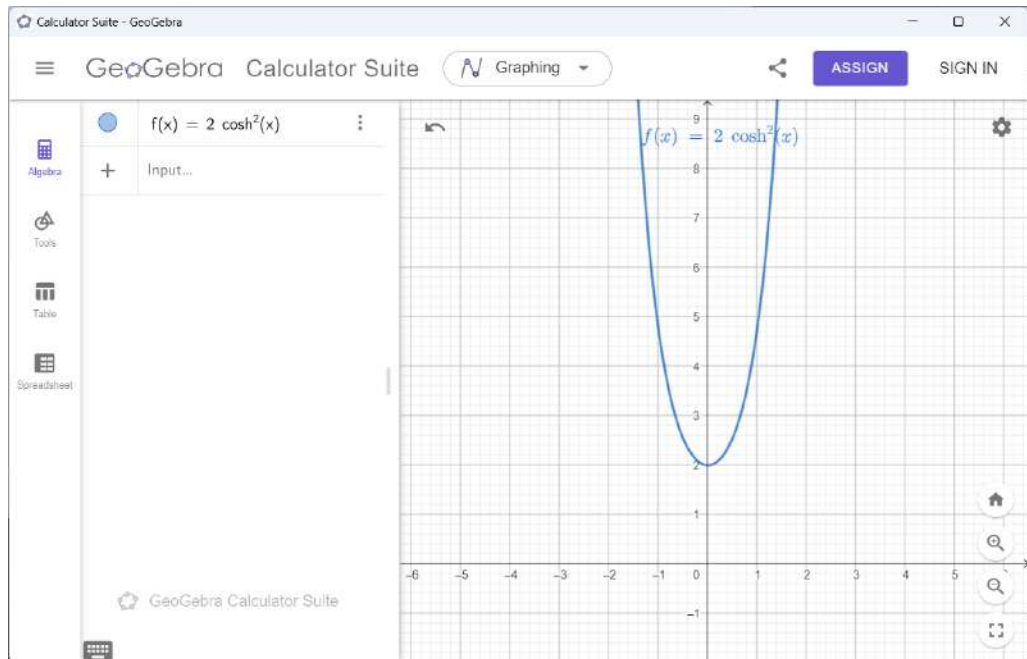
Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.18:

- Ketikkan  $\sinh\left(\frac{1}{2}x - 2\right)$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.35.



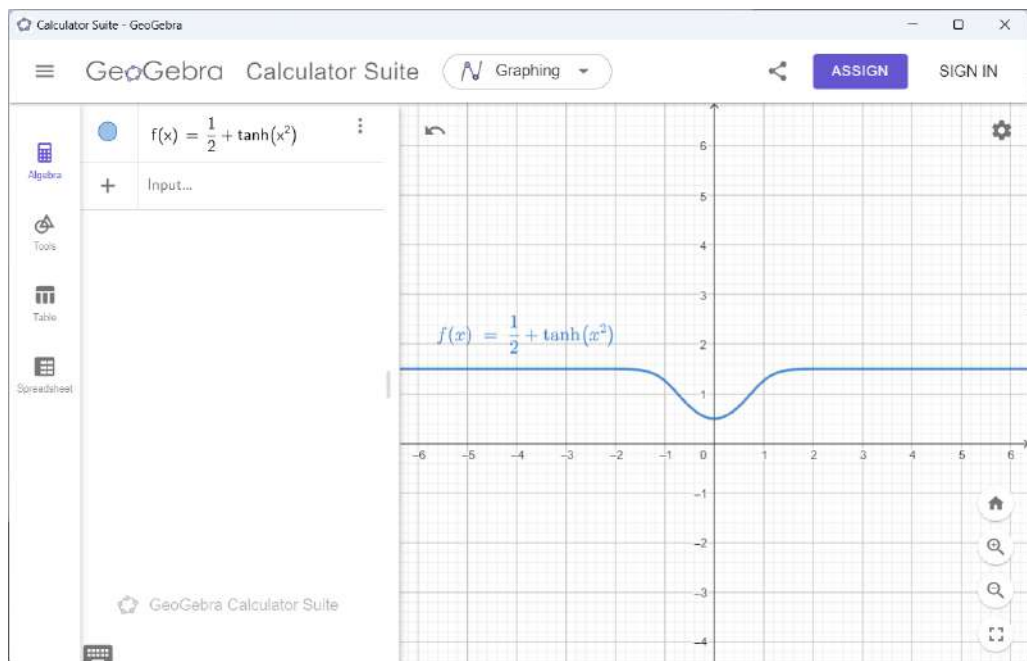
**Gambar 2.35** Grafik  $f(x) = \sinh\left(\frac{1}{2}x - 2\right)$ .

- Ketikkan  $2\cosh^2 x$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.36.



**Gambar 2.36** Grafik  $f(x) = 2 \cosh^2 x$ .

- Ketikkan  $\frac{1}{2} + \tanh x^2$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.37.

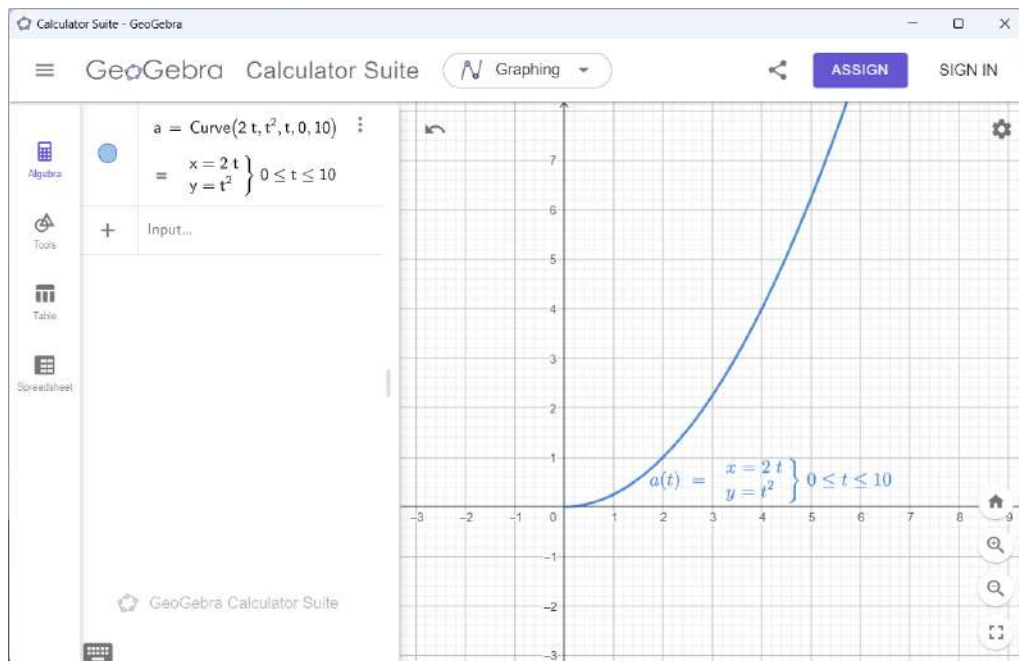


**Gambar 2.37** Grafik  $f(x) = \frac{1}{2} + \tanh x^2$ .

### Praktikum 1.19

Gambarlah grafik fungsi parameter  $\begin{cases} x = 2t \\ y = t^2 \end{cases}, 0 < t < 10$

Praktikum 1.19 dapat diselesaikan dengan langsung mengetikkan  $\text{Curve}(2t, t^2, t, 0, 10)$  di kolom input sehingga diperoleh grafik seperti pada Gambar 2.38.



**Gambar 2.38** Grafik fungsi parameter  $\begin{cases} x = 2t \\ y = t^2 \end{cases}, 0 < t < 10$ .

Yang tak hingga bagaimana?





### 3. Persamaan Kuadrat

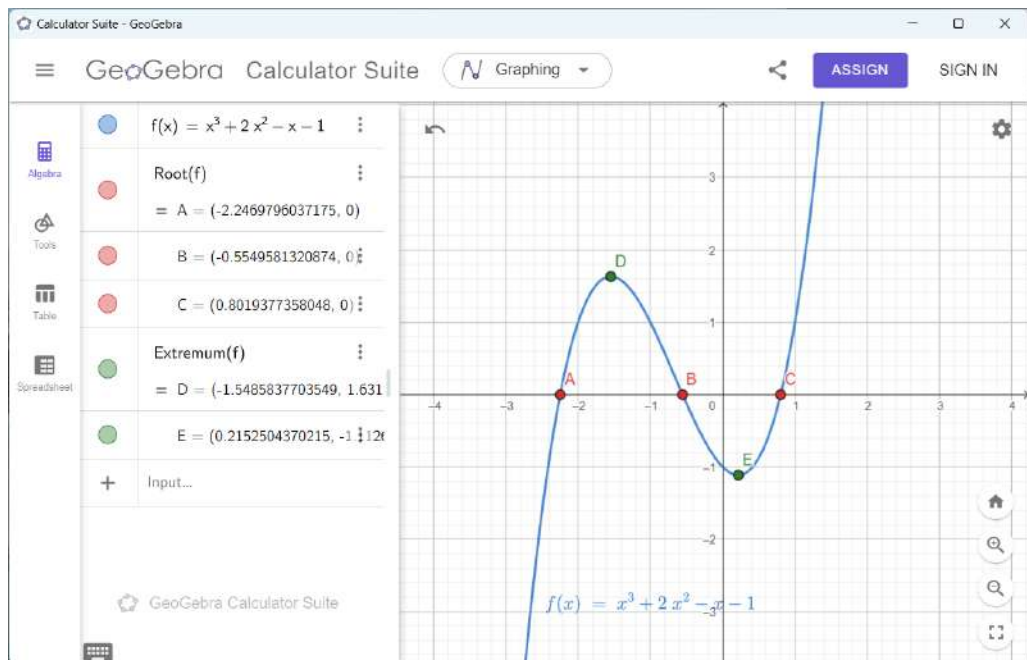
Pada BMP MATA4110 Kalkulus I Modul 7 dan Modul 8 telah dibahas mengenai penggunaan turunan untuk menentukan titik ekstrim, asimtot, dan akar persamaan. Sebelum menggunakan *software* GeoGebra untuk menentukan titik ekstrim, asimtot, dan akar persamaan kuadrat diharapkan Anda telah mempelajari materi tersebut.

### Praktikum 1.20

Gambarlah grafik fungsi  $f(x) = x^3 + 2x^2 - x - 1$ , kemudian tentukan akar persamaan dan titik ekstrimnya!

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.20:

- Ketikkan  $x^3 + 2x^2 - x - 1$  di kolom input.
- Klik Tools , kemudian pilih Roots  dan klik gambar grafik  $f(x) = x^3 + 2x^2 - x - 1$  yang telah dibuat diperoleh titik ekstrim  $A = (-2.247, 0)$ ,  $B = (-0.555, 0)$ , dan  $C = (0.802, 0)$ .
- Klik Tools , kemudian pilih Extremum  dan klik gambar grafik  $f(x) = x^3 + 2x^2 - x - 1$  yang telah dibuat diperoleh titik ekstrim  $D = (-1.549, 1.632)$  dan  $E = (0.215, -1.113)$ .



**Gambar 2.39** Grafik fungsi parameter  $f(x) = x^3 + 2x^2 - x - 1$ , akar persamaan, dan titik ekstremnya.

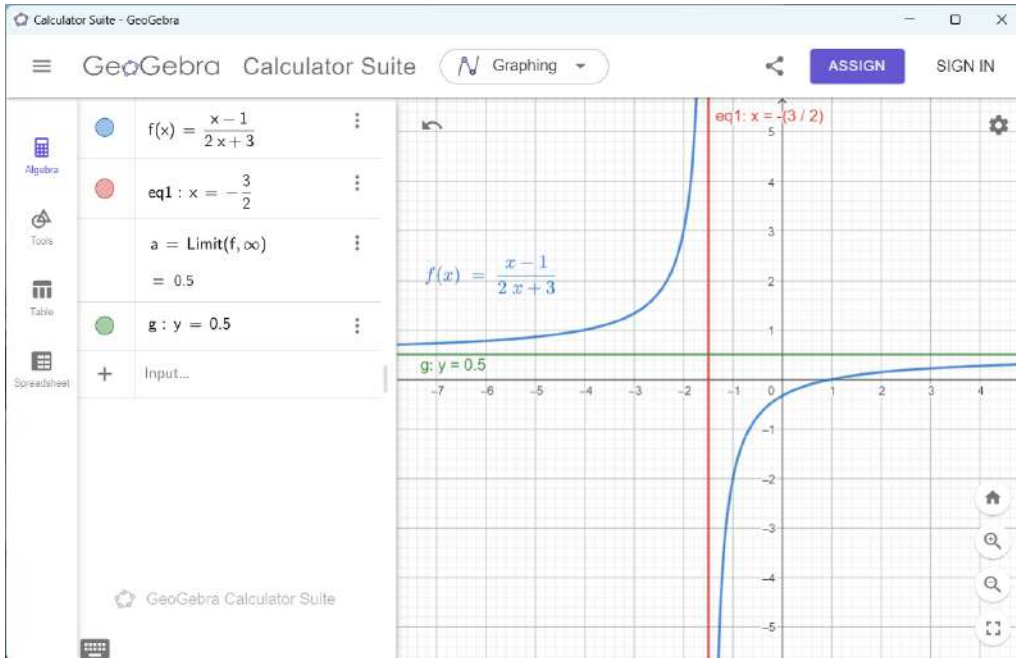
### Praktikum 1.21

Gambarlah grafik fungsi  $f(x) = \frac{x-1}{2x+3}$ , serta gambarkan asimtot tegak dan asimtot datarnya!

Berikut ini adalah langkah-langkah untuk menyelesaikan Praktikum 1.21:

- Ketikkan  $\frac{x-1}{2x+3}$  di kolom input
- Tentukan asimtot tegak dari penyebut  $2x + 3 = 0$  sehingga diperoleh asimtot tegak  $x = -3/2$ . Ketikkan  $x = -3/2$  di kolom input.

- Tentukan asimtot datar dari  $\lim_{x \rightarrow \infty} \frac{x-1}{2x+3}$  dengan mengetikkan  $\text{Limit}(f, \infty)$  di kolom input sehingga diperoleh  $\lim_{x \rightarrow \infty} \frac{x-1}{2x+3} = 0,5$ . Maka asimtot datarnya adalah  $y = 0,5$ . Kemudian ketikkan  $y = 0,5$  di kolom input.



**Gambar 2.40** Grafik fungsi parameter  $f(x) = \frac{x-1}{2x+3}$ , asimtot tegak, dan asimtot datarnya.

#### 4. Latihan

Untuk memperdalam pemahaman Anda dalam melaksanakan praktikum kerjakan soal latihan berikut!

1. Diketahui  $f(x) = 2x^2 - 1$  dan  $g(x) = \frac{1}{\sqrt{x}}$ . Tentukan

Apa perlu dituliskan  $x > 0$ ??

- a.  $f(x) + \frac{1}{2}f(x)g(x)$
- b.  $\frac{1}{f(x)} + g(x)$
- c.  $2(f \circ g)(x)$
- d.  $\frac{1}{2}(g \circ f)(x)$
- e. *invers*

serta gambar grafik dari masing-masing operasi fungsi tersebut!

2. Gambarlah titik  $A(-5,1)$ ,  $B(3,3)$ , dan  $C(4,-3)$ !

3. Gambarlah garis yang melalui dua titik, yaitu titik  $A(-3,3)$  dan  $B(3,-5)$ . Kemudian gambar garis sejajar dan garis tegak lurus dengan garis pertama!
4. Gambarlah sudut  $35^\circ$ ,  $110^\circ$ , dan  $315^\circ$ !
5. Gambarlah segi lima dengan panjang sisinya 2 satuan panjang dan segi tiga dengan tinggi 3 dan panjang alas 4 satuan panjang!
6. Gambarlah balok dengan panjang 4, lebar 2, dan tinggi 3 satuan panjang serta jaring-jaringnya!

### C. PRAKTIKUM VISUALISASI GRAF

#### 1. Materi

Is simply dummy text of the printing and typesetting industry.

#### 2. Pelaksanaan

Is simply dummy text of the printing and typesetting industry.

### D. PRAKTIKUM PEMROGRAMAN LINEAR DAN TAK LINEAR

#### 1. Materi

Is simply dummy text of the printing and typesetting industry.

#### 2. Pelaksanaan

Is simply dummy text of the printing and typesetting industry.

### E. PELAPORAN DAN PENILAIAN

Is simply dummy text of the printing and typesetting industry.

## MODUL 3

### VISUALISASI DATA

Visualisasi data merupakan suatu alat dan cara untuk mengkomunikasikan data dan membantu memahami data dengan memanfaatkan sistem indra visual. Dalam mempersiapkan bahan visualisasi, seseorang perlu memahami dengan baik bagaimana mentransformasikan data (baik yang berupa data mentah maupun statistik ringkasan hasil analisis) menjadi suatu tampilan visual sedemikian rupa sehingga informasi yang ingin disampaikan dapat diterima dengan tepat dan mudah oleh siapapun yang melihatnya.

Salah satu aplikasi yang dapat digunakan untuk memvisualisasikan data adalah Microsoft Excel (disingkat Ms. Excel). Ms. Excel merupakan salah satu perangkat lunak yang populer untuk mengolah angka atau *spreadsheet* (lembar kerja). Ms. Excel dirilis oleh Microsoft sejak tahun 1985 dan telah mengalami sejumlah transformasi yang berkembang dan canggih. Perangkat ini memiliki fitur-fitur yang mendukung proses penyajian data dalam bentuk tabel dan grafik agar mudah dipahami oleh pihak lain.

#### F. PERSYARATAN PESERTA

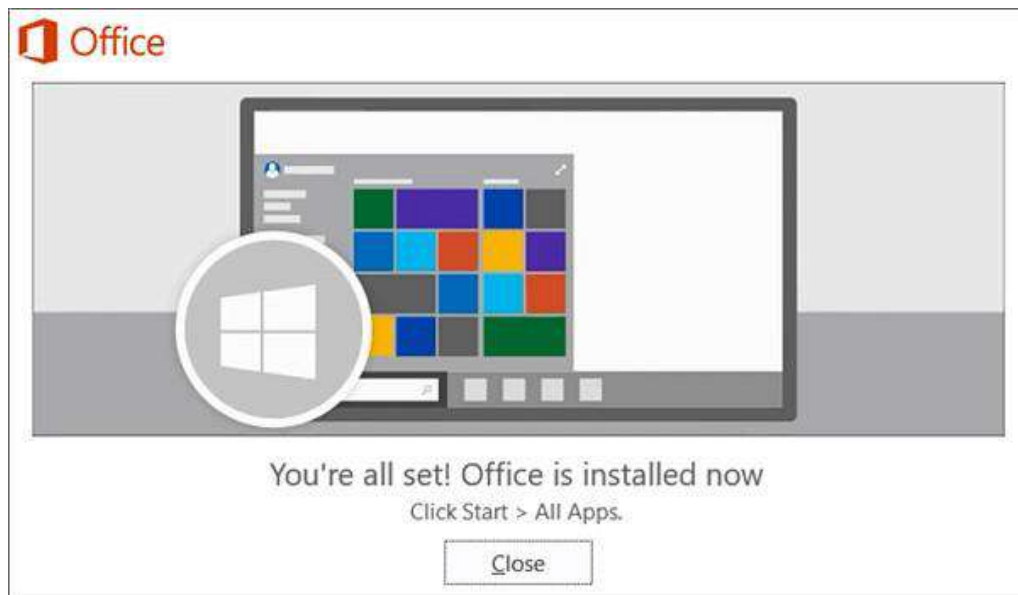
Materi tentang data, penyajian data, dan visualisasi data dapat dipelajari pada mata kuliah Metode Statistika I/ SATS4121 serta Pemodelan dan Visualisasi Data/STMA4312. Oleh karena itu, sebelum mengikuti praktikum visualisasi data, mahasiswa telah mendaftarkan kedua mata kuliah tersebut terlebih dahulu. Selain itu, mahasiswa perlu melakukan instalasi aplikasi Excel di Microsoft 365, Instalasi Add-ins Analysis ToolPak dan VBA pada Ms. Excel, dan persiapan data.

#### 1. Instalasi Aplikasi Excel di Microsoft 365



Praktikum visualisasi data dilaksanakan dengan penggunaan aplikasi Ms. Excel di Microsoft 365 pada Windows berbahasa Inggris. Aplikasi Ms. Excel dipilih karena aplikasi tersebut umum digunakan mahasiswa sejak sekolah menengah. Sebelum melaksanakan praktikum, Anda memastikan bahwa Microsoft 365 tersedia pada perangkat komputer/laptop atau sebagainya. Apabila Anda belum memiliki aplikasi tersebut, Anda perlu melakukan instalasi Microsoft 365 pada PC dengan langkah-langkah sebagai berikut:

- Buka [www.office.com](http://www.office.com), lalu pilih **Sign in**.
- Isikan akun microsoft dan password (contoh nama akun: [nim@ecampus.ut.ac.id](mailto:nim@ecampus.ut.ac.id)) dan pilih **Sign in**
- Pada laman Microsoft 365, pilih **Install Office**
- Selanjutnya, Anda mengikuti perintah di bagian “**Install Microsoft 365**”
- Untuk menginstall office, bergantung pada browser Anda, pilih **Run** (di **Microsoft Edge** atau Internet Explorer)/ **Setup** (di Chrome), atau **Save File** (di Firefox)
- Jika melihat perintah kontrol akun pengguna yang berisi “**Do you want to allow app to make changes to your device?**” Pilih **Yes**
- Penginstalan selesai saat Anda melihat frasa “**You're all set! Office is installed now**”. Pilih **Close**, seperti pada Gambar 3.1



Gambar 3.1 Tampilan setelah Office berhasil diinstal

- Untuk membuka aplikasi, pilih tombol **Start** (sudut kiri bawah layar Anda) lalu ketikkan nama aplikasi, seperti **Excel**
- Untuk membuka aplikasi, pilih ikon **Excel** di hasil pencarian
- Saat aplikasi terbuka, terima *license agreement*. **Microsoft 365** atau **Office** diaktifkan dan siap digunakan. Apabila mahasiswa mengalami kendala saat instal atau mengaktifkan office, mahasiswa dapat mencari informasi pada laman Microsoft Support di tautan <https://support.microsoft.com/>

Selain Microsoft 365, aplikasi Excel dapat juga ditemukan pada Microsoft versi lainnya seperti Microsoft Office 2021 pada PC atau Mac. Namun beberapa fitur dan fungsi yang dipelajari pada praktikum ini kemungkinan tidak tersedia atau perlu penyesuaian pada versi lainnya

## 2. Instalasi Add-ins Analysis ToolPak dan VBA pada Ms. Excel

Fitur-fitur pada Excel diantaranya yaitu **Get Data** pada tab **Data** untuk mengambil/ mengimpor data serta **Charts** pada tab **Insert** untuk memvisualisasikan data dalam bentuk yang lebih mudah dipahami dan menarik. Analisis data dan visualisasi data yang lebih kompleks pada Ms. Excel dapat menggunakan **Analysis ToolPak** dan **Visual Basic for Application (VBA)** pada menu **Add-ins**. Langkah-langkah mengaktifkan **Analysis ToolPak** dan **VBA** adalah sebagai berikut:

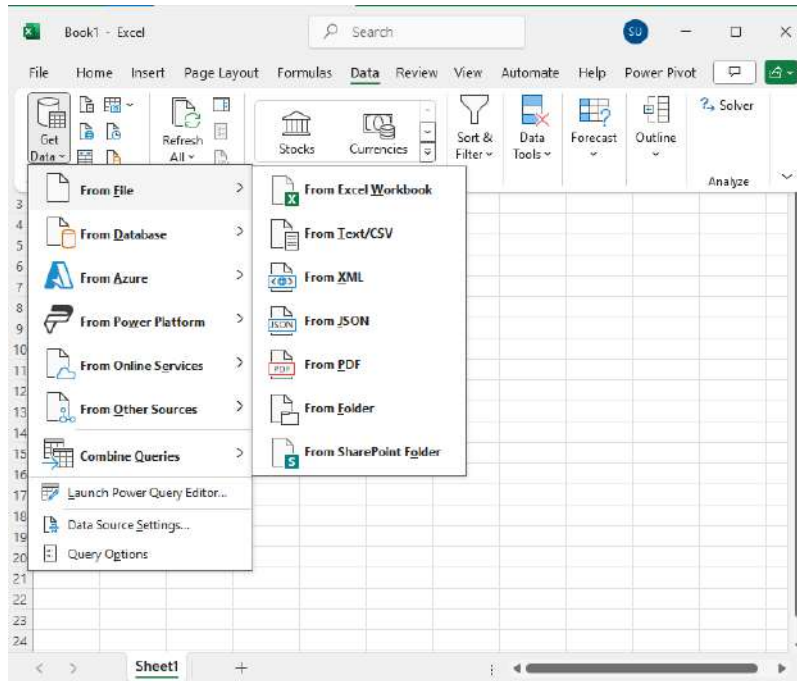
- Buka **Ms. Excel**
- Klik tab **File**, klik **Options**, lalu klik **Add-ins**
- Pada kotak **manage**, pilih **Excel Add-ins**, lalu klik **Go**
- Pada kotak **Add-ins**, centang kotak **Analysis ToolPak** dan **Analysis ToolPak – VBA**, lalu klik **OK**.
  - Jika **Analysis ToolPak** dan **Analysis ToolPak – VBA** tidak terdaftar dalam kotak **Add-ins** yang tersedia, klik **browse** untuk menemukannya
  - Jika terdapat pemberitahuan bahwa **Analysis ToolPak** dan **Analysis ToolPak – VBA** tidak terinstal di komputer user, klik **Ya** untuk melakukan instal.

### 3. Persiapan Data

Untuk bekerja dengan Excel, data perlu dikumpulkan dari beberapa sumber seperti buku, file, dan internet. Beberapa data tersedia pada modul praktikum ini, selainnya data dapat diunduh pada URL berikut : <https://sl.ut.ac.id/data-PVO> . Langkah-langkah input data di lembar kerja diberikan sebagai berikut:


- Buka **Ms. Excel**
- Klik sel yang kosong atau klik dua kali sel yang tidak kosong
- Ketikkan data (angka atau teks) yang ingin diinputkan
- Tekan **ENTER** atau **TAB** pada keyboard

Selain input data di atas, data dapat diimpor ke lembar kerja melalui fitur **Get Data** pada tab **Data** seperti pada Gambar 3.2. Data dapat di impor dari file (Excel Workbook, Text/CSV, XML, PDF, dan sebagainya), Database, Azure, dan sumber lainnya.



Gambar 3.2 Fitur **Get Data** pada tab **Data**

Untuk impor data dari sumber web, Anda dapat mengikuti langkah-langkah berikut ini:

- Buka **Ms. Excel**
- Klik tab **Data**, pilih **Get Data From Web** ()
- Salin **URL Web** yang memuat data yang akan di impor, sebagai ilustrasi Data Harga Saham Historis Bank Syariah Indonesia dengan tautan [https://ir.bankbsi.co.id/historical\\_price.html](https://ir.bankbsi.co.id/historical_price.html)
- Pada **From Web**, pilih **Basic** dan tempel **URL Web** yang telah disalin, dan klik **OK**

- Selanjutnya, pada panel kiri jendela **Navigator** yang menampilkan daftar tabel, pilih **Table** yang akan diimpor ke lembar kerja, dan klik **Load**. Data tabel yang dipilih telah diimpor ke lembar kerja seperti pada Gambar 3.3 dan siap dirapikan.


Tanggal	Perbaikan	Terjadi	Tindakan	Status
18/09/2024	1.480	0.570	1.850	3.900
19/09/2024	1.440	0.240	1.680	3.360
20/09/2024	1.540	2.790	1.020	3.810
21/09/2024	1.760	2.290	1.740	3.740
22/09/2024	1.740	0.330	1.290	3.490
23/09/2024	1.800	0.330	1.980	3.960
24/09/2024	1.810	0.570	1.980	3.960
25/09/2024	1.440	0.330	1.980	3.960
26/09/2024	1.690	1.110	1.980	3.960
27/09/2024	1.020	0.660	1.980	3.960
28/09/2024	1.120	0.660	1.980	3.960
29/09/2024	1.050	0.330	1.980	3.960

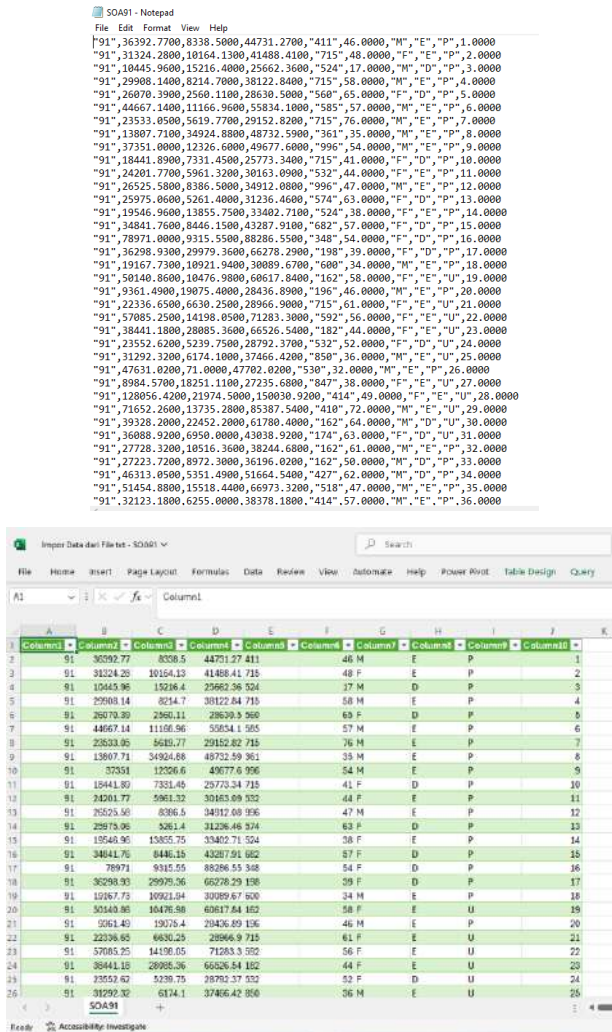
  

A	B	C	D	E	F	G
18/09/2024	1.480	0.570	1.850	3.900		
19/09/2024	1.440	0.240	1.680	3.360		
20/09/2024	1.540	2.790	1.020	3.810		
21/09/2024	1.760	2.290	1.740	3.740		
22/09/2024	1.740	0.330	1.290	3.490		
23/09/2024	1.800	0.330	1.980	3.960		
24/09/2024	1.810	0.570	1.980	3.960		
25/09/2024	1.440	0.330	1.980	3.960		
26/09/2024	1.690	1.110	1.980	3.960		
27/09/2024	1.020	0.660	1.980	3.960		
28/09/2024	1.120	0.660	1.980	3.960		
29/09/2024	1.050	0.330	1.980	3.960		

Gambar 3.3. Impor data tabel dari web (kiri) ke lembar kerja (kanan)

Untuk impor data dari file .txt, Anda dapat mengikuti langkah-langkah berikut ini:

- Buka **Ms. Excel**
- Klik tab **Data**, pilih **Get Data**, lalu pilih **From File**, dan pilih **From Text/CSV** ()
- Pilih file txt yang memuat data yang akan diimpor, sebagai ilustrasi file Group Medical Insurance Large Claims Database Tahun 1991 dengan nama file SOA91.txt yang tersedia pada tautan <https://sl.ut.ac.id/data-PVO>. Selanjutnya klik **Import** dan **Load**. Data tabel yang dipilih telah diimpor ke lembar kerja seperti pada Gambar 3.4 dan siap dirapikan.



Gambar 3.4 Impor data tabel dari file .txt (kiri) ke lembar kerja (kanan)

**G. MATERI PRAKTIKUM VISUALISASI DATA**

Beberapa fungsi pada Excel yang diperlukan untuk visualisasi data diberikan pada Tabel 3.1.

Tabel 3.1 Beberapa Fungsi pada Ms. Excel

Fungsi	Kegunaan
--------	----------

COUNT	menghitung jumlah sel yang berisi angka dalam suatu rentang sel atau baris tertentu
MIN	mencari nilai terkecil dari sekumpulan angka dalam suatu rentang
MAX	mencari nilai tertinggi dalam sekumpulan data yang ditentukan
ROUNDDOWN	membulatkan angka ke bawah ke bilangan bulat terdekat atau ke sejumlah tempat desimal tertentu
FREQUENCY	menghitung frekuensi kemunculan nilai dalam rentang nilai, lalu mengembalikan array vertikal angka.
SUM	menjumlahkan nilai-nilai dalam suatu rentang, referensi sel, atau angka individual

Secara umum, bentuk-bentuk penyajian data dalam rangka visualisasi data diklasifikasikan ke dalam dua bagian, yaitu tabel dan grafik. Prosedur penyajian data tersebut akan dibahas satu persatu.

### 1. TABEL

Tabel terbagi menjadi tiga, yaitu tabel satu arah, tabel silang, dan tabel frekuensi. Praktikum masing-masing tabel diberikan berikut ini

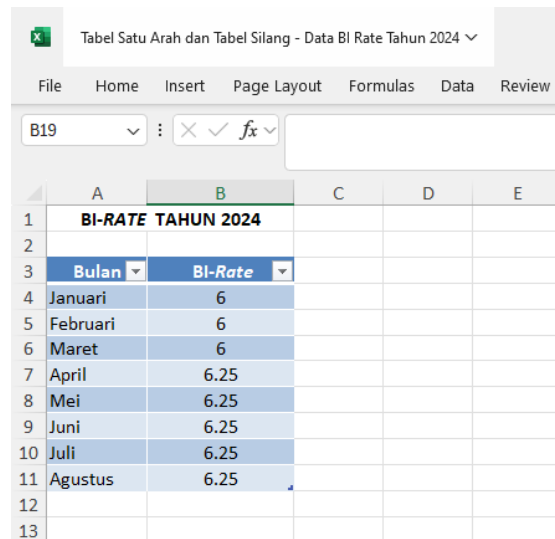
#### Praktikum 3.1 (Tabel Satu Arah)

Bank Indonesia menetapkan suku bunga yang disebut *BI-Rate*. Berdasarkan Buku Statistik Indonesia Tahun 2024, *BI-Rate* bulan Januari, Februari, dan Maret tahun 2024 sebesar 6% dan meningkat 0,25% pada bulan April sampai dengan bulan Agustus 2024. Tentukan tabel dari data *BI-Rate* tersebut!

#### Jawaban Praktikum 3.1

Tabel yang dapat digunakan untuk memvisualisasi data *BI-Rate* tersebut yaitu tabel satu arah. Langkah-langkah membuat tabel satu arah adalah sebagai berikut:

- Lakukan input data ke dalam bentuk baris atau kolom atau impor data pada lembar kerja (*spreadsheet*). Pada praktikum 3.1, baris mengilustrasikan bulan dan kolom menggambarkan besaran *BI-Rate* dalam persen (%)
- Klik menu **Insert**
- Pilih **Table**
- Blok sel-sel yang memuat data yang akan dibuatkan tabel. Selanjutnya untuk membuat header, beri centang pada **My Table has Headers**
- Pilih **OK**. Selanjutnya, akan tampil tabel *BI-Rate* Tahun 2024. Anda dapat menambahkan judul pada sel kosong di atas tabel yang dihasilkan seperti pada Gambar 3.5 berikut ini.



BI-RATE TAHUN 2024	
Bulan	BI-Rate
Januari	6
Februari	6
Maret	6
April	6.25
Mei	6.25
Juni	6.25
Juli	6.25
Agustus	6.25

Gambar 3.5. Tabel satu arah *BI-Rate* Tahun 2024

### Praktikum 3.2 (Tabel Silang)

Jumlah mahasiswa pada Perguruan Tinggi Negeri dan Perguruan Tinggi Swasta di bawah Kemendikbudristek/ Kemendikbud 5 tahun terakhir tercatat pada Buku



Statistik Indonesia tahun 2019 sampai dengan 2022. Buku Statistik Indonesia tersedia pada tautan <https://sl.ut.ac.id/data-PVO> . Tentukan tabel yang sesuai untuk memvisualisasikan data tersebut!

### Jawaban Praktikum 3.2

Tabel yang dapat digunakan untuk memvisualisasi data dua variabel (jumlah mahasiswa pada Perguruan Tinggi Negeri dan Perguruan Tinggi Swasta di bawah Kemendikbudristek/ Kemendikbud) dapat menggunakan tabel silang. Langkah-langkah membuat tabel silang diberikan sebagai berikut:

- Lakukan input data ke dalam bentuk baris atau kolom atau impor data pada lembar kerja. Untuk praktikum 3.2, baris merupakan tahun dan dua kolom menggambarkan jumlah mahasiswa Perguruan Tinggi Negeri dan Perguruan Tinggi Swasta)
- Klik menu **Insert**
- Pilih **Table**
- Blok sel-sel yang memuat data yang akan dibuatkan tabel. Selanjutnya untuk membuat header, beri centang pada **My Table has Headers**
- Pilih **OK**. Selanjutnya, akan tampil tabel Jumlah Mahasiswa (Negeri dan Swasta) di Bawah Kemendikbudristek/ Kemendikbud di Indonesia Tahun 2018-2022. Anda dapat menambahkan judul pada sel kosong di atas tabel yang dihasilkan seperti pada Gambar 3.6 berikut ini.

The screenshot shows an Excel spreadsheet with a pivot table. The pivot table is titled "Jumlah Mahasiswa (Negeri dan Swasta) di Bawah Kemendikbudristek/ Kemendikbud di Indonesia Tahun 2018-2022". The rows represent years from 2018 to 2022, and the columns represent "Negeri" and "Swasta".

Tahun	Jumlah Mahasiswa	
	Negeri	Swasta
2018	2492103	4459021
2019	2928403	4410761
2020	2994015	4374994
2021	3205606	4459910
2022	3379828	4495453

Gambar 3.6. Tabel silang Tahun 2024

**Praktikum 3.3 (Tabel Frekuensi)**

Tentukan tabel frekuensi (frekuensi, frekuensi kumulatif, dan frekuensi relatif) pada Excel dari data nilai ujian 100 mahasiswa yang dibangkitkan secara acak. Data nilai ujian 100 mahasiswa tersebut diberikan pada tabel 3.2.

Tabel 3.2. Data Nilai Ujian 100 Mahasiswa

79	16	13	16	1	50	33	8	89	57
7	19	31	41	53	82	78	19	31	96
28	99	31	10	49	23	32	32	1	63
25	93	19	17	23	65	36	93	68	37
36	35	80	21	80	68	63	82	34	94
61	78	66	7	27	44	84	45	82	3
95	31	51	99	13	6	0	54	98	40
13	20	6	34	86	14	91	87	56	12
62	54	57	57	23	2	70	87	56	43

44	51	16	37	86	92	75	39	90	82
----	----	----	----	----	----	----	----	----	----

### Jawaban Praktikum 3.3

Untuk menyelesaikan praktikum 3.3, langkah-langkah menyusun tabel frekuensi pada Ms. Excel adalah sebagai berikut:

- Input data.

Misal input data nilai ujian 100 mahasiswa pada tabel 2.2 ke data array **A3:J12** (baris 3 sampai dengan baris 12 dan kolom A sampai dengan kolom J pada lembar kerja) seperti pada Gambar 3.7.

	A	B	C	D	E	F	G	H	I	J
1	DATA NILAI UJIAN 100 MAHASISWA									
2										
3	79	16	13	16	1	50	33	8	89	57
4	7	19	31	41	53	82	78	19	31	96
5	28	99	31	10	49	23	32	32	1	63
6	25	93	19	17	23	65	36	93	68	37
7	36	35	80	21	80	68	63	82	34	94
8	61	78	66	7	27	44	84	45	82	3
9	95	31	51	99	13	6	0	54	98	40
10	13	20	6	34	86	14	91	87	56	12
11	62	54	57	57	23	2	70	87	56	43
12	44	51	16	37	86	92	75	39	90	82
13										
14										
15										

Gambar 3.7. Tabel Data Nilai Ujian 100 Mahasiswa di Lembar Kerja

- Siapkan kerangka tabel frekuensi (data array L13:O20) dengan *header* Kategori Nilai pada sel **L13**, Frekuensi pada sel **M13**, Frekuensi Kumulatif pada sel **N13**, dan Frekuensi Relatif pada sel **O13** seperti pada Gambar 3.8.

The screenshot shows a spreadsheet with a table of student exam scores and a frequency table structure. The table of scores is titled "DATA NILAI UJIAN 100 MAHASISWA" and contains 12 rows of data. The frequency table structure is located to the right of the data table and has four columns: "Kategori Nilai", "Frekuensi", "Frekuensi Kumulatif", and "Frekuensi Relatif".

DATA NILAI UJIAN 100 MAHASISWA										
79	16	13	16	1	50	33	8	89	57	
7	19	31	41	53	82	78	19	31	96	
28	99	31	10	49	23	32	32	1	63	
25	93	19	17	23	65	36	93	68	37	
36	35	80	21	80	68	63	82	34	94	
61	78	66	7	27	44	94	45	82	3	
95	31	51	99	13	6	0	54	98	40	
13	20	6	34	86	14	91	87	56	12	
62	34	37	37	23	2	70	87	36	43	
44	51	16	37	86	92	75	39	90	82	

Kategori Nilai	Frekuensi	Frekuensi Kumulatif	Frekuensi Relatif

Gambar 3.8. Kerangka Tabel Frekuensi

- Menentukan nilai tertinggi data.

Misal klik sel **M3** (sel sembarang) dan input fungsi MAX terhadap data array atau dapat diketikkan pada sel **M3**:

$$=MAX(A3:J12)$$

dan tekan **ENTER**.

- Menentukan nilai terendah data.

Misal klik sel **M4** (sel sembarang) dan input fungsi MIN terhadap data array atau dapat diketikkan pada sel **M4**:

$$=MIN(A3:J12)$$

dan tekan **ENTER**.

- Menentukan range, yaitu selisih nilai tertinggi dengan nilai terendah.

Misal klik sel **M5** (sel sembarang) untuk memuat nilai range. Ketikkan pada sel **M5**:

$$=M3-M4$$

dan tekan **ENTER**.

- Menentukan banyak kelas dengan fungsi **ROUND**DOWN tanpa desimal. Pendekatan yang cukup baik digunakan adalah  $k = 1 + 3,3 \log (n)$ .

Misal klik sel **M6** (sel sembarang) dan ketikkan:

**=ROUND**DOWN(1+(3.3)\*LOG(100),0)

dan tekan **ENTER**.

- Menentukan lebar interval dengan fungsi **ROUND**DOWN tanpa desimal terhadap Range/ Banyak Kelas.

Misal klik sel **M7** dan ketikkan:

**=ROUND**DOWN(M5/M6, 0)

dan tekan **ENTER**.

- Menetapkan batas-batas kelas pada kolom Kategori Nilai pada data\_array **L14:L20** dengan memperhatikan lebar interval
- Menghitung frekuensi atau banyaknya data di tiap kelas dengan fungsi **FREQUENCY** terhadap data\_array A3:J12 dan bin\_array O4:O10. **Bin** adalah kolom untuk data batas atas kelas.

Misal klik sel **P4** (sel sembarang) dan ketikkan

**=FREQUENCY**(A3:J12, \$O\$4:\$O\$10)

lalu tekan **ENTER**. **Salin** kolom **P4:P10** dan **tempelkan** pada data array **M14:M20** sebagai frekuensi data.

- Menghitung frekuensi kumulatif.

Misal klik sel **N14** (sel sembarang) dan ketikkan

**=M14**

Nilai di sel **N14** merupakan data pertama frekuensi kumulatif yang sama besarnya dengan frekuensi di **M14**. Untuk frekuensi kumulatif interval kelas berikutnya, isi dengan nilai frekuensi pada interval kelas tersebut dan tambahkan dengan nilai frekuensi kumulatif interval kelas sebelumnya.

Misal pada interval kelas 14-28, klik sel **N15** (sembarang) dan ketikkan

$$=M15+N14$$

dan tekan **ENTER**. Hal yang sama untuk interval kelas berikutnya. Frekuensi kumulatif interval kelas terakhir adalah banyak data, yaitu 100.

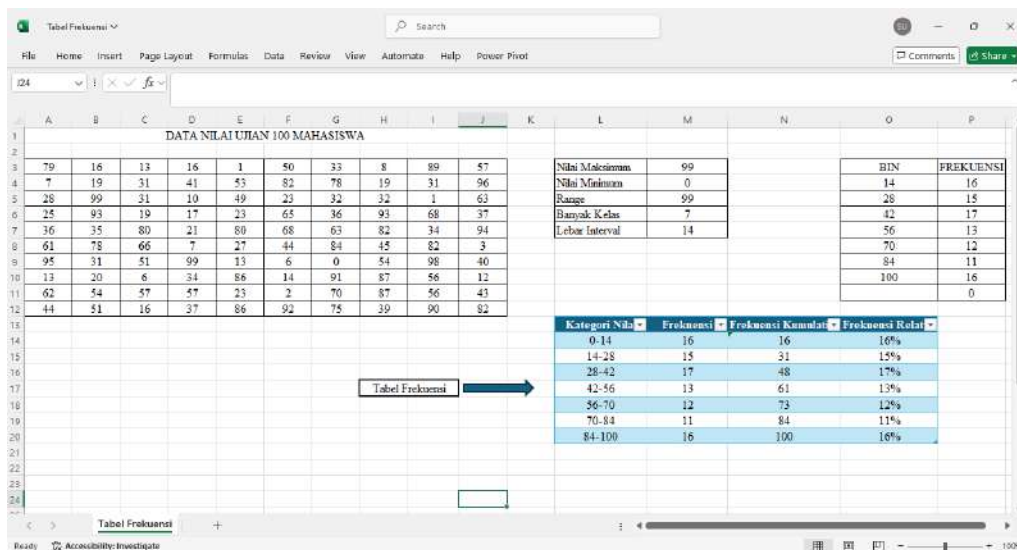
- Menghitung frekuensi relatif yang merupakan rasio frekuensi dengan banyak data.

Misal frekuensi relatif interval kelas pertama (0-14), klik **O14** dan ketikkan:

$$=N14/SUM(M14:M20)$$

lalu tekan **ENTER**. Lakukan hal yang sama untuk interval kelas lainnya. Upayakan frekuensi relatif dalam persen. Cara ubah desimal ke persen adalah klik tab **Home**, pilih **Number**, dan klik **Percent Style**.

- Atur tabel yang telah terisi data dengan rapi dan selesai. Tabel frekuensi (data array L13: O20) yang telah dibuat dapat dilihat pada gambar 3.9.



Gambar 3.9 Tabel Frekuensi Data Nilai Ujian 100 Mahasiswa

**Catatan Tambahan:**

Data bangkitan nilai ujian 100 mahasiswa dibuat dengan memadukan fungsi RAND dan INT. Jika Anda mencoba membangkitkan nilai ujian 100 mahasiswa pada Excel saat ini, data yang dihasilkan mungkin berbeda dari tabel 2.2 karena fungsi RAND menghasilkan data acak dan berubah setiap kali sel dihitung. Namun, mahasiswa perlu mengetahui langkah-langkah untuk membangkitkan data. Penjelasan untuk membangkitkan nilai ujian 100 mahasiswa diberikan pada poin-poin berikut ini:

- Fungsi RAND menghasilkan angka acak yang terdistribusi secara merata, lebih besar atau sama dengan 0, dan kurang dari 1. Untuk menghasilkan angka acak pada 0-100, maka fungsi RAND dikali dengan 100 atau dapat dituliskan :  $RAND()*100$ .
- Fungsi INT pada Ms. Excel berfungsi untuk membulatkan angka ke bawah hingga bilangan bulat terdekat, sehingga kombinasi fungsi yang digunakan untuk membangkitkan nilai ujian acak pada 0-100 adalah  $INT(RAND()*100)$
- Untuk data acak yang tidak berubah, blok 100 nilai yang dihasilkan, salin (CTRL+C) dan tempel (CTRL+V) pada sel-sel data array yang sama. Pastikan fitur tempel yang digunakan adalah tempel spesial nilai.

## 2. GRAFIK

Beberapa tampilan visual grafik yaitu disajikan dalam bentuk *bar chart*, *grouped-bar chart*, *stacked-bar chart*, histogram, *line chart*, *scatter plot*, *pie chart*, dan *box-plot*.

### Praktikum 3.4 (*Bar Chart*)

Tabel 3.4 merupakan tabel satu arah yang menunjukkan data Volume Ekspor ke Negara Asean di Tahun 2023 (dalam ribu ton).

Tabel 3.4. Data Volume Ekspor ke Negara Asean di Tahun 2023 (dalam ribu ton)

No	Nama Negara	Volume Ekspor
1	Thailand	22521.1
2	Singapura	19331.3
3	Filipina	31116.4
4	Malaysia	34199.3
5	Myanmar	1329.7
6	Kamboja	2893
7	Brunei	650.6
8	Laos	1.1
9	Vietnam	18488.4


Sumber: Statistik Indonesia 2023 pada [www.bps.go.id](http://www.bps.go.id)

Buatlah *bar chart* tegak dan horizontal dari data tersebut!

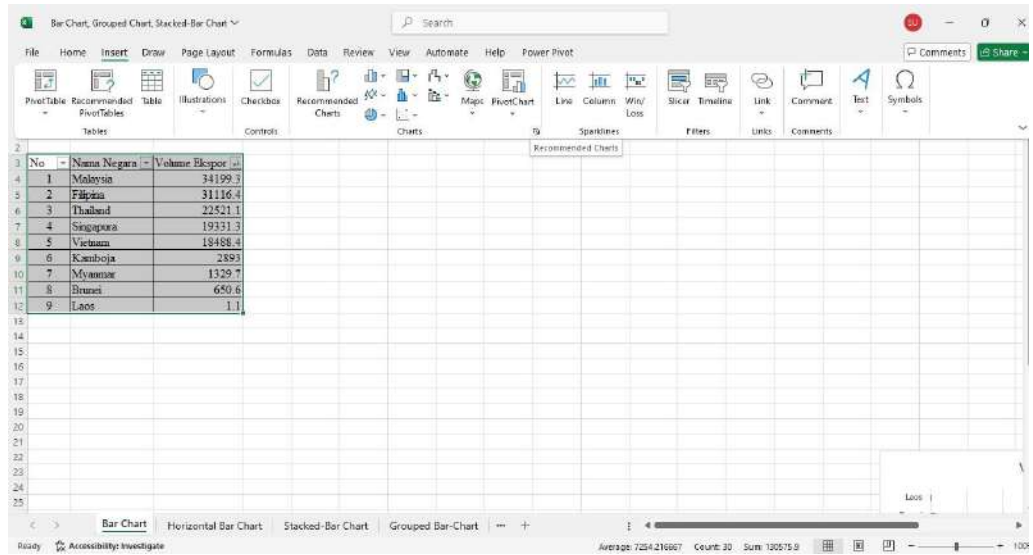
#### Jawaban Praktikum 3.4

Langkah-langkah menyajikan diagram batang (*bar chart*) dari data pada Tabel 3.5 adalah sebagai berikut:

- Input data Volume Ekspor ke Negara Asean di Tahun 2023 pada lembar kerja dan organisir
- Blok kolom **Nama Negara** dan **Volume Ekspor (dalam ribu rupiah)**.

Pilih tab **Insert**. Klik **Chart** (pojok kanan bawah ) seperti pada Gambar 3.10



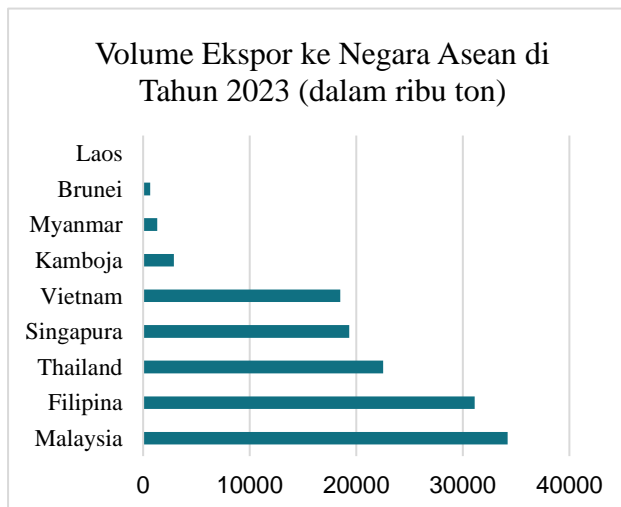
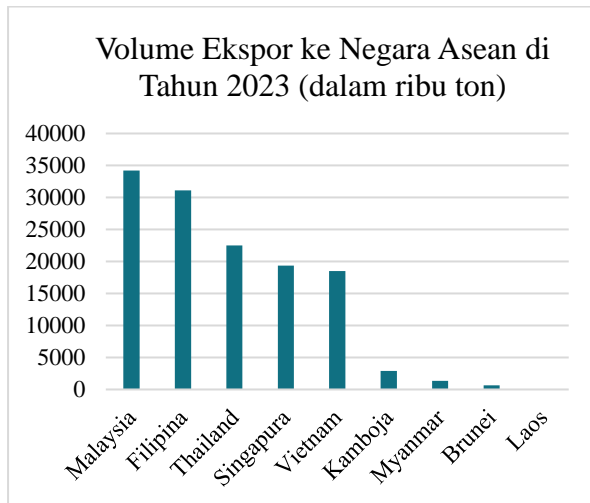


Gambar 3.10 Menu Insert dan pilihan chart

### 3. Pilih **All Charts**

4. Pilih **Columns** untuk diagram batang tegak atau pilih **Bar** untuk diagram batang horizontal. Pilih diagram batang yang sesuai.

5. Klik **OK**. Luaran yang dihasilkan seperti pada Gambar 3.11.



Sumber: Statistik Indonesia 2023 pada [www.bps.go.id](http://www.bps.go.id)

Gambar 3.11 Diagram batang tegak (kiri) dan diagram batang horizontal (kanan) yang memberikan visualisasi data volume ekspor ke negara Asean di tahun 2023

Interpretasi:

Gambar 3.11 menampilkan diagram batang tegak dan diagram batang horizontal dari data volume ekspor ke negara Asean di tahun 2023 yang dihasilkan dari Ms. Excel. Volume ekspor dinyatakan dalam satuan ribu ton. Secara umum dengan melihat diagram ini kita dapat mengetahui bahwa Malaysia merupakan tujuan

utama ekspor Indonesia pada tahun 2023 dengan volume ekspor mencapai 34199.3 ribu ton.


### **Praktikum 3.5 (*Grouped-Bar Chart dan Stacked-Bar Chart*)**

Sajikanlah data Jumlah Mahasiswa di Perguruan Tinggi Negeri (PTN) dan Perguruan Tinggi Swasta (PTS) di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018- 2022 dari tabel silang yang dihasilkan pada praktikum 3.2 dengan diagram *grouped-bar chart* dan *stacked-bar chart*!

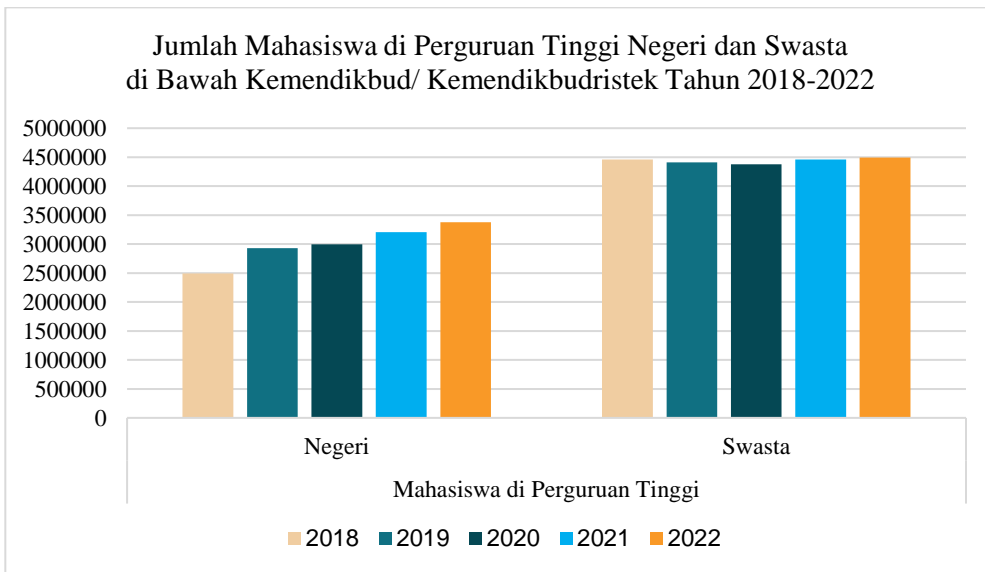
### **Jawaban Praktikum 3.5**

Varian lain dari diagram batang yaitu *grouped-bar chart* di mana untuk setiap kategori yang ada masih terbagi lagi ke dalam beberapa subkategori. Selain itu, penyajian dengan batang yang ditampilkan secara berkelompok dapat pula dilakukan dengan menumpuk batang-batang tersebut. Batang dari satu kategori di kelompok yang sama diletakkan di atas batang dari kategori lainnya. Penyajian ini disebut dengan *stacked-bar chart*.

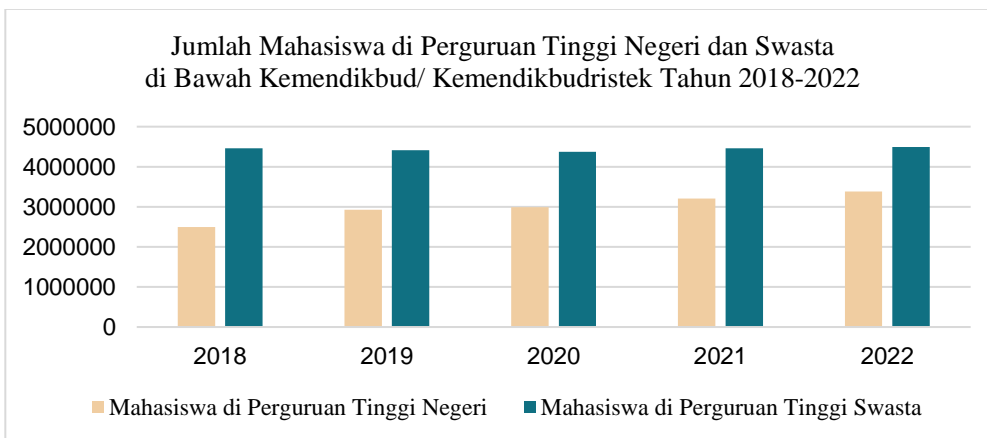
Langkah-langkah menyajikan data dari tabel silang dengan *grouped-bar chart* diberikan pada uraian berikut:

- Input atau impor data Jumlah Mahasiswa di Perguruan Tinggi Negeri dan Swasta di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018-2022 dari hasil praktikum 3.2 ke dalam suatu tabel pada Sheet Excel. Pada tabel, baris tabel menggambarkan subkategori jenis PT dan kolom tabel menggambarkan tahun
- Blok Tabel data, Pilih tab **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Untuk menampilkan *grouped-bar chart* pilih **Columns** → **Clustered Column** untuk diagram batang tegak atau pilih **Bar** untuk diagram batang horizontal.

- Pilih diagram batang yang sesuai
- Klik **OK**. Luaran *grouped-bar chart* dari data Jumlah Mahasiswa di Perguruan Tinggi Negeri dan Swasta di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018-2022 masing-masing dapat dilihat pada Gambar 3.11




(a)

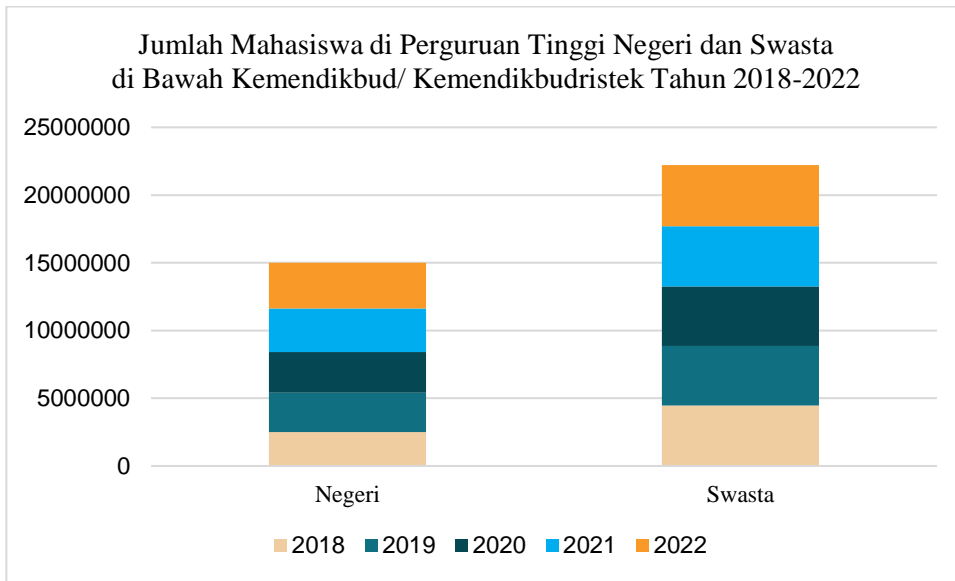


(b)

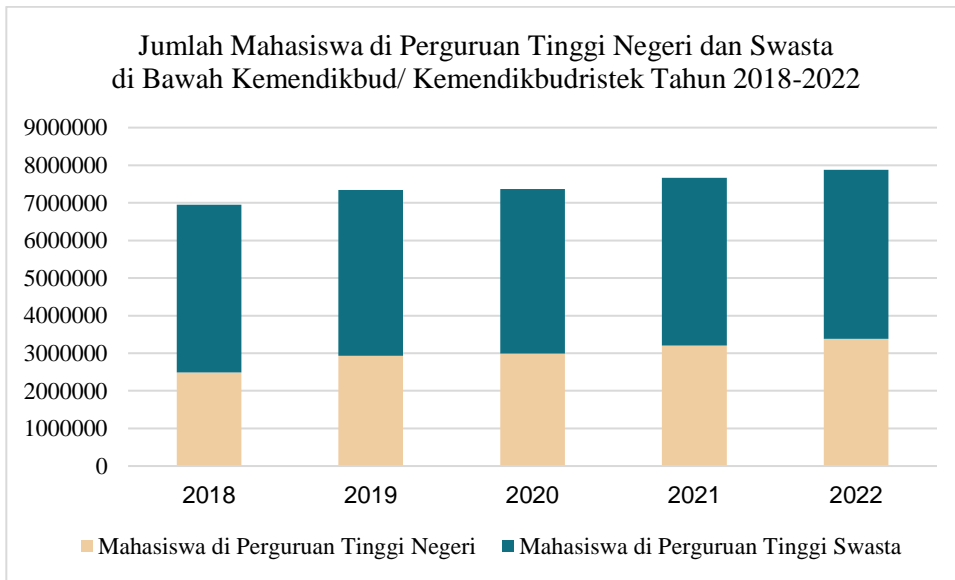
Gambar 3.11 (a) dan (b) *Grouped-bar chart* data Jumlah Mahasiswa di Perguruan Tinggi Negeri dan Swasta di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018-2022

Langkah-langkah menyajikan data dari tabel silang dengan *stacked-bar chart* diberikan pada uraian berikut:

- Input atau impor data Jumlah Mahasiswa di Perguruan Tinggi Negeri dan Swasta di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018-2022 dari hasil praktikum 3.2 ke dalam suatu tabel pada Sheet Excel. Pada tabel, baris tabel menggambarkan subkategori jenis PT dan kolom tabel menggambarkan tahun
- Blok Tabel data, Pilih tab **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Untuk menampilkan *stacked-bar chart*, pilih **Columns** → **Stacked Column**.
- Pilih diagram batang yang sesuai
- Klik **OK**. Luaran *stacked bar chart* dari data Jumlah Mahasiswa di Perguruan Tinggi Negeri dan Swasta di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018-2022 masing-masing dapat dilihat pada Gambar 3.12.



(a)



(b)

Gambar 3.12 (a) dan (b) *Stacked-bar chart* data Jumlah Mahasiswa di Perguruan Tinggi Negeri dan Swasta di Bawah Kemendikbud/ Kemendikbudristek Tahun 2018-2022

Interpretasi:

Pada Gambar 3.11 (a) dan 3.12 (a) disajikan banyaknya mahasiswa per tahun masuk untuk masing-masing kategori perguruan tinggi. Terdapat dua kategori perguruan tinggi yang ada pada data tersebut yaitu perguruan tinggi negeri dan perguruan tinggi swasta. Sedangkan Gambar 3.11 (b) dan 3.12 (b) memberikan ilustrasi banyaknya mahasiswa di setiap tahun terdiri atas dua berdasarkan kategori perguruan tinggi. Gambar 3.11 dan Gambar 3.12 keduanya memberikan informasi bahwa jumlah mahasiswa di perguruan tinggi swasta di bawah Kemendikbud/Kemendikbudrsitek tahun 2018-2022 lebih banyak dibandingkan jumlah mahasiswa di perguruan tinggi negeri. Jumlah mahasiswa di perguruan tinggi negeri mengalami kenaikan di tiap tahunnya sedangkan mahasiswa di perguruan tinggi swasta menunjukkan penurunan jumlah mahasiswa pada tahun 2018 sampai dengan 2020 dan kenaikan jumlah mahasiswa kembali pada tahun 2021 sampai dengan tahun 2022.


### **Praktikum 3.6 (Histogram)**

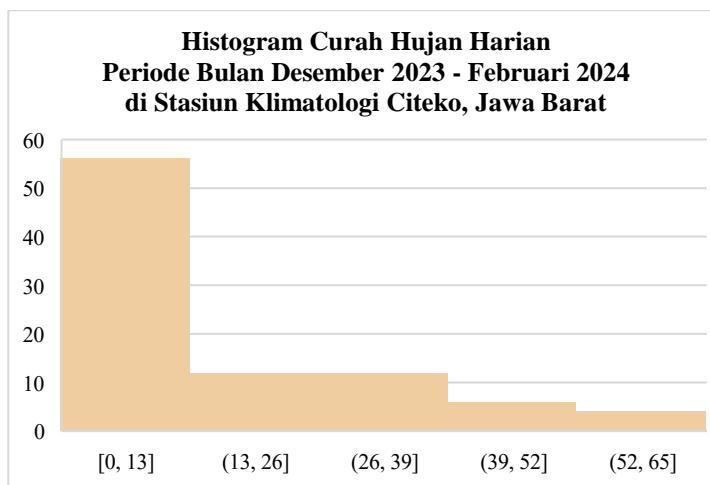
Data curah hujan harian periode bulan Desember 2023 sampai dengan bulan Februari 2024 di Stasiun Klimatologi Jawa Barat yang diperoleh dari web Data Online Pusat Database-BMKG dan telah di *cleaning* tersedia pada <https://sl.ut.ac.id/data-PVO>. Buatlah histogram dari data curah hujan harian tersebut dan berikan interpretasi dari visualisasi tersebut!

### **Jawaban Praktikum 3.6**

Histogram dapat dibuat dengan dua cara, yaitu melalui fitur Charts atau fitur Data Analysis.

- a) Langkah-langkah untuk membuat histogram sebaran data curah hujan harian periode bulan Desember 2023 sampai dengan bulan Februari 2024 di Stasiun Klimatologi Jawa Barat melalui fitur Charts adalah sebagai berikut:

- Siapkan data tabel (tersedia pada file Excel di tautan <https://sl.ut.ac.id/data-PVO>)
- Blok Tabel data, Pilih tab **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Untuk menampilkan histogram, pilih **Histogram**.
- Klik **OK**. Luaran histogram dengan fitur charts diberikan pada Gambar 3.13.

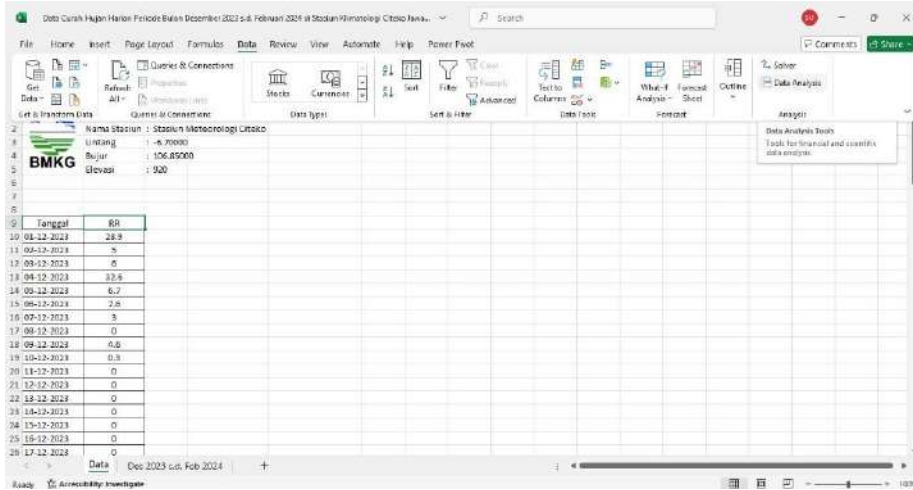


Gambar 3.13. Output Histogram Data Curah Hujan Harian Periode Bulan Desember 2023 sampai dengan Bulan Februari 2024

- b) Langkah-langkah untuk membuat histogram sebaran data curah hujan harian periode bulan Desember 2023 sampai dengan bulan Februari 2024 di Stasiun Klimatologi Jawa Barat melalui fitur Data Analysis adalah sebagai berikut:
- Siapkan data tabel (tersedia pada file Excel di tautan <https://sl.ut.ac.id/data-PVO>) dan tentukan batas-batas atas kelas (*bin*) yang akan digunakan. Prosedur penentuan *bin* dapat dipelajari kembali pada praktikum 3.3 (Tabel Frekuensi)

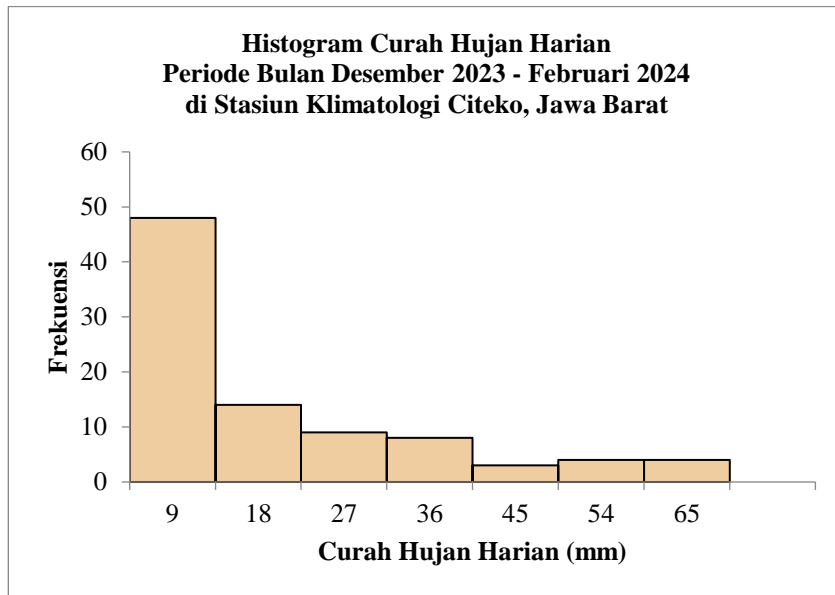


- Klik tab Data, lalu klik Data Analysis (jika tidak tersedia fitur Data Analysis, instalasi tersedia pada bagian A Persiapan Peserta) seperti pada gambar 3.14



Gambar 3.14. *Data Analysis* pada tab Data

- Pada Data Analysis, pilih Histogram, dan klik OK
- Input nilai data pada *Input Range* dan nilai Bin pada *input Bin Range*, lalu Centang *Labels*. Tentukan sel kosong untuk *Output Range* (berupa tabel frekuensi), lalu Centang *Chart Output*, klik OK.
- Selanjutnya, histogram yang dihasilkan diatur lebar batang dengan Format Data Series. Tata cara mengatur lebar batang yaitu dengan klik batang pada histogram, pilih Format Data Series serta ubah Series Overlap dan Gap Width masing-masing menjadi 0%. Lalu pada menu Border, ubah color dengan warna hitam Histogram dapat dilihat pada Gambar 3.15.



Sumber Data : Web BMKG Data Online di <https://dataonline.bmkg.go.id>

Gambar 3.15 Output Histogram Data Curah Hujan Harian Periode Bulan Desember 2023 sampai dengan Bulan Februari 2024

Interpretasi:


Berdasarkan Gambar 3.14 atau 3.15, nilai terendah curah hujan sebesar 0 mm dan nilai tertinggi curah hujan sebesar 64. Selain itu, sebaran data membentuk pola menjulur ke kanan yang mengindikasikan bahwa sebagian besar amatan cenderung bernilai rendah, namun ada sedikit amatan yang bernilai tinggi. Untuk pola ini, nilai rata-rata lebih besar dibandingkan median. Diketahui rata-rata data curah hujan harian tersebut adalah 14,21 mm dan median sebesar 7,55 mm. Berdasarkan data, diketahui tidak ada informasi nilai outlier atau data ekstrem.

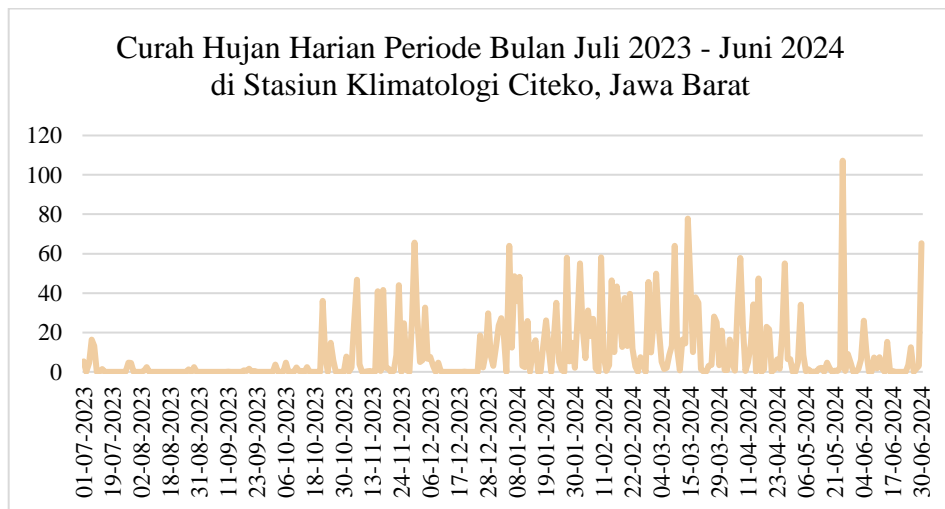
### Praktikum 3.7 (Line Chart dan Scatter Plot Data Tunggal)

Diberikan data curah hujan harian periode bulan Juli 2023 sampai dengan bulan Juni 2024. Data dapat diakses pada <https://sl.ut.ac.id/data-PVO> . Buatlah grafik garis dan plot pencaran (*scatter plot*) serta berikan interpretasinya!

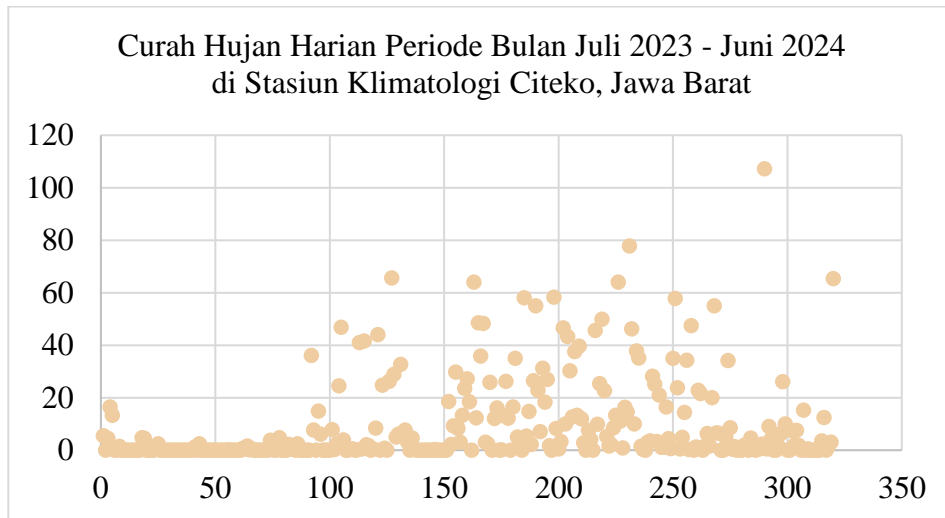
### Jawaban Praktikum 3.7

Langkah-langkah untuk memvisualisasikan grafik garis dan plot pencaran adalah sebagai berikut:

- Blok Tabel data, pilih menu **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Pilih **Line** untuk menampilkan grafik garis atau **X Y (Scatter)** untuk menampilkan plot pencaran
- Klik **OK**. Luaran grafik garis dan plot pencaran diberikan pada gambar 3.16.



(a)



(b)

Gambar 3.16 Luaran (a) grafik garis dan (b) plot pencaran dengan data Curah Hujan Harian Periode Bulan Juli 2023 sampai dengan Bulan Juni 2024 di Stasiun Klimatologi Citeko Jawa Barat

Interpretasi:

Berdasarkan gambar 3.16 (a) dan (b), pola titik-titik curah hujan harian menaik mulai bulan Oktober 2023 dan menurun kembali pada bulan Maret 2024. Selain itu, plot data dan grafik garis menunjukkan bahwa terdapat curah hujan di atas 100 mm per/ hari pada bulan Mei tahun 2024. Data tersebut disebut outlier.

Grafik garis dan plot tebaran dapat digunakan untuk memvisualisasikan beberapa data (jamak) dalam satu grafik atau plot sehingga dapat berperan untuk membandingkan beberapa data dengan kategori yang sama. Perhatikan contoh soal pada praktikum 3.8.

### **Praktikum 3.8 (Line Chart dan Scatter Plot Data Jamak)**

Diberikan data Suku Bunga Bank Kredit Rupiah pada Bank Persero Modal Kerja, Investasi, dan Konsumsi Periode Bulan Agustus 2022 sampai dengan Juli 2024 seperti pada Tabel 3.5 (Data tersedia pada <https://sl.ut.ac.id/data-PVO>).


Buatlah grafik garis dan plot pencaran (*scatter plot*) serta berikan interpretasinya!

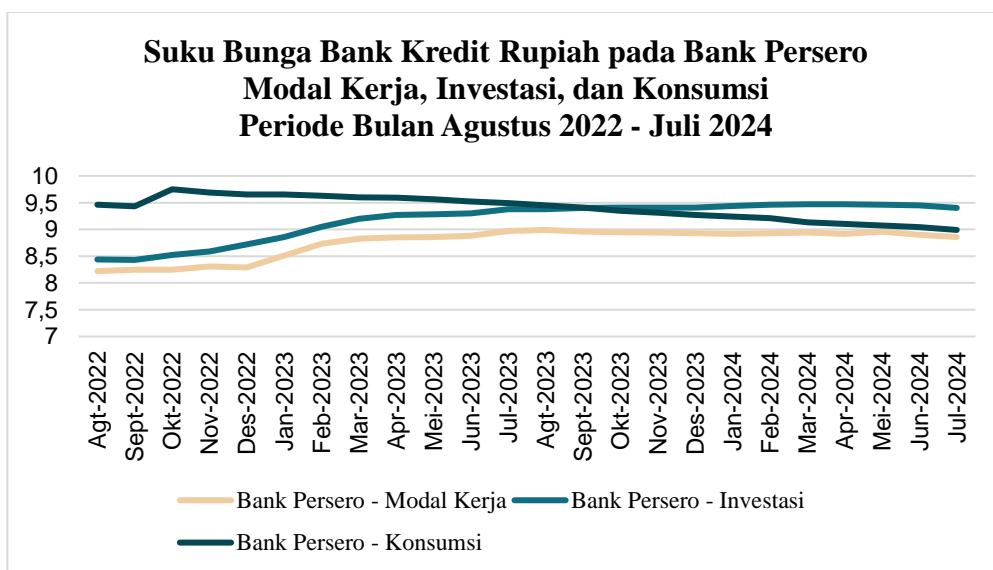
Tabel 3.5 Suku Bunga Bank Kredit Rupiah pada Bank Persero Modal Kerja, Investasi, dan Konsumsi Periode Bulan Agustus 2022 - Juli 2024

No	Waktu	Bank Persero		
		Modal Kerja	Investasi	Konsumsi
1	Agt-2022	8.22	8.44	9.46
2	Sept-2022	8.25	8.43	9.43
3	Okt-2022	8.25	8.52	9.75
4	Nov-2022	8.31	8.59	9.69
5	Des-2022	8.29	8.72	9.65
6	Jan-2023	8.51	8.86	9.65
7	Feb-2023	8.73	9.05	9.63
8	Mar-2023	8.83	9.2	9.6
9	Apr-2023	8.85	9.27	9.59
10	Mei-2023	8.86	9.28	9.56
11	Jun-2023	8.88	9.3	9.52
12	Jul-2023	8.97	9.38	9.49
13	Agt-2023	8.99	9.38	9.45
14	Sept-2023	8.96	9.4	9.41
15	Okt-2023	8.95	9.41	9.35
16	Nov-2023	8.94	9.41	9.31
17	Des-2023	8.93	9.41	9.27
18	Jan-2024	8.92	9.44	9.24
19	Feb-2024	8.93	9.46	9.21
20	Mar-2024	8.94	9.47	9.13
21	Apr-2024	8.92	9.47	9.1
22	Mei-2024	8.96	9.46	9.07

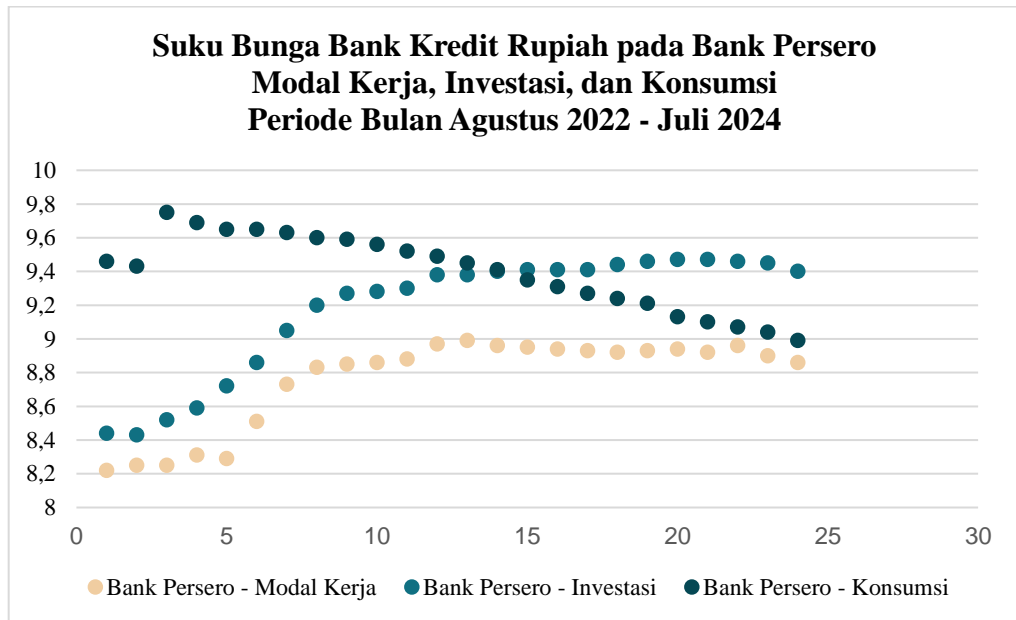
23	Jun-2024	8.9	9.45	9.04
24	Jul-2024	8.86	9.4	8.99

Langkah-langkah untuk memvisualisasikan grafik garis dan plot pencarian adalah sebagai berikut:

- Blok Tabel data Suku Bunga Bank Kredit Rupiah pada Bank Persero Modal Kerja, Investasi, dan Konsumsi Periode Bulan Agustus 2019 sampai dengan Juli 2024,
- pilih menu **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Pilih **Line** untuk menampilkan grafik garis atau **X Y (Scatter)** untuk menampilkan plot pencarian
- Klik **OK**. Luaran grafik garis dan plot pencarian diberikan pada gambar 3.17.



(a)



(b)

Gambar 3.17 Luaran (a) grafik garis dan (b) plot pencaran dengan data Suku Bunga Bank Kredit Rupiah pada Bank Persero Modal Kerja, Investasi, dan Konsumsi Periode Bulan Agustus 2019 sampai dengan Juli 2024

Interpretasi:

Berdasarkan Gambar 3.17 (a) dan (b) diketahui bahwa garis atau plot berwarna abu-abu suku bunga bank kredit rupiah pada bank persero konsumsi mengalami penurunan dari waktu ke waktu pada periode bulan Agustus 2019 sampai dengan Juli 2024 sedangkan garis atau plot berwarna biru dan oren masing-masing suku bunga bank kredit rupiah pada bank persero modal kerja dan investasi mengalami fluktuasi dan relatif menaik dari waktu ke waktu pada periode bulan Agustus 2019 sampai dengan Juli 2024.


**Praktikum 3.9 (Pie Chart dan Diagram Donat)**

Grafik lingkaran (*pie chart*) atau diagram donat umum digunakan untuk memvisualisasikan proporsi sederhana. Contoh soal yang berkaitan dengan grafik lingkaran dan diagram donat diberikan berikut ini.

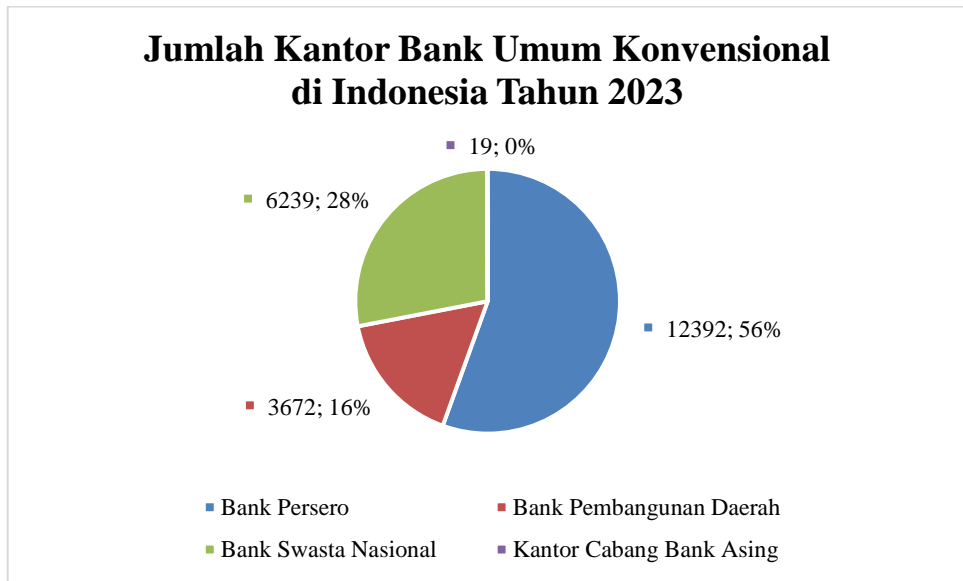
Data Jumlah kantor Bank Umum Konvensional di Indonesia Tahun 2023 dapat diakses pada <https://sl.ut.ac.id/data-PVO>. Buatlah grafik lingkaran dan diagram donat serta berikan interpretasinya!

### Jawaban Praktikum 3.9

Langkah-langkah untuk memvisualisasikan grafik lingkaran dan diagram donat adalah sebagai berikut:

- Impor dan organisir data Jumlah Kantor Bank Umum Konvensional di Indonesia Tahun 2023 ke dalam suatu tabel pada Sheet Excel.
- Blok Tabel data, Pilih menu **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Pilih **Pie** untuk menampilkan diagram/ grafik lingkaran atau pilih **Doughnut** untuk menampilkan diagram donat.
- Pilih diagram yang sesuai
- Klik **OK**.
- Untuk menampilkan nilai jumlah kantor bank umum konvensional di Indonesia tahun 2023 dengan klik sembarang pada diagram lingkaran/ diagram donat, lalu klik **Add Data Labels**. Untuk menampilkan persentase atau pengaturan lainnya, dapat klik 2 kali pada nilai atau klik kanan nilai, lalu pilih **Format Data Label**. Selanjutnya pengaturan dapat disesuaikan dengan kebutuhan. Luaran grafik lingkaran dan diagram donat diberikan pada gambar 3.18





(a)



Gambar 3.18 Luaran (a) grafik lingkaran dan (b) diagram donat dengan data Jumlah Kantor Bank Umum Konvensional di Indonesia Tahun 2023

Interpretasi:

Berdasarkan Gambar 2.14, dapat dilihat bahwa jumlah bank persero (56%) mendominasi jumlah bank pembangunan daerah, bank swasta nasional, dan kantor cabang bank asing pada jumlah kantor bank umum konvensional di Indonesia tahun 2023.

### **Praktikum 3.10 (Boxplot)**

*Boxplot* dapat digunakan untuk menggambarkan sebaran data. *Boxplot* terdiri atas dua bagian yaitu sebuah kotak mendatar atau menaik dan garis ke arah kanan dan ke kiri atau atas dan bawah dari kotak tersebut sehingga *boxplot* dikenal dengan *diagram kotak garis*. Umumnya, sebelum mengilustrasikan data dengan *boxplot*, data terlebih dahulu diidentifikasi beberapa statistik ringkasan dan menggunakan nilai-nilai statistik ringkasan untuk ditampilkan pada grafik. Statistik yang perlu diidentifikasi yaitu kuartil pertama ( $Q_1$ ), kuartil kedua atau median ( $Q_2$ ), dan kuartil ketiga ( $Q_3$ ). Ketiga nilai tersebut ditampilkan pada *boxplot*. Selain itu, *boxplot* juga dapat memvisualisasikan adanya titik-titik amatan pada data yang merupakan pencilan (*outlier*). Amatan yang disebut pencilan adalah amatan yang nilainya berada di luar selang antar pagar bawah dan pagar atas. Formula pagar bawah dan pagar atas diberikan sebagai berikut:

$$\text{Pagar Bawah} = Q_1 - 1,5 (Q_3 - Q_1)$$


$$\text{Pagar Atas} = Q_3 + 1,5 (Q_3 - Q_1)$$

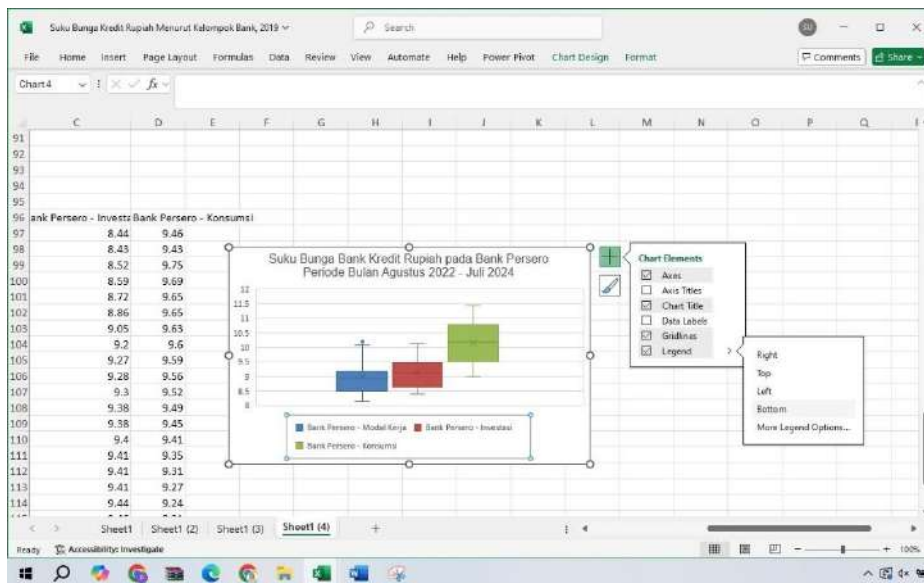
Contoh soal *boxplot* tersedia berikut ini.

Diberikan data Suku Bunga Bank Kredit Rupiah pada Bank Persero Modal Kerja, Investasi, dan Konsumsi Periode Bulan Agustus 2019 sampai dengan Juli 2024 seperti pada Tabel 3.5. Buatlah *boxplot* dari data tersebut dan berikan interpretasinya!

### **Jawaban Praktikum 3.10**

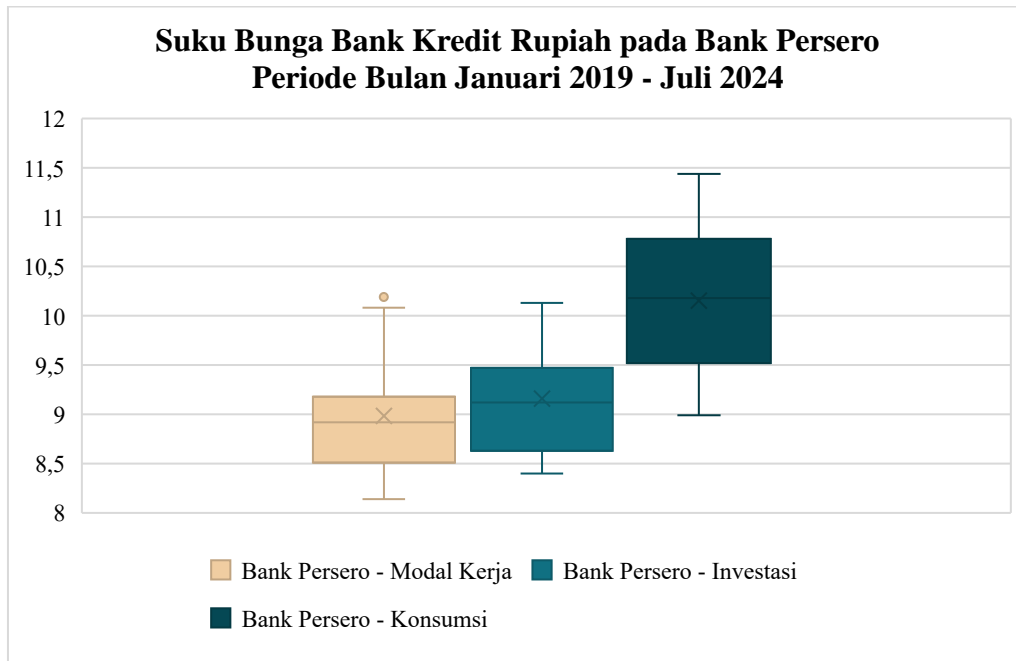
Langkah-langkah untuk memvisualisasikan data pada Tabel 3.5 dengan *boxplot* diberikan sebagai berikut:

- Blok Tabel data, Pilih menu **Insert**. Klik **Chart** (pojok kanan bawah )
- Pilih **All Charts**
- Pilih **Box and Whisker** untuk menampilkan box-plot
- Klik **OK**. Luan *boxplot* dihasilkan.
- Untuk mendapatkan visualisasi yang menarik, klik sumbu x atau lainnya yang belum sesuai, lalu klik kanan dan pilih Delete. Lalu, Anda dapat menambahkan judul dan Legend. Tata cara menambahkan Legend yaitu klik Chart Area (luaran *boxplot*), klik tanda plus (*Chart Elements*) di sisi kiri atau kanan Chart Area, beri centang pada Legend, dan pilih Buttom. Perhatikan Gambar 3.19 berikut ini.



Gambar 3.19 Menu *Chart Elements*

- Selesai. Luan *boxplot* diberikan pada gambar 3.20.



Gambar 3.20 Luaran *boxplot* dengan data Suku Bunga Bank Kredit Rupiah pada Bank Persero Modal Kerja, Investasi, dan Konsumsi Periode Bulan Januari 2019 sampai dengan Juli 2024

Interpretasi:

Berdasarkan Gambar 3.20, garis tengah masing-masing kotak menggambarkan median masing-masing data. Pada kotak berwarna biru yang menggambarkan suku bunga kredit rupiah pada bank persero modal kerja, terdapat titik amatan yang merupakan pencilan atau *outlier* karena titik tersebut berada di luar pagar atas data. Dari tiga suku bunga bank persero, suku bunga bank persero konsumsi lebih tinggi dibandingkan suku bunga bank persero modal kerja dan konsumsi.

### Praktikum 3.11 (*Density Plot*)

- Data
- Langkah-langkah
- Output

**Praktikum 3.12 (QQ *Plot*)**

- Data
- Langkah-langkah
- Output

## H. PRAKTIKUM MASALAH TRANSPORTASI

Masalah transportasi menjadi bentuk khusus model pemrograman linear yang termasuk dalam masalah pengambilan keputusan. Model transportasi termasuk dalam model jaringan, model yang berkaitan dengan pendistribusian barang dan jasa dari suatu lokasi (titik pasokan) ke lokasi lainnya (titik permintaan), dengan tujuan menentukan banyaknya variabel keputusan (barang dan jasa) agar dapat memaksimalkan keuntungan atau meminimalkan biaya pengiriman (transportasi). Untuk menyelesaikan masalah transportasi, Anda dapat membaca asumsi standar dan variasi masalah pada bahan ajar MATA4303 Riset Operasi, Kegiatan Belajar 1.

### Praktikum 4.3

Sebuah perusahaan tekstil mempunyai tiga pabrik yang berlokasi di Bandung, Pekalongan, dan Solo. Setiap pabrik mempunyai kemampuan dalam memproduksi jenis bahan katun dalam satu kali produksi. Ketiga pabrik tersebut akan memasok bahan katun untuk kebutuhan garmen yang membuat seragam sekolah di empat lokasi yaitu Bandung, Tangerang, Bekasi, dan Bogor. Adapun tabel parameter dan formulasi (model aljabar) masalah transportasi pengiriman bahan katun diberikan pada tabel berikut.

**Tabel Parameter**

Lokasi pabrik	Biaya pengiriman ke lokasi garmen (Rp)				Pasokan (meter)
	Bandung	Tangerang	Bekasi	Bogor	
Bandung	250	475	400	450	10000
Pekalongan	450	550	500	600	11500
Solo	550	650	700	800	8500
Permintaan (meter)	8000	7900	6600	7500	

### Model Aljabar

$i$  = lokasi pabrik;  $i = 1, 2, 3$ .

$j$  = lokasi garmen;  $j = 1, 2, 3, 4$ .

$x_{ij}$  = banyaknya kain yang dikirim dari pabrik  $i$  ke garmen  $j$  (meter)

- Fungsi Objektif

$$\min Z = 250x_{11} + 475x_{12} + 400x_{13} + 450x_{14} + 450x_{21} + 550x_{22} + 500x_{23} + 600x_{24} + 550x_{31} + 650x_{32} + 700x_{33} + 800x_{34}.$$

- Kendala

Kapasitas lokasi pabrik

$$x_{11} + x_{12} + x_{13} + x_{14} = 10000$$

$$x_{21} + x_{22} + x_{23} + x_{24} = 11500$$

$$x_{31} + x_{32} + x_{33} + x_{34} = 8500$$

Permintaan dari lokasi garmen

$$x_{11} + x_{21} + x_{31} = 8000$$

$$x_{12} + x_{22} + x_{32} = 7900$$

$$x_{13} + x_{23} + x_{33} = 6600$$

$$x_{14} + x_{24} + x_{34} = 7500$$

Ketaknegatifan

$$x_{11}, x_{12}, x_{13}, x_{14}, x_{21}, x_{22}, x_{23}, x_{24}, x_{31}, x_{32}, x_{33}, x_{34} \geq 0$$

Praktikum dilakukan dengan langkah sebagai berikut.

### Menyiapkan Spreadsheet Masalah Transportasi

1. Ketik pada suatu sel teks “Tabel Parameters”, kemudian di bawahnya ketik “ $c_{ij}$ ” yang menyatakan biaya pengiriman dari suatu lokasi pabrik ke suatu lokasi garmen.

Secreenshot ...

	A	B	C	D	E	F	G	H
1								
2	Tabel Parameter							
3	c_ij							
4								
5								
6								

2. Ketik teks lokasi pabrik “Bandung”, “Pekalongan”, dan “Solo” pada sel di bawah teks “ $c_{ij}$ ” (berurutan dari atas ke bawah), kemudian pada

sel di sebelah kanannya ketik teks lokasi garmen “Bandung”, “Tangerang”, “Bekasi”, dan “Bogor” (berurutan dari kiri ke kanan).

Secreenshot ...

	A	B	C	D	E	F	G	H
1								
2	Tabel Parameter							
3	c_ij	Bandung	Tangerang	Bekasi	Bogor			
4	Bandung							
5	Pekalongan							
6	Solo							
7								

- Ketik biaya pengiriman dari suatu lokasi pabrik ke suatu lokasi garmen pada sel yang sesuai seperti yang tercantum pada tabel parameter. Dimulai dari biaya pengiriman dari lokasi pabrik di “Bandung” ke lokasi garmen di “Bandung” sebesar 250 hingga biaya pengiriman dari lokasi pabrik di “Solo” ke lokasi garmen di “Bogor” sebesar 800.

Secreenshot ...

	A	B	C	D	E	F	G	H	I
1									
2	Tabel Parameter								
3	c_ij	Bandung	Tangerang	Bekasi	Bogor				
4	Bandung	250	475	400	450				
5	Pekalongan	450	550	500	600				
6	Solo	550	650	700	800				
7									

- Ketik teks “Permintaan” di bawah sel lokasi pabrik yang terakhir (Solo), kemudian ketik jumlah permintaan setiap garmen yaitu 8000, 7900, 6600, dan 7500 (berurutan dari kiri ke kanan). Dilanjutkan dengan mengetik teks “Pasokan” sebelah kanan sel lokasi garmen yang terakhir (Bogor), kemudian ketik jumlah pasokan setiap lokasi pabrik yaitu 10000, 11500, dan 8500 (berurutan dari atas ke bawah).

Secreenshot ...

Tabel Parameter						
c_ij	Bandung	Tangerang	Bekasi	Bogor	Pasokan	
Bandung	250	475	400	450	10000	
Pekalongan	450	550	500	600	11500	
Solo	550	650	700	800	8500	
Permintaan	8000	7900	6600	7500		



5. Setelah memindahkan data tabel parameter pada spreadsheet, dilanjutkan dengan mengisi variabel keputusan.

Ketik teks “Variabel Keputusan” pada sel di bawah sel yang berisi teks “Permintaan” (berikan jarak satu baris untuk memisahkan data “Tabel Parameter” dengan “Variabel Keputusan”), kemudian di bawahnya ketik “ $x_{ij}$ ” yang menyatakan banyaknya kain yang dikirim dari pabrik  $i$  ke garmen  $j$ .

Secreenshot ...

	A	B	C	D	E	F	G	H
1								
2	Tabel Parameter							
3	c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan		
4	Bandung	250	475	400	450	10000		
5	Pekalongan	450	550	500	600	11500		
6	Solo	550	650	700	800	8500		
7	Permintaan	8000	7900	6600	7500			
8								
9	Variabel Keputusan							
10	x <sub>ij</sub>							

6. Ketik teks lokasi pabrik “Bandung”, “Pekalongan”, dan “Solo” pada sel di bawah teks “ $x_{ij}$ ” (berurutan dari atas ke bawah), kemudian pada sel di sebelah kanannya ketik teks lokasi garmen “Bandung”, “Tangerang”, “Bekasi”, dan “Bogor” (berurutan dari kiri ke kanan).

Secreenshot ...

	A	B	C	D	E	F	G	H	I
1									
2	Tabel Parameter								
3	c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan			
4	Bandung	250	475	400	450	10000			
5	Pekalongan	450	550	500	600	11500			
6	Solo	550	650	700	800	8500			
7	Permintaan	8000	7900	6600	7500				
8									
9	Variabel Keputusan								
10	x <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor				
11	Bandung								
12	Pekalongan								
13	Solo								
14									

7. Ketik nilai (sebarang) variabel keputusan pada sel yang menghubungkan masing-masing suatu lokasi pabrik ke masing-masing lokasi garmen, pada praktikum ini ambil nilai “1”. Dimulai dari



9. Ketik teks “SUM” di sebelah kanan sel lokasi garmen yang paling kanan (Bogor), kemudian pada sel sebelah bawahnya lakukan formula menjumlahkan banyaknya kain yang dikirim pabrik ke setiap garmen yang dimulai dari pabrik “Bandung” dari pabrik “Bandung”, “Tangerang”, “Bekasi”, dan “Bogor” dimana masing-masing bernilai “1” (dari kiri ke kanan). Kemudian dilanjutkan hitungan formulasi dari pabrik “Pekalongan”, dan “Solo”.

Secreenshot ...

	A	B	C	D	E	F	G	H	I
1									
2	Tabel Parameter								
3	c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan			
4	Bandung	250	475	400	450	10000			
5	Pekalongan	450	550	500	600	11500			
6	Solo	550	650	700	800	8500			
7	Permintaan	8000	7900	6600	7500				
8									
9	Variabel Keputusan								
10	x <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	SUM			
11	Bandung	1	1	1	1	4			
12	Pekalongan	1	1	1	1	4			
13	Solo	1	1	1	1	4			
14									
15	SUM	3	3	3	3				

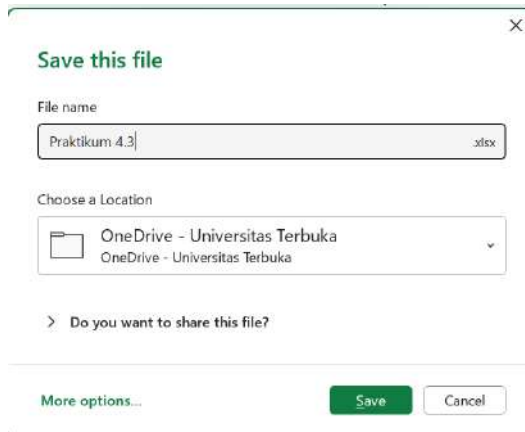
10. Ketik teks “Fungsi Objektif” di bawah sel yang berisi teks “SUM” pada langkah nomor 8. (berikan jarak satu baris), kemudian menyetik teks “Z” (fungsi objektif) pada sel di bawahnya. Dilanjutkan pada sel disebelah kanannya, lakukan formula penjumlahan dari perkalian setiap biaya pengiriman dengan setiap variabel keputusan (bernilai 1) dari lokasi pabrik ke lokasi garmen.

Secreenshot ...

	A	B	C	D	E	F	G	H	I
2	Tabel Parameter								
3	c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan			
4	Bandung	250	475	400	450	10000			
5	Pekalongan	450	550	500	600	11500			
6	Solo	550	650	700	800	8500			
7	Permintaan	8000	7900	6600	7500				
8									
9	Variabel Keputusan								
10	x <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	SUM			
11	Bandung	1	1	1	1	4			
12	Pekalongan	1	1	1	1	4			
13	Solo	1	1	1	1	4			
14									
15	SUM	3	3	3	3				
16									
17	Fungsi Objektif								
18		Z	12						

11. Save spreadsheet dengan nama file **Praktikum 4.3.xlsx**.

**Screenshot ...**



## Penyelesaian Masalah Transportasi

1. Dari file **Praktikum 4.3.xlsx**, klik icon Solver untuk menjalankan.

**Screenshot ....**

The screenshot shows the Microsoft Excel spreadsheet with the following data:

Tabel Parameter					
c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan
Bandung	250	475	400	450	10000
Pekalongan	450	550	500	600	11500
Solo	550	650	700	800	8500
Permintaan	8000	7900	6600	7500	

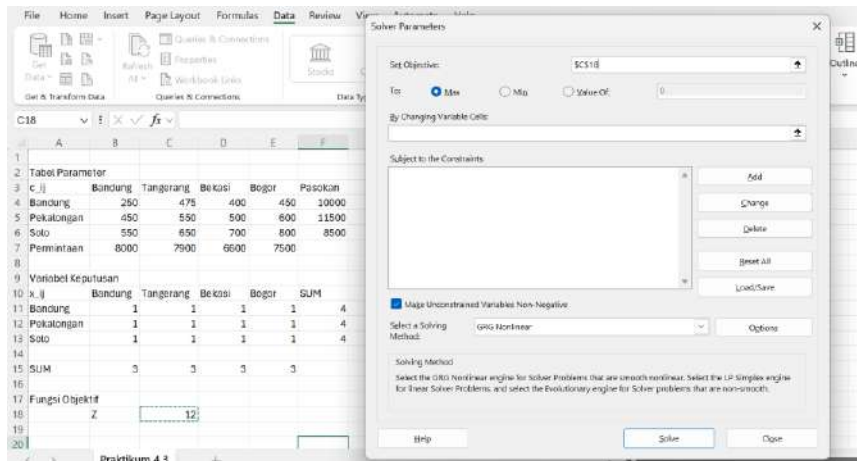
Variabel Keputusan					
x <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	SUM
Bandung	1	1	1	1	4
Pekalongan	1	1	1	1	4
Solo	1	1	1	1	4
SUM	3	3	3	3	

Fungsi Objektif: Z = 12

2. Klik kotak pada sisi teks “Set Objective”, kemudian klik nilai “Z” (fungsi objektif) pada spreadsheet yaitu nilai hasil penjumlahan dari perkalian setiap biaya pengiriman dengan setiap variabel keputusan (nilai 1) dari lokasi pabrik ke lokasi garmen. (dari langkah nomor 10. dalam menyiapkan spreadsheet).

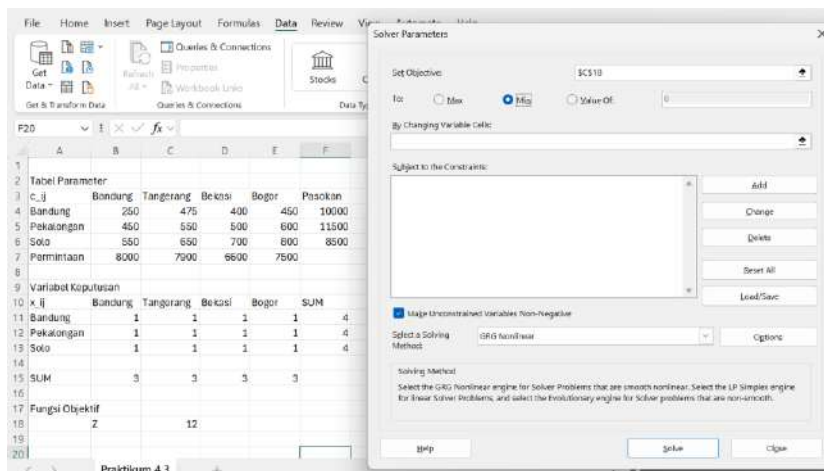
Sebelumnya perhatikan kotak Set Objektive harus tidak ada isinya (kosong), jika ada isinya maka dihapus kemudian lakukan langkah 2. ini.

**Screenshot ....**



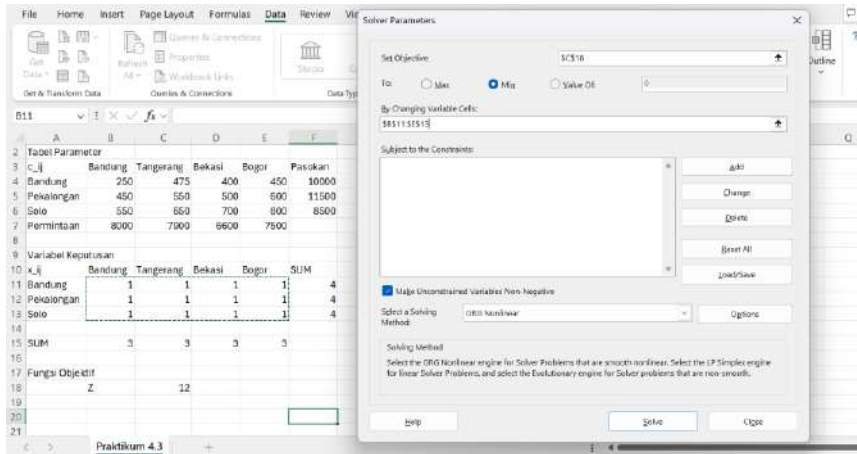
- Pilih “Min” sisi teks “To” sesuai dengan tujuan fungsi objektif masalah Transportasi.

**Screenshot ....**



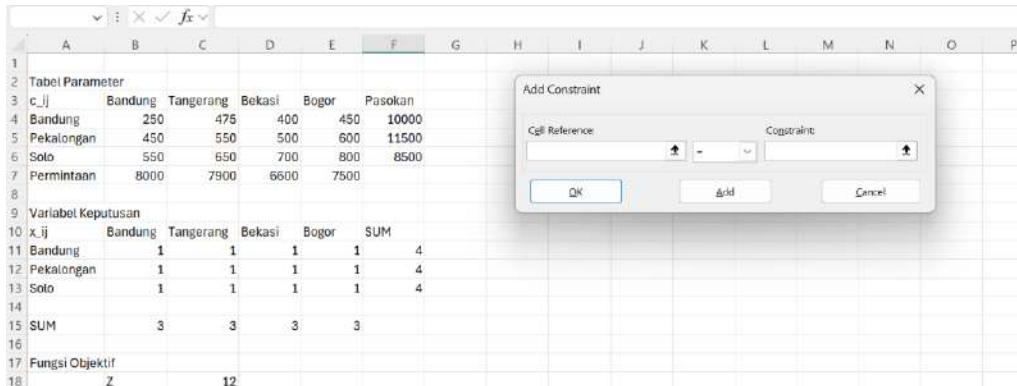
- Klik kotak pada sisi teks “By Changing Variabel Cell”, kemudian klik (blok) sel yang berisi variabel keputusan (nilai 1) yang menghubungkan lokasi pabrik ke lokasi garmen! (dari langkah nomor 7. dalam menyiapkan spreadsheet)

**Screenshot ....**



5. Klik “Add” yang terdapat di sebelah kanan “Subject to the Constrains” untuk menambahkan fungsi kendala pada Solver, maka akan muncul jendela “Add Constraint”.

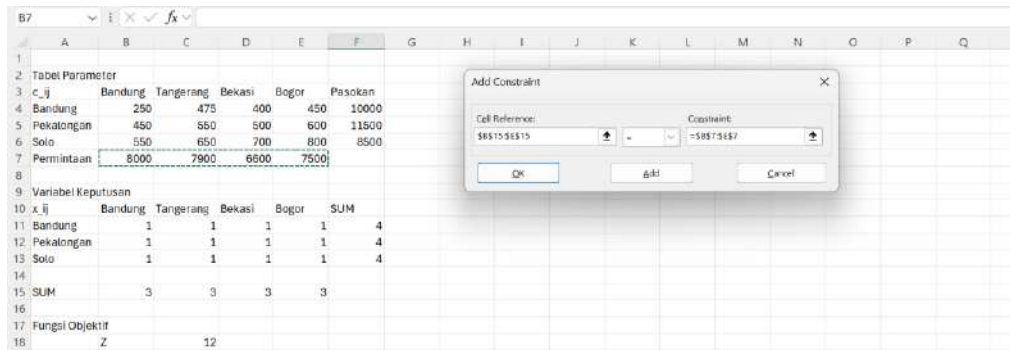
Screenshot ....



6. Pada model masalah Transportasi, ada dua kendala yaitu permintaan lokasi garmen dan kapasitas lokasi pabrik dimana masing-masing mempunyai tiga fungsi kendala. Bentuk karakter yang menghubungkan ruas kiri dengan ruas kanan fungsi kendala sama yaitu “sama dengan” (=), sehingga untuk menambahkan tiga fungsi setiap kendala dapat dilakukan bersamaan.

Perhatikan spreadsheet, klik (blok) sel pada variabel keputusan yang berisi jumlah (sum) permintaan lokasi garmen, kemudian memilih karakter “=” di kotak tengah, kemudian klik (blok) sel yang berisi jumlah permintaan lokasi garmen (Bandung, Tangerang, Bekasi, dan Bogor).

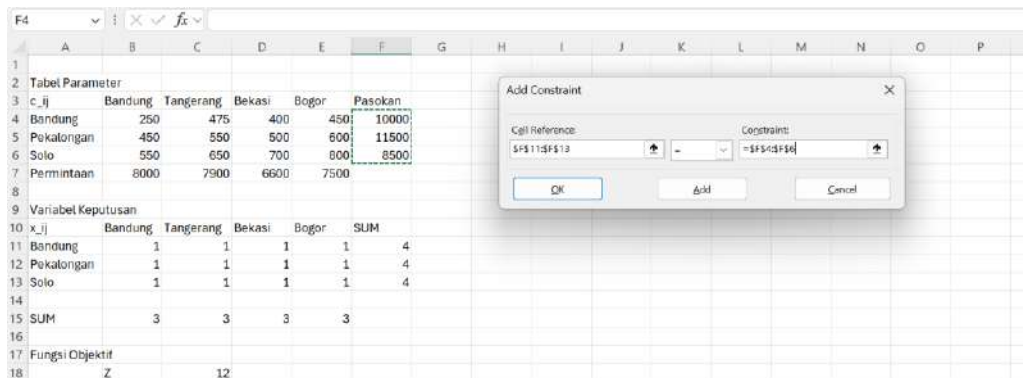
Screenshot ....



Tabel Parameter					
c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan
Bandung	250	475	400	450	10000
Pekalongan	450	550	500	600	11500
Solo	550	650	700	800	8500
Permintaan	8000	7900	6600	7500	

7. Kemudian klik (blok) sel pada variabel keputusan yang berisi jumlah (sum) kapasitas lokasi pabrik, kemudian memilih karakter “=” di kotak tengah, kemudian klik (blok) sel yang berisi jumlah kapasitas lokasi pabrik (Bandung, Pekalongan, dan Solo).

Screenshot ....

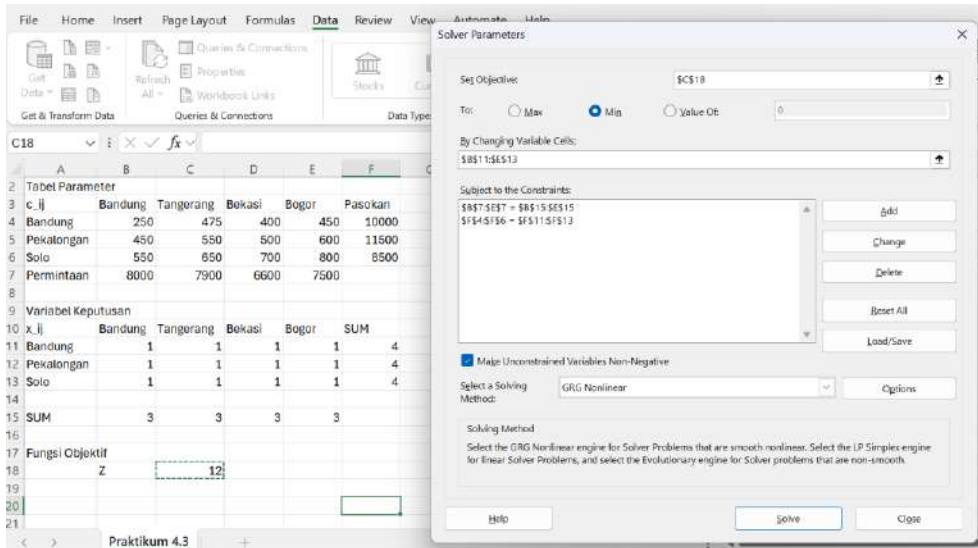


Variabel Keputusan					
x <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	SUM
Bandung	1	1	1	1	4
Pekalongan	1	1	1	1	4
Solo	1	1	1	1	4
SUM	3	3	3	3	

8. Untuk kendala ketaknegatifan tidak diketik pada “Subject to the Constrains” tetapi mencentang (√) Make Unconstrained Variabels Non-Negative.

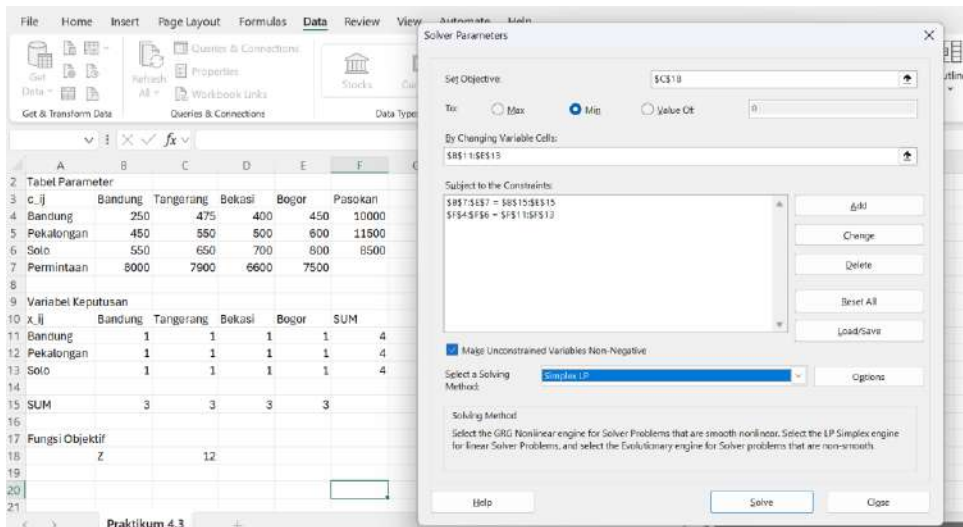
Screenshot ....





9. Klik tanda panah ke bawah (v) pada kotak “Select a Solving Method”, kemudian pilih Simpleks LP karena fungsi objektif dan fungsi kendala masalah PL berbentuk linear.

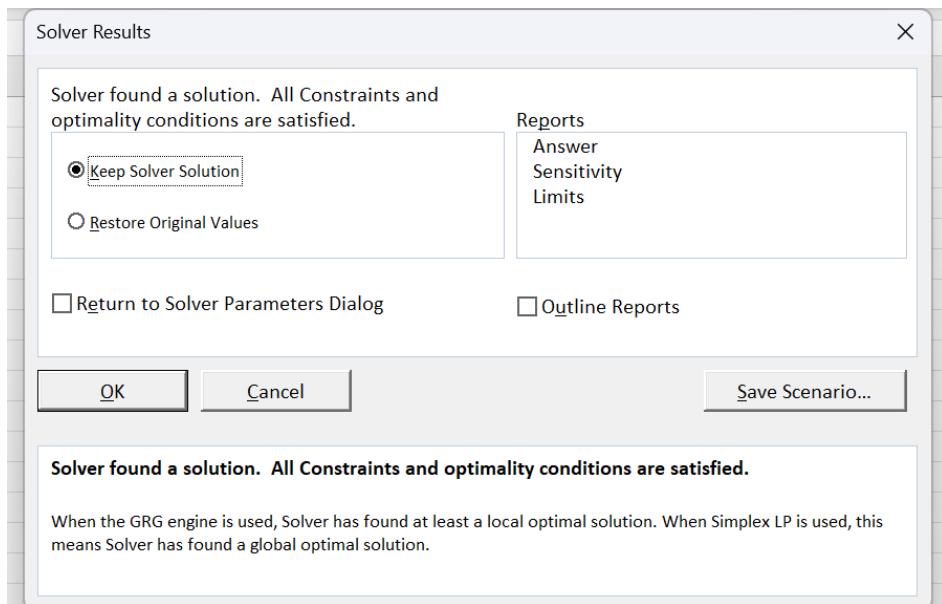
Screenshot ....



10. Jika semua komponen yang terdapat pada jendela “Solver parameters” sudah terisi dengan benar, maka klik “Solve”. Solver bekerja selama waktu yang dibutuhkan, kemudian muncul jendela “Solver Result” dan dibagian bawah ada tulisan yang menyatakan Solver memperoleh solusi optimum yaitu “Solver found a solution. All Constraints and optimality conditions are satisfied”.

Screenshot ....





11. Pilih Keep Solver Solution, kemudian klik OK. Anda akan melihat model spreadsheet yang baru dimana nilai beberapa sel berubah dari model spreadsheet yang disiapkan diawal praktikum.

### Screenshot

	A	B	C	D	E	F	G
1							
2	Tabel Parameter						
3	c <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	Pasokan	
4	Bandung	250	475	400	450	10000	
5	Pekalongan	450	550	500	600	11500	
6	Solo	550	650	700	800	8500	
7	Permintaan	8000	7900	6600	7500		
8							
9	Variabel Keputusan						
10	x <sub>ij</sub>	Bandung	Tangerang	Bekasi	Bogor	SUM	
11	Bandung	0	7900	2100	0	10000	
12	Pekalongan	8000	0	0	3500	11500	
13	Solo	0	0	4500	4000	8500	
14							
15	SUM	8000	7900	6600	7500		
16							
17	Fungsi Objektif						
18	Z		30000				
19							

Tuliskan perubahan-perubahan yang dapat dilihat pada model spreadsheet yang terakhir/baru!

- Sebelumnya semua variabel Keputusan diisi dengan nilai awal 1, setelah solver dijalankan, nilai variabel keputusan diperbarui untuk memenuhi semua kendala.
- Pada baris SUM, solver memastikan jumlah pengiriman ke setiap garmen memenuhi permintaan.
- Pada kolom SUM, solver memastikan jumlah pengiriman dari setiap pabrik tidak melebihi kapasitas pasokan.
- Nilai fungsi objektif (Z) berubah setelah solver dijalankan dan menjadi nilai minimum.

.....

1. Pelaksanaan

....

2. Pelaporan

.....

## I. PRAKTIKUM PEMROGRAMAN TAK LINEAR

Masalah pemrograman taklinear (PTL) adalah model optimasi yang mensyaratkan terdapat paling sedikit satu dari fungsi objektif ataupun fungsi kendala berupa fungsi taklinear. Praktikum PTL di dalam modul ini meliputi PTL dengan satu variabel dan PTL dengan banyak variabel. Selain itu, metode penyelesaian yang dapat digunakan pada PTL adalah metode GRG Nonlinear dan metode Evolutionary. Namun, PTL yang dibahas pada praktikum ini dibatasi pada fungsi smooth, sehingga metode yang akan digunakan cukup dengan GRG Nonlinear. GRG adalah singkatan dari “Generalized Reduced Gradient”. Metode penyelesaian ini melihat gradien atau kemiringan fungsi objektif saat nilai peubah keputusan berubah dan menentukan bahwa metode tersebut telah mencapai solusi optimal saat turunan parsial sama dengan nol.

### Praktikum 4.1

*Menyelesaikan masalah PTL tak berkendala dengan satu variabel menggunakan Solver berdasarkan contoh 5.11 di modul 5 BMP MATA4303 Riset Operasi.*

#### Contoh Kasus:

Pak Koko memiliki seekor sapi. Untuk membiayai sekolah anaknya, ia merencanakan untuk menjual sapi yang saat ini berbobot 100 kilogram. Jika Pak Koko menunda penjualan sapi, maka bobot sapi tersebut bertambah 1 kilogram per hari, tetapi ia harus menanggung biaya pemeliharaan Rp3.000,00 per hari. Saat ini harga 1 kilogram sapi adalah Rp60.000,00 dan harga ini turun Rp500,00 per hari. Jika ingin diketahui berapa hari Pak Koko harus menunggu menjual sapi agar keuntungan yang diperoleh maksimum, maka dapat diformulasikan suatu pemrograman taklinear yang menjawab masalah tersebut.

#### Model PTL:

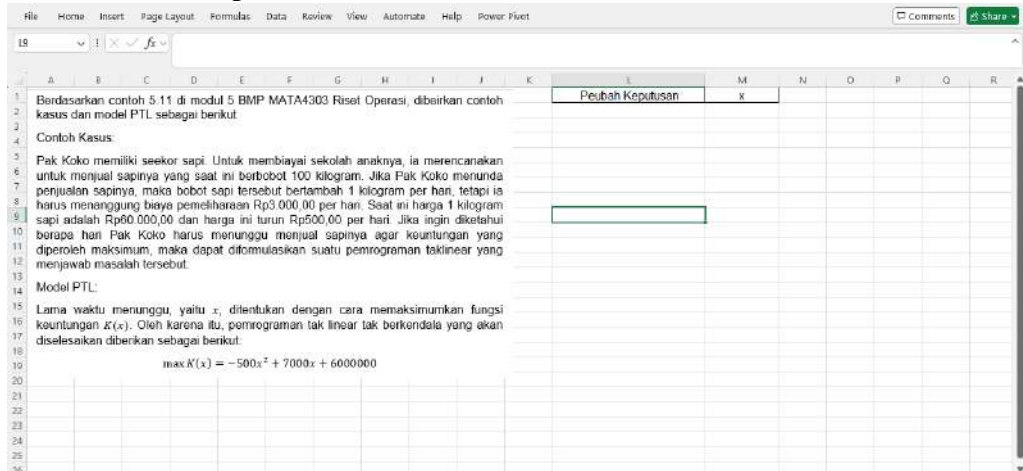
Lama waktu menunggu, yaitu  $x$ , ditentukan dengan cara memaksimalkan fungsi keuntungan  $K(x)$ . Oleh karena itu, pemrograman tak linear tak berkendala yang akan diselesaikan diberikan sebagai berikut:

$$\max K(x) = -500x^2 + 7000x + 6000000$$

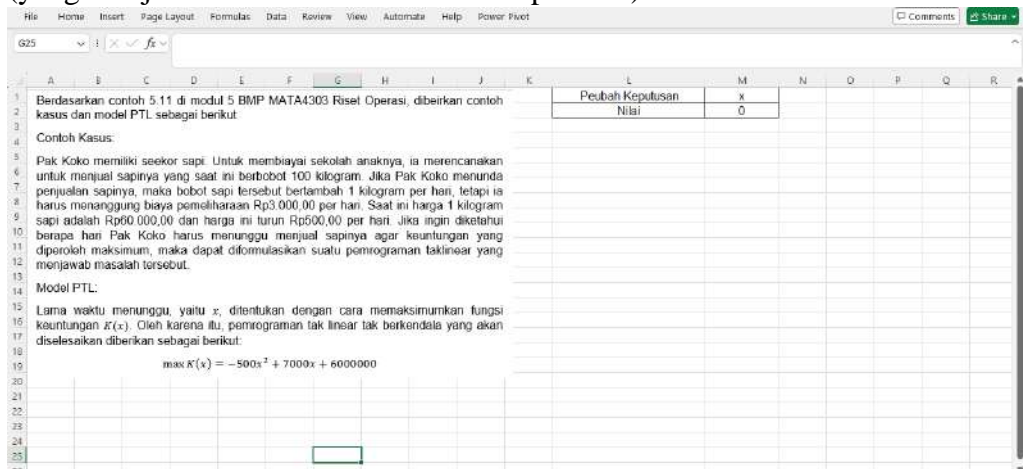
Praktikum dilakukan dengan langkah sebagai berikut.

- **Persiapan spreadsheet masalah PTL**

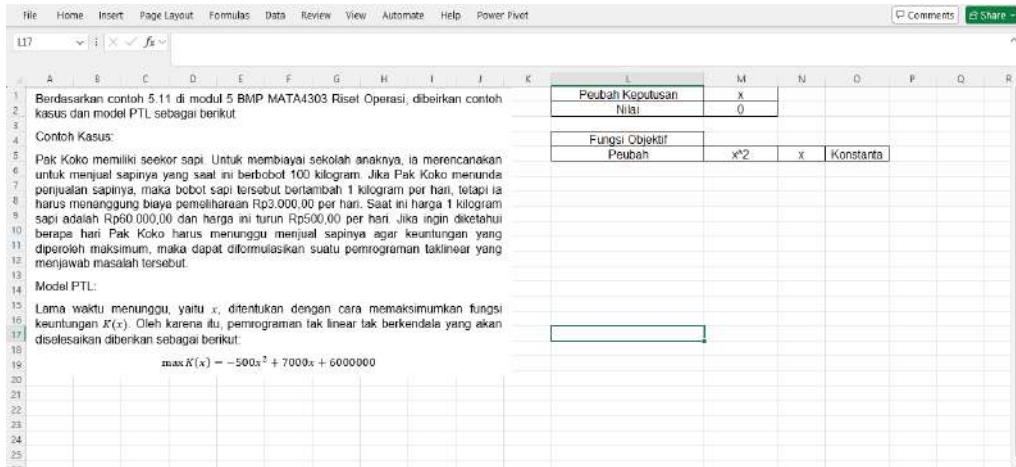
12. Ketikkan pada suatu sel (misal sel L1 pada spreadsheet) dengan teks “Peubah Keputusan”, kemudian pada sel di sebelah kanannya (sel M1) ketikkan notasi variabel keputusan  $x$ .



13. Ketikkan teks “Nilai” pada sel dibawah teks “Peubah Keputusan” (sel L2). Kemudian pada sel sebelah kanannya (sel M2) ketikkan bilangan sebarang (yang menjadi nilai awal dari variabel keputusan)

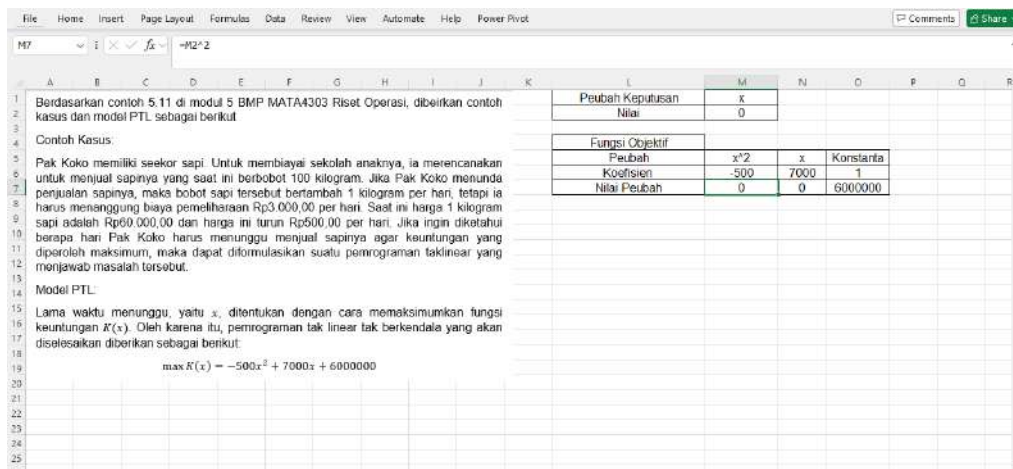


14. Pada umumnya, fungsi objektif PTL merupakan kombinasi beberapa suku. Oleh karena itu, ketikkan teks “Fungsi Objektif” pada sel kosong sebarang (sel L4). Ketikkan teks “Peubah” pada sel kosong sebarang (sel L5), kemudian input peubah-peubah pada fungsi objektif PTL di sebelah kanannya (sel M5 dan seterusnya). Jika terdapat suku konstanta, ketikkan teks “Konstanta” di sebelah kanan sel “Peubah”.



15. Ketikkan teks “Koefisien” pada sel kosong sebarang (misal sel L6), kemudian input koefisien tiap suku pada sel kanannya. Jika terdapat suku konstanta, isi koefisien dengan 1. Selanjutnya, ketikkan teks “Nilai” pada sel kosong (sel L7) di bawah sel Koefisien, kemudian ketikkan nilai peubah tiap masing-masing suku terhadap peubah keputusan dengan fungsi-fungsi atau nilai sebagai berikut:

- pada M7, ketikkan fungsi :  $=M2^2$ ,
- pada N7, ketikkan fungsi :  $=M2$ , dan
- pada O7, ketikkan 6000000.



16. Ketikkan Fungsi objektif PTL yang akan dimaksimalkan, misal teks “Keuntungan” pada sel kosong sebarang (sel L8). Lalu, ketikkan nilai Keuntungan pada M8 dengan fungsi sebagai berikut:

$$= \text{SUMPRODUCT}(M6:O6, M7:O7)$$

Berdasarkan contoh 5.11 di modul 5 BMP MATA4303 Riset Operasi, diberikan contoh kasus dan model PTL sebagai berikut

Contoh Kasus:

Pak Koko memiliki seekor sapi. Untuk membiayai sekolah anaknya, ia merencanakan untuk menjual sapi yang saat ini berbobot 100 kilogram. Jika Pak Koko menunda penjualan sapi, maka bobot sapi tersebut bertambah 1 kilogram per hari, tetapi ia harus menanggung biaya pemeliharaan Rp3.000,00 per hari. Saat ini harga 1 kilogram sapi adalah Rp600.000,00 dan harga ini turun Rp500,00 per hari. Jika ingin diketahui berapa hari Pak Koko harus menunggu menjual sapi agar keuntungan yang diperoleh maksimum, maka dapat diformulasikan suatu pemrograman taklinear yang menjawab masalah tersebut.

Model PTL:

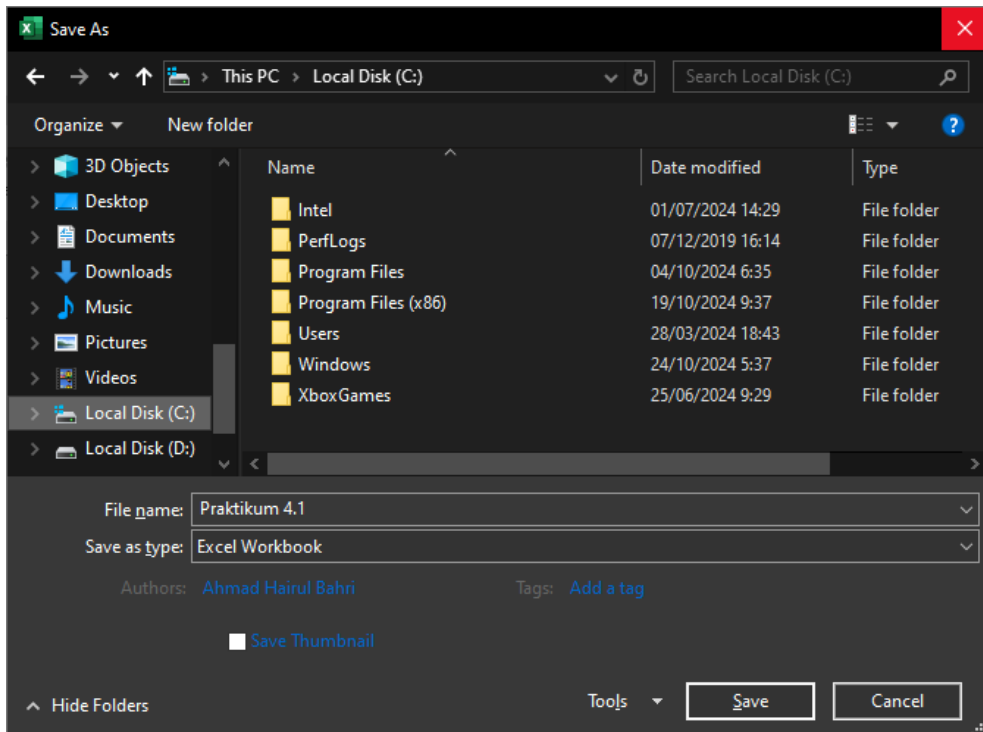
Lama waktu menunggu, yaitu  $x$ , ditentukan dengan cara memaksimalkan fungsi keuntungan  $K(x)$ . Oleh karena itu, pemrograman tak linear tak berkendala yang akan diselesaikan diberikan sebagai berikut:

$$\max K(x) = -500x^2 + 7000x + 6000000$$

Peubah Keputusan	$x$		
Nilai	0		

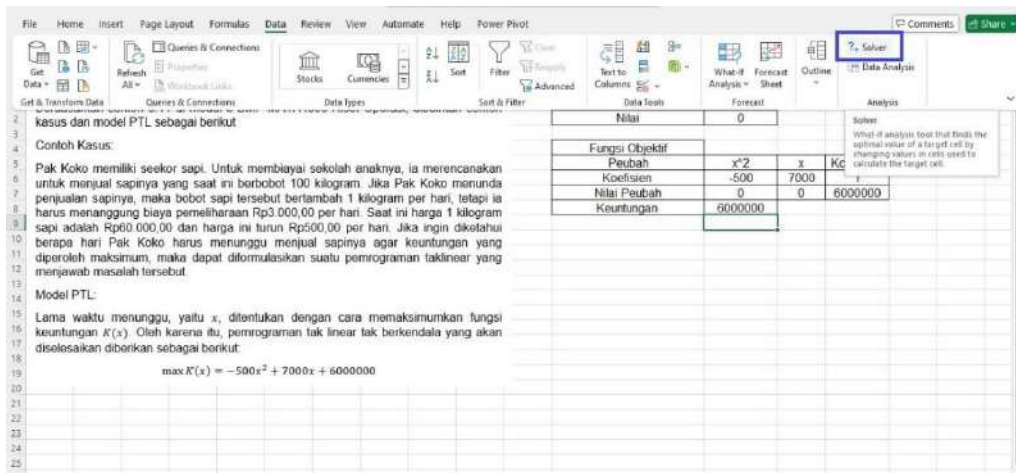
Fungsi Objektif			
Peubah	$x^2$	$x$	Konstanta
Koefisien	-500	7000	1
Nilai Peubah	0	0	6000000
Keuntungan			6000000

17. Save spreadsheet dengan nama file **Praktikum 4.1.xlsx**.



- **Penyelesaian masalah PTL**

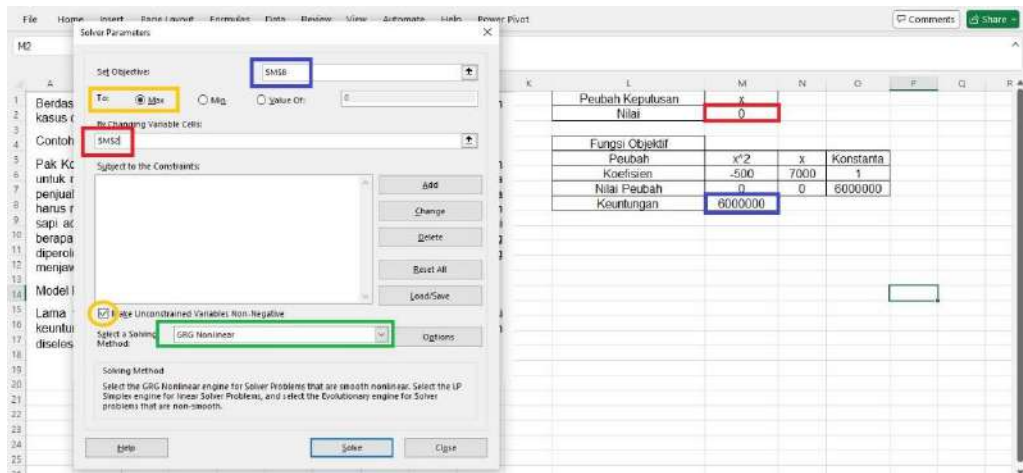
12. Dari file Praktikum 4.1.xlsx, klik icon Solver pada tab Data dan grup Analysis untuk menjalankan Solver.



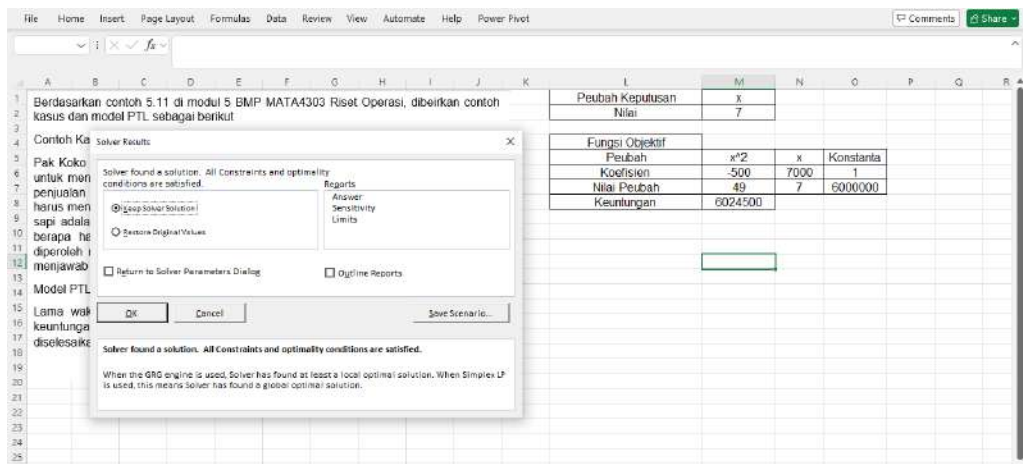
### 13. Pada Solve Parameters:

- klik kotak pada sisi teks “Set Objective” (diberi tanda kotak warna biru) dengan sel M8 yang berisi fungsi objektif Keuntungan yang akan dimaksimumkan
- klik to Max untuk memaksimumkan fungsi objektif Keuntungan
- klik kotak di bawah teks “By Changing Variable Cells” (diberi tanda kotak berwarna merah) dengan dengan sel M2 yang berisi nilai awal peubah keputusan
- Beri centang (✓) pada kotak di sebelah kiri teks “Make Unconstrained Variables Non-Negative” agar peubah keputusan tak berkendala menjadi nilai yang PTL tidak negatif
- Pilih metode penyelesaian GRG Nonlinear pada kotak di sebelah kanan teks “Select a Solving Method”. Metode GRG Nonlinear dipilih karena model PTL satu variabel tersebut tidak berkendala dan fungsi halus.
- Jika semua komponen yang terdapat pada jendela “Solver parameters” sudah terisi dengan benar, maka klik “Solve”. Solver bekerja selama waktu yang dibutuhkan.

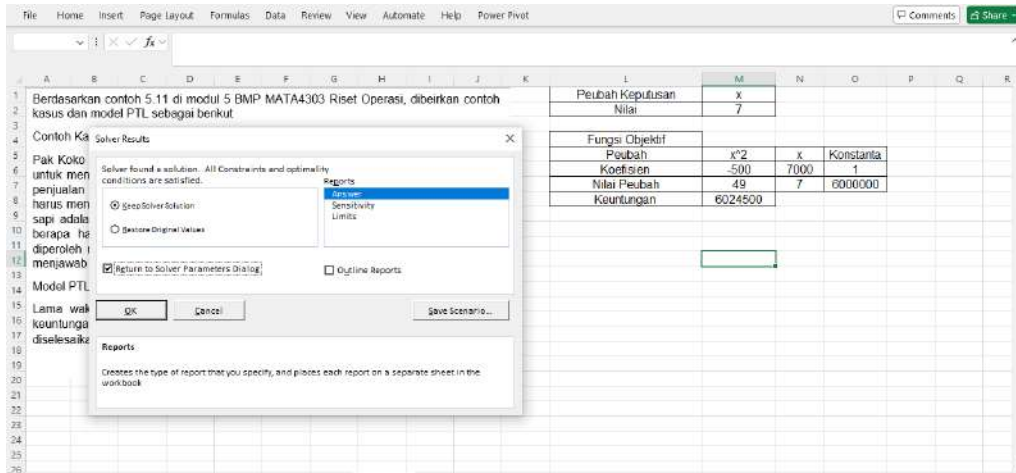




14. Kemudian muncul jendela “Solver Result” dan dibagian bawah ada tulisan yang menyatakan Solver memperoleh solusi optimum yaitu “Solver found a solution. All Constraints and optimality conditions are satisfied”.

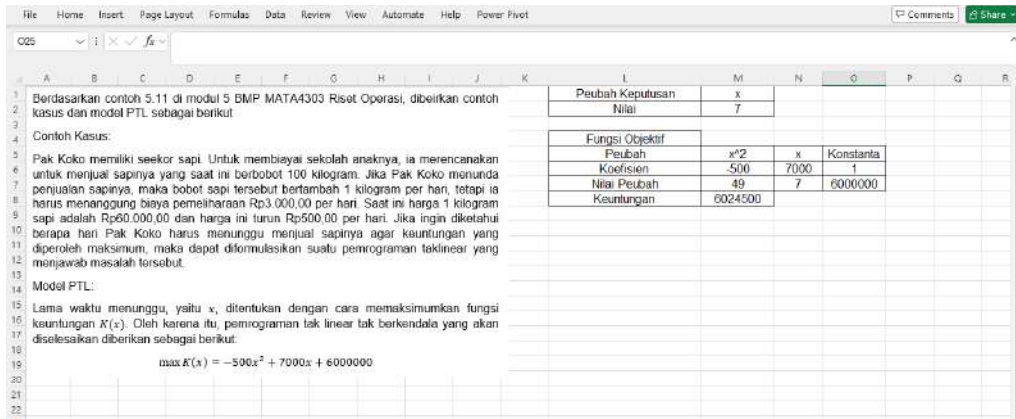


15. Selanjutnya, pilih Keep Solver Solution. Pada opsi Reports, klik Answer. Lalu, klik OK. Anda akan melihat model spreadsheet yang baru dimana nilai beberapa sel berubah dari model spreadsheet yang disiapkan di awal praktikum dan akan tampil sheet baru dengan nama “Answer Report”.



16. Tuliskan perubahan-perubahan yang dapat dilihat pada spreadsheet setelah Solver dijalankan!

- Tampilan Spreadsheet setelah Solver dijalankan:



- Tampilan Answer Report setelah Solver dijalankan:

The screenshot shows the 'Microsoft Excel 16.0 Answer Report' window. The report details the following information:

- Worksheet:** [Book2]Sheet1 (6)
- Report Created:** 11/2/2024 11:47:26 AM
- Result:** Solver found a solution. All Constraints and optimality conditions are satisfied.
- Solver Engine:** GRG Nonlinear
- Solution Time:** 0.016 Seconds
- Iterations:** 2 Subproblems: 0
- Solver Options:** Max Time Unlimited, Iterations Unlimited, Precision 0.000001, Convergence 0.0001, Population Size 100, Random Seed 0, Derivatives Central, Max Subproblems Unlimited, Max Integer Sols Unlimited, Integer Tolerance 1%, Assume Non-Negative
- Objective Cell (Max):**

Cell	Name	Original Value	Final Value
\$M\$2	Keuntungan x^2	6000000	6024500
- Variable Cells:**

Cell	Name	Original Value	Final Value	Integer
\$M\$2	Nilai x	0	7	Constr
- Constraints:** NONE

Berdasarkan spreadsheet setelah solver dijalankan dan answer report, terdapat perubahan nilai  $x$  dan nilai Keuntungan. Nilai  $x$  (peubah keputusan) berubah dari awalnya 0 menjadi 7 dan nilai Keuntungan berubah dari awalnya 6000000 menjadi 6024500. Oleh karena itu, dapat disimpulkan bahwa Pak Koko membutuhkan waktu menunggu selama 7 hari jika ingin mendapatkan keuntungan yang maksimum, yaitu sebesar Rp6.024.500,00.

## Praktikum 4.2

Menyelesaikan masalah PTL berkendala dengan satu variabel menggunakan Solver berdasarkan contoh 5.9 di modul 5 BMP MATA4303 Riset Operasi.

Tentukan nilai maksimum dan minimum global dari fungsi  $f(x) = x + \frac{4}{x}$  untuk  $1 \leq x \leq 8$ .

Model PTL:

1. Model PTL 1 memaksimumkan fungsi objektif dengan peubah keputusan  $x_1$

$$\text{Max } f(x_1) = x_1 + \frac{4}{x_1}$$

$$x_1 \geq 1 \text{ (Kendala 1)}$$

$$x_1 \leq 8 \text{ (kendala 2)}$$

2. Model PTL 2 meminimumkan fungsi objektif dengan peubah keputusan  $x_2$

$$\text{Min } f(x_2) = x_2 + \frac{4}{x_2}$$

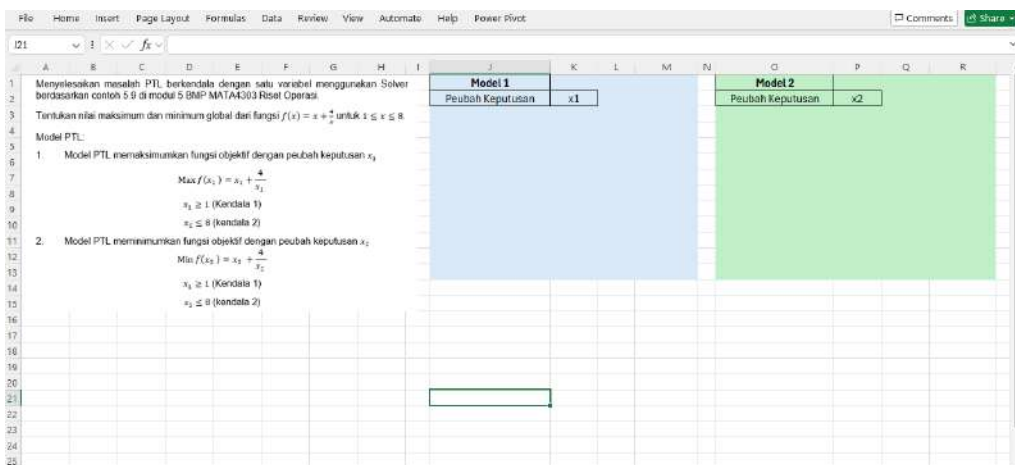
$$x_1 \geq 1 \text{ (Kendala 1)}$$

$$x_1 \leq 8 \text{ (kendala 2)}$$

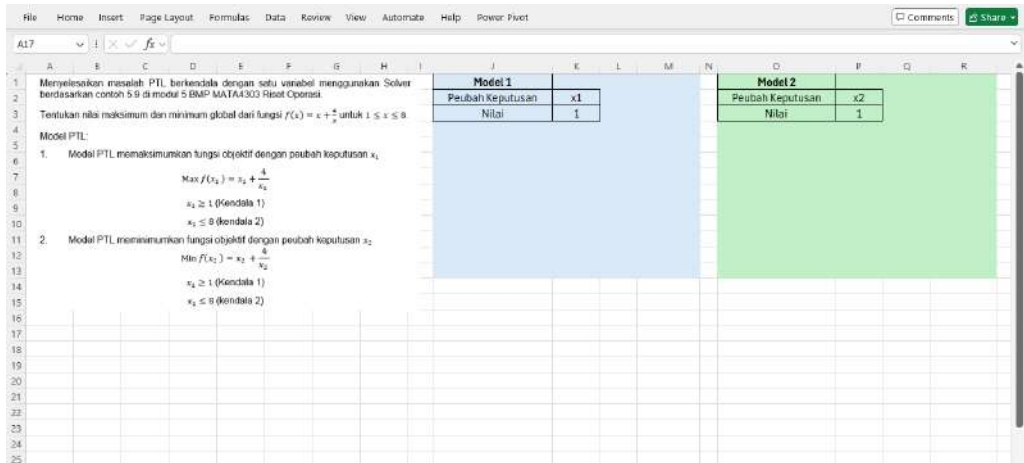
Model PTL 1 dan 2 hanya berbeda perlakuan pada tujuan optimasi. Oleh karena itu, praktikum akan dilakukan dalam satu spreadsheet dengan langkah sebagai berikut.

• **Persiapan spreadsheet masalah PTL**

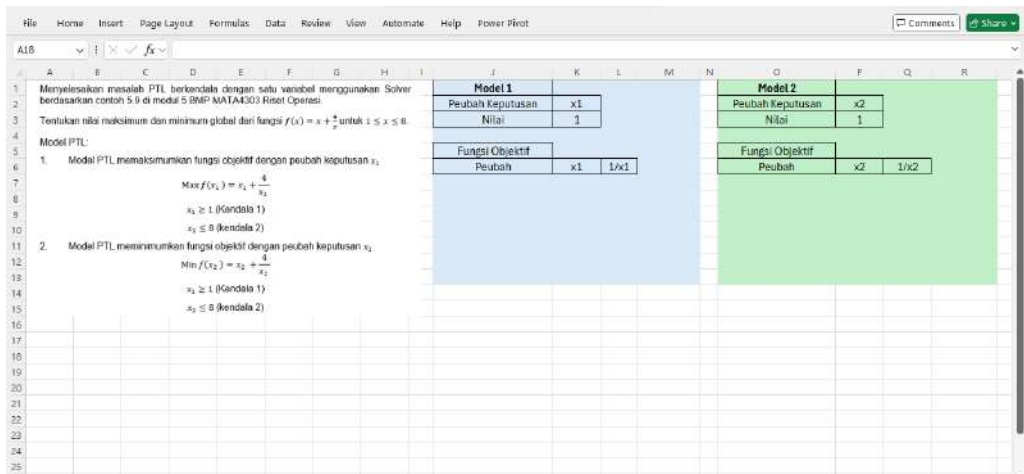
1. Untuk model PTL 1, ketikkan pada suatu sel (misal sel J2 pada spreadsheet) dengan teks “Peubah Keputusan”, kemudian pada sel di sebelah kanannya (sel K2) ketikkan notasi variabel keputusan  $x_1$ . Untuk model PTL 2, ketikkan pada suatu sel (misal sel O2 pada spreadsheet) dengan teks “Peubah Keputusan”, kemudian pada sel di sebelah kanannya (sel P2) ketikkan notasi variabel keputusan  $x_2$ .



18. Untuk model PTL 1, ketikkan teks “Nilai” pada sel dibawah teks “Peubah Keputusan” (sel J3). Kemudian pada sel sebelah kanannya (sel K3) ketikkan bilangan sebarang (yang menjadi nilai awal dari variabel keputusan  $x_1$ ). Untuk model PTL 2, ketikkan teks “Nilai” pada sel dibawah teks “Peubah Keputusan” (sel O3). Kemudian pada sel sebelah kanannya (sel P3) ketikkan bilangan sebarang (yang menjadi nilai awal dari variabel keputusan  $x_2$ ).

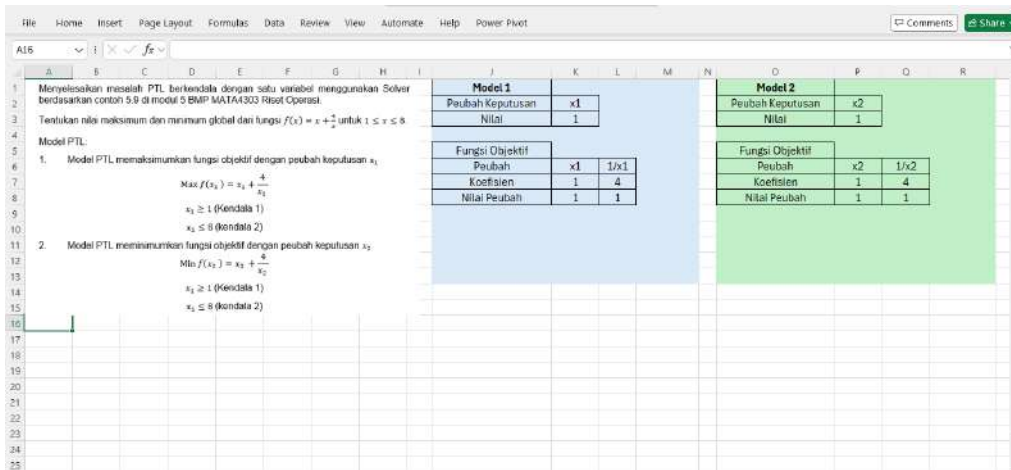


19. Untuk model PTL 1, ketikkan teks “Fungsi Objektif” pada sel kosong sebarang (sel J5). Ketikkan teks “Peubah” pada sel kosong sebarang (sel J6), kemudian input peubah-peubah pada fungsi objektif PTL di sebelah kanannya (sel K6 dan seterusnya). Untuk model PTL 2, ketikkan teks “Fungsi Objektif” pada sel kosong sebarang (sel O5). Ketikkan teks “Peubah” pada sel kosong sebarang (sel O6), kemudian input peubah-peubah pada fungsi objektif PTL di sebelah kanannya (sel P6 dan seterusnya).



20. Untuk model PTL 1, ketikkan teks “Koefisien” pada sel kosong sebarang (misal sel J7), kemudian input koefisien tiap suku pada sel kanannya. Selanjutnya, ketikkan teks “Nilai” pada sel kosong (sel J8) di bawah sel Koefisien. Untuk model PTL 2, ketikkan teks “Koefisien” pada sel kosong sebarang (misal sel O7), kemudian input koefisien tiap suku pada sel kanannya. Selanjutnya, ketikkan teks “Nilai” pada sel kosong (sel O8) di bawah sel Koefisien. Kemudian ketikkan nilai peubah tiap masing-masing suku terhadap peubah keputusan dengan fungsi-fungsi atau nilai sebagai berikut:

- pada K8, ketikkan fungsi : =K3,
- pada L8, ketikkan fungsi : =1/K3,
- pada P8, ketikkan fungsi : =P3, dan
- pada Q8, ketikkan fungsi : =1/P3



21. Pada model PTL 1, ketikkan Fungsi objektif PTL yang akan dimaksimumkan, misal teks “Nilai Maksimum Fungsi” pada sel kosong sebarang (sel K9). Lalu, ketikkan nilai maksimum fungsi objektif pada K9 dengan fungsi sebagai berikut:

$$= \text{SUMPRODUCT}(K7: L7, K8: L8)$$

File Home Insert Page Layout Formulas Data Review View Automate Help Power Pivot

K9  $=\text{SUMPRODUCT}(K7:L7,K8:L8)$

1 Menyelesaikan masalah PTL berkendala dengan satu variabel menggunakan Solver berdasarkan contoh 5.9 di modul 5 BMP MATA4303 Riset Operasi.

2 Tentukan nilai maksimum dan minimum global dari fungsi  $f(x) = x + \frac{1}{x}$  untuk  $1 \leq x \leq 8$ .

3 Model PTL:

4

5 1. Model PTL memaksimumkan fungsi objektif dengan peubah keputusan  $x_1$

6  $\text{Max } f(x_1) = x_1 + \frac{1}{x_1}$

7  $x_1 \geq 1$  (Kendala 1)

8  $x_1 \leq 8$  (Kendala 2)

9

10

11 2. Model PTL meminimumkan fungsi objektif dengan peubah keputusan  $x_2$

12  $\text{Min } f(x_2) = x_2 + \frac{1}{x_2}$

13  $x_2 \geq 1$  (Kendala 1)

14  $x_2 \leq 8$  (Kendala 2)

15

16

17

18

19

20

21

22

23

24

25

Model 1			Model 2		
Peubah Keputusan	$x_1$		Peubah Keputusan	$x_2$	
Nilai	1		Nilai	1	
Fungsi Objektif			Fungsi Objektif		
Peubah	$x_1$	$1/x_1$	Peubah	$x_2$	$1/x_2$
Koefisien	1	4	Koefisien	1	4
Nilai Peubah	1	1	Nilai Peubah	1	1
Nilai Maksimum Fungsi	5		Nilai Maksimum Fungsi	5	

22. Pada model PTL 2, ketikkan Fungsi objektif PTL yang akan diminimumkan, misal teks “Nilai Minimum Fungsi” pada sel kosong sebarang (sel P9). Lalu, ketikkan nilai minimum fungsi objektif pada P9 dengan fungsi sebagai berikut:

$$= \text{SUMPRODUCT}(P7:Q7, P8:Q8)$$

File Home Insert Page Layout Formulas Data Review View Automate Help Power Pivot

P9  $=\text{SUMPRODUCT}(P7:Q7, P8:Q8)$

1 Menyelesaikan masalah PTL berkendala dengan satu variabel menggunakan Solver berdasarkan contoh 5.9 di modul 5 BMP MATA4303 Riset Operasi.

2 Tentukan nilai maksimum dan minimum global dari fungsi  $f(x) = x + \frac{1}{x}$  untuk  $1 \leq x \leq 8$ .

3 Model PTL:

4

5 1. Model PTL memaksimumkan fungsi objektif dengan peubah keputusan  $x_1$

6  $\text{Max } f(x_1) = x_1 + \frac{1}{x_1}$

7  $x_1 \geq 1$  (Kendala 1)

8  $x_1 \leq 8$  (Kendala 2)

9

10

11 2. Model PTL meminimumkan fungsi objektif dengan peubah keputusan  $x_2$

12  $\text{Min } f(x_2) = x_2 + \frac{1}{x_2}$

13  $x_2 \geq 1$  (Kendala 1)

14  $x_2 \leq 8$  (Kendala 2)

15

16

17

18

19

20

21

22

23

24

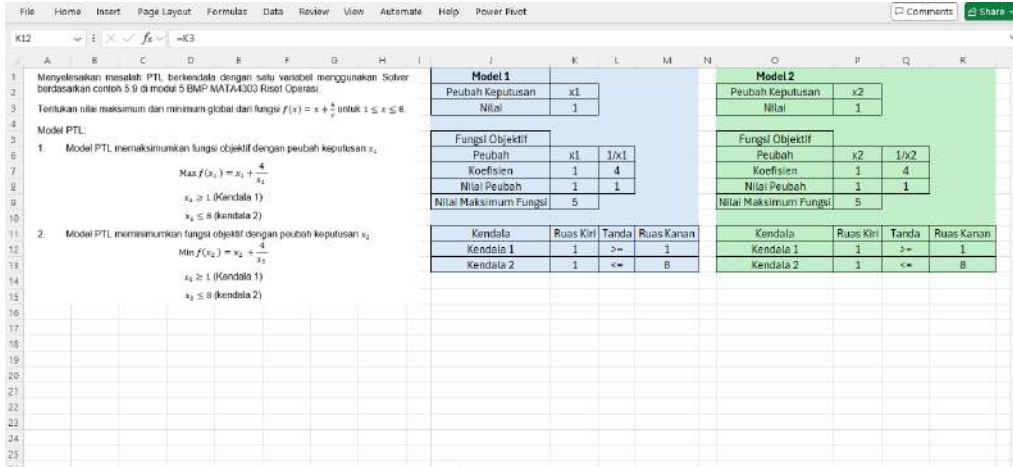
Model 1			Model 2		
Peubah Keputusan	$x_1$		Peubah Keputusan	$x_2$	
Nilai	1		Nilai	1	
Fungsi Objektif			Fungsi Objektif		
Peubah	$x_1$	$1/x_1$	Peubah	$x_2$	$1/x_2$
Koefisien	1	4	Koefisien	1	4
Nilai Peubah	1	1	Nilai Peubah	1	1
Nilai Maksimum Fungsi	5		Nilai Maksimum Fungsi	5	

23. Selanjutnya, pada masing-masing model terdapat dua kendala, yaitu kendala 1 dan kendala 2. Susun masing-masing kendala menjadi 3 (tiga) kolom, yaitu kolom ruas kiri, tanda, dan ruas kanan. Kolom ruas kiri berisi nilai suku-suku pada pertaksamaan kendala yang akan menerima batasan, dalam hal ini isi:

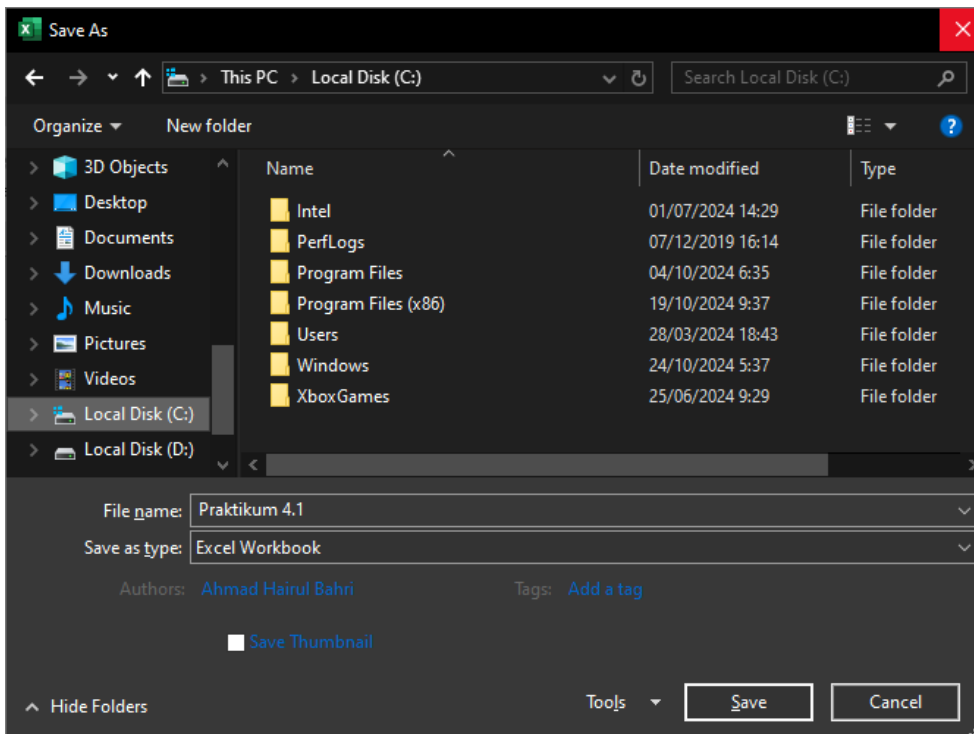
- sel K12 dengan K3,
- sel K13 dengan K3,
- sel P12 dengan P3, dan
- sel P13 dengan P3.



Selanjutnya, isi kolom tanda sesuai tanda pertaksamaan di masing-masing kendala. Kolom ruas kanan berisi nilai batasan untuk masing-masing kendala.



24. Save spreadsheet dengan nama file **Praktikum 4.2.xlsx**.



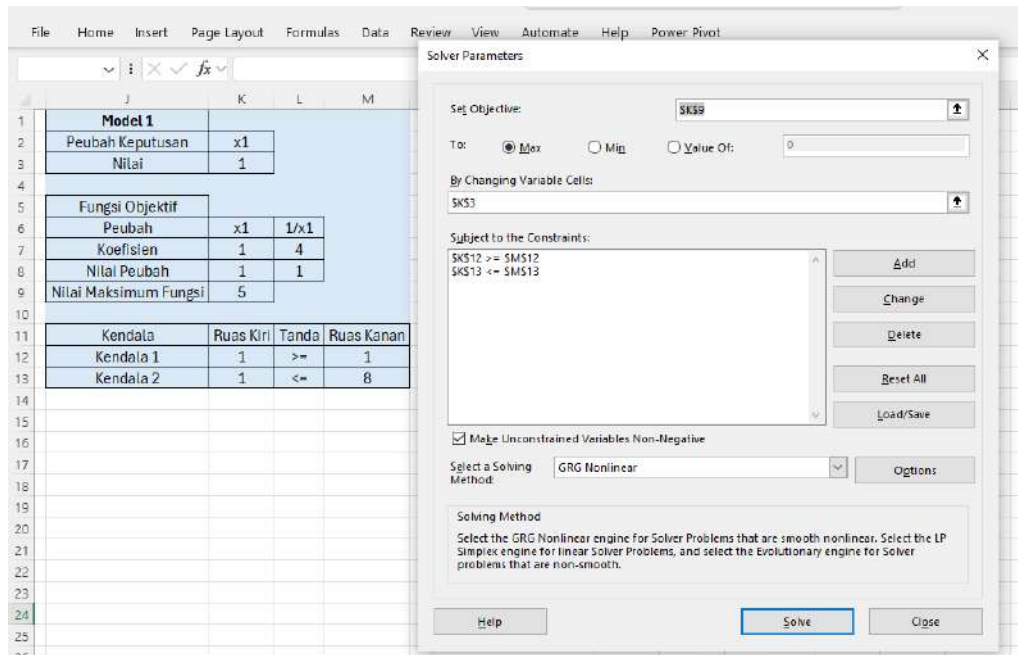


- **Penyelesaian masalah PTL**

1. Dari file Praktikum 4.2.xlsx, klik icon Solver pada tab Data grup Analysis untuk menjalankan Solver.
2. Untuk model PTL 1, pada Solve Parameters:
  - klik kotak pada sisi teks “Set Objective” dengan sel K9 yang berisi fungsi objektif nilai maksimum
  - klik to Max untuk memaksimalkan fungsi objektif
  - klik kotak di bawah teks “By Changing Variable Cells” dengan dengan sel K3 yang berisi nilai awal peubah keputusan
  - Beri centang (✓) pada kotak di sebelah kiri teks “Make Unconstrained Variables Non-Negative” agar peubah keputusan tak berkendala menjadi nilai yang tidak negatif
  - Pilih metode penyelesaian GRG Nonlinear pada kotak di sebelah kanan teks “Select a Solving Method”. Metode GRG Nonlinear dipilih karena model PTL satu variabel tersebut tidak berkendala dan fungsi halus.
  - Jika semua komponen yang terdapat pada jendela “Solver parameters” sudah terisi dengan benar, maka klik “Solve”. Solver bekerja selama waktu yang dibutuhkan.

Pada model PTL 1, setelah klik Solve pada jendela Solver Parameters, kemudian muncul jendela “Solver Result” dan dibagian bawah ada tulisan yang menyatakan Solver memperoleh solusi optimum yaitu “Solver found a solution. All Constraints and optimality conditions are satisfied”.

Selanjutnya, pilih Keep Solver Solution. Pada opsi Reports, klik Answer. Lalu, klik OK. Anda akan melihat model spreadsheet yang baru dimana nilai beberapa sel berubah dari model spreadsheet yang disiapkan diawal praktikum dan akan tampil sheet baru dengan nama “Answer Report”.



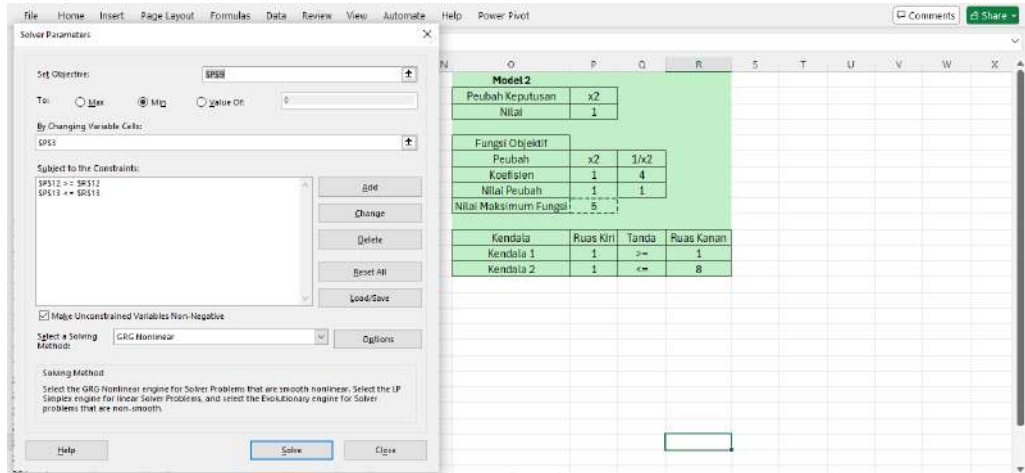
### 3. Untuk model PTL 2, pada Solve Parameters:

- klik kotak pada sisi teks “Set Objective” dengan sel P9 yang berisi fungsi objektif nilai minimum
- klik to Min untuk meminimumkan fungsi objektif
- klik kotak di bawah teks “By Changing Variable Cells” dengan dengan sel P3 yang berisi nilai awal peubah keputusan
- Beri centang () pada kotak di sebelah kiri teks “Make Unconstrained Variables Non-Negative” agar peubah keputusan tak berkendala menjadi nilai yang tidak negatif
- Pilih metode penyelesaian GRG Nonlinear pada kotak di sebelah kanan teks “Select a Solving Method”. Metode GRG Nonlinear dipilih karena model PTL satu variabel tersebut tidak berkendala dan fungsi halus.
- Jika semua komponen yang terdapat pada jendela “Solver parameters” sudah terisi dengan benar, maka klik “Solve”. Solver bekerja selama waktu yang dibutuhkan

Pada model PTL 2, setelah klik Solve pada jendela Solver Parameters, kemudian muncul jendela “Solver Result” dan dibagian bawah ada tulisan yang menyatakan Solver memperoleh solusi optimum yaitu “Solver found a solution. All Constraints and optimality conditions are satisfied”.

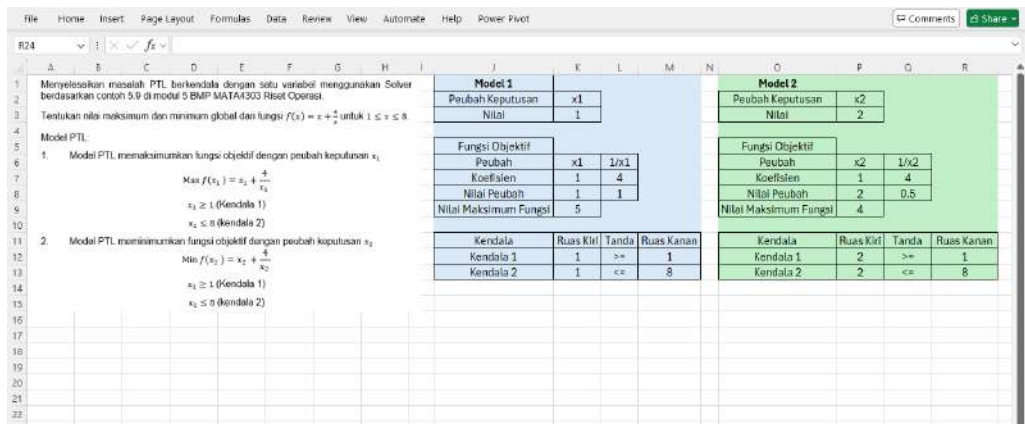
Selanjutnya, pilih Keep Solver Solution. Pada opsi Reports, klik Answer. Lalu, klik OK. Anda akan melihat model spreadsheet yang baru dimana nilai

beberapa sel berubah dari model spreadsheet yang disiapkan diawal praktikum dan akan tampil sheet baru dengan nama “Answer Report”.

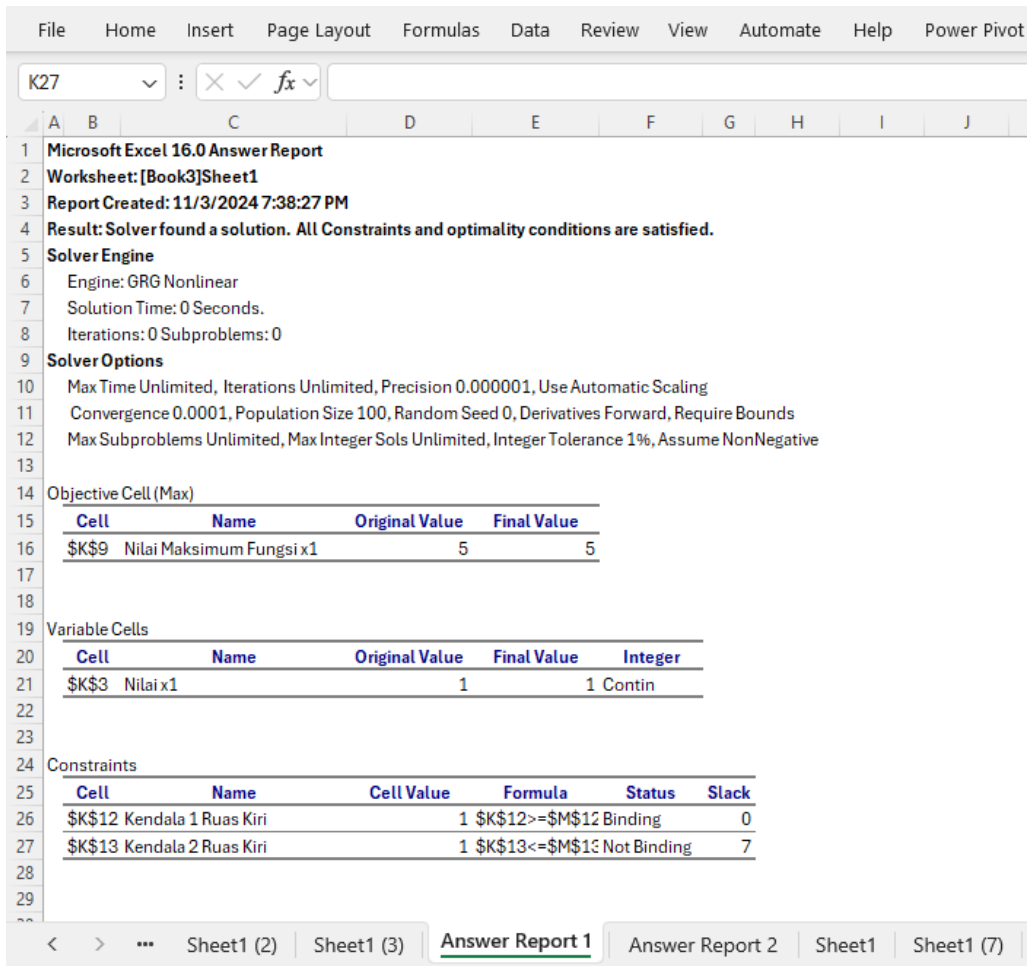


4. Tuliskan perubahan-perubahan yang dapat dilihat pada spreadsheet setelah Solver dijalankan!

- Tampilan Spreadsheet setelah Solver dijalankan:



- Tampilan Answer Report model PTL 1 setelah Sover dijalankan:



- Tampilan Answer Report model PTL 2 setelah Solver dijalankan:

Microsoft Excel 16.0 Answer Report

Worksheet: [Book3]Sheet1

Report Created: 11/3/2024 7:43:39 PM

Result: Solver found a solution. All Constraints and optimality conditions are satisfied.

**Solver Engine**

Engine: GRG Nonlinear  
 Solution Time: 0.016 Seconds.  
 Iterations: 3 Subproblems: 0

**Solver Options**

Max Time Unlimited, Iterations Unlimited, Precision 0.000001, Use Automatic Scaling  
 Convergence 0.0001, Population Size 100, Random Seed 0, Derivatives Forward, Require Bounds  
 Max Subproblems Unlimited, Max Integer Sols Unlimited, Integer Tolerance 1%, Assume NonNegative

**Objective Cell (Min)**

Cell	Name	Original Value	Final Value
\$P\$9	Nilai Maksimum Fungsi x2	5	4

**Variable Cells**

Cell	Name	Original Value	Final Value	Integer
\$P\$3	Nilai x2	1	2.000000028	Contin

**Constraints**

Cell	Name	Cell Value	Formula	Status	Slack
\$P\$12	Kendala 1 Ruas Kiri	2.000000028	\$P\$12 >= \$R\$12	Not Binding	1.000000003
\$P\$13	Kendala 2 Ruas Kiri	2.000000028	\$P\$13 <= \$R\$13	Not Binding	5.999999997

Berdasarkan spreadsheet setelah Solver dijalankan dan answer report, terdapat beberapa informasi yang diperoleh sebagai berikut:

- Nilai  $x_1$  (peubah keputusan model 1) tidak berubah dari awalnya 1 tetap menjadi 1
- Nilai maksimum fungsi objektif tidak berubah dari awalnya 5 tetap menjadi 5
- Nilai kendala 1 pada ruas kiri model 1 tidak berubah dari awalnya 1 tetap menjadi 1
- Nilai kendala 2 pada ruas kiri model 1 tidak berubah dari awalnya 1 tetap menjadi 1
- Nilai  $x_2$  (peubah keputusan model 2) berubah dari awalnya 1 menjadi 2
- Nilai minimum fungsi objektif berubah dari awalnya menjadi 4
- Nilai kendala 1 pada ruas kiri model 2 berubah dari awalnya 1 menjadi 2
- Nilai kendala 2 pada ruas kiri model 2 berubah dari awalnya 1 menjadi 2

Menurut informasi setelah Solver dijalankan, model PTL 1 tidak ada perubahan dan nilai maksimum fungsi yaitu sebesar 5 dengan nilai peubah keputusan sebesar 1. Sedangkan berdasarkan solusi pada contoh 5.9 di BMP MATA4303 Riset Operasi, nilai maksimum fungsi yang dihasilkan tersebut adalah nilai maksimum lokal.

Metode GRG Nonlinear adalah yang tercepat. Namun, kelemahannya adalah solusi yang diperoleh dengan algoritma ini sangat bergantung pada nilai awal. Solver akan berhenti pada nilai optimum lokal yang paling dekat dengan nilai awal sehingga solusi yang diberikan mungkin bukan nilai ekstrem global.

Apabila Anda mencoba untuk menjalankan solver kembali dengan nilai awal peubah keputusan  $x_1$  yaitu sebesar 2 dan nilai maksimum fungsi objektif sebesar 4, maka solusi yang dihasilkan tetap tidak berubah karena nilai awal yang dimuat merupakan solusi optimum lokal.

The screenshot shows an Excel spreadsheet with a Solver Results dialog box open. The spreadsheet contains the following data:

Model 1			
Peubah Keputusan	x1		
Nilai	2		
Fungsi Objektif			
Peubah	x1	1/x1	
Koefisien	1	4	
Nilai Peubah	2	0.5	
Nilai Maksimum Fungsi	4		
Kendala			
	Ruas Kiri	Tanda	Ruas Kanan
Kendala 1	2	>=	1
Kendala 2	2	<=	8

The Solver Results dialog box displays the following text:

Solver found a solution. All Constraints and optimality conditions are satisfied.

Keep Solver Solution  
 Restore Original Values

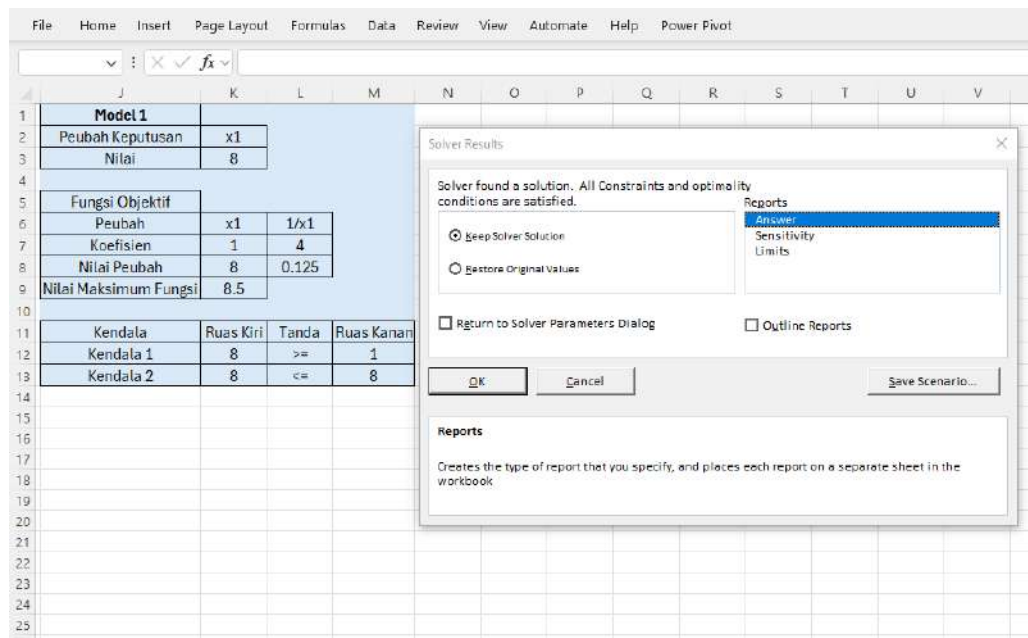
Return to Solver Parameters Dialog  
 Outline Reports

OK Cancel Save Scenario...

Solver found a solution. All Constraints and optimality conditions are satisfied.

When the GRG engine is used, Solver has found at least a local optimal solution. When Simplex LP is used, this means Solver has found a global optimal solution.

Apabila Anda mencoba untuk menjalankan solver kembali dengan nilai awal peubah keputusan  $x_1$  yaitu sebesar 3 dan nilai maksimum fungsi objektif sebesar 4,33, maka nilai peubah keputusan  $x_1$  berubah menjadi 8 dan nilai maksimum fungsi objektif sebesar 8,5.



Perpaduan yang baik antara kecepatan algoritma GRG Nonlinear dan kekuatan algoritma Evolusioner adalah GRG Nonlinear Multistart. Anda dapat mengaktifkan opsi ini melalui jendela Options pada Solver, di bawah tab GRG Nonlinear. Algoritma ini menciptakan populasi nilai awal yang didistribusikan secara acak yang masing-masing dievaluasi menggunakan algoritma GRG Nonlinear tradisional. Dengan memulai beberapa kali dari nilai awal yang berbeda, ada peluang yang jauh lebih besar bahwa solusi yang ditemukan adalah solusi optimal global.

#### • Penyelesaian PTL dengan Multistart GRG Nonlinear

Berdasarkan solusi model PTL 1 dengan GRG Nonlinear, solusi yang dihasilkan belum maksimum global. Oleh karena itu, akan dijalankan Solver dengan Multistart GRG Nonlinear. Setelah mempersiapkan spreadsheet untuk model PTL 1 dengan contoh 5.9 di modul 5 BMP MATA4303 Riset Operasi, pada Solve Parameters di fitur Solver:

- klik kotak pada sisi teks “Set Objective” dengan sel K9 yang berisi fungsi objektif nilai maksimum
- klik to Max untuk memaksimalkan fungsi objektif
- klik kotak di bawah teks “By Changing Variable Cells” dengan dengan sel K3 yang berisi nilai awal peubah keputusan
- Beri centang (✓) pada kotak di sebelah kiri teks “Make Unconstrained Variables Non-Negative” agar peubah keputusan tak berkendala menjadi nilai yang tidak negatif

- Pilih metode penyelesaian GRG Nonlinear pada kotak di sebelah kanan teks “Select a Solving Method”. Metode GRG Nonlinear dipilih karena model PTL satu variabel tersebut tidak berkendala dan fungsi halus.
- Klik Options, Pilih GRG Nonlinear, beri centang () pada kotak Use Multistart, dan klik OK

The screenshot shows the Excel Solver Options dialog box for the GRG Nonlinear method. The dialog box is open over a spreadsheet containing a linear programming model. The spreadsheet data is as follows:

Model 1			
Peubah Keputusan	x1		
Nilai	1		
Fungsi Objektif			
Peubah	x1	1/x1	
Koefisien	1	4	
Nilai Peubah	1	1	
Nilai Maksimum Fungsi	5		
Kendala			
	Ruas Kiri	Tanda	Ruas Kanan
Kendala 1	1	>=	1
Kendala 2	1	<=	8

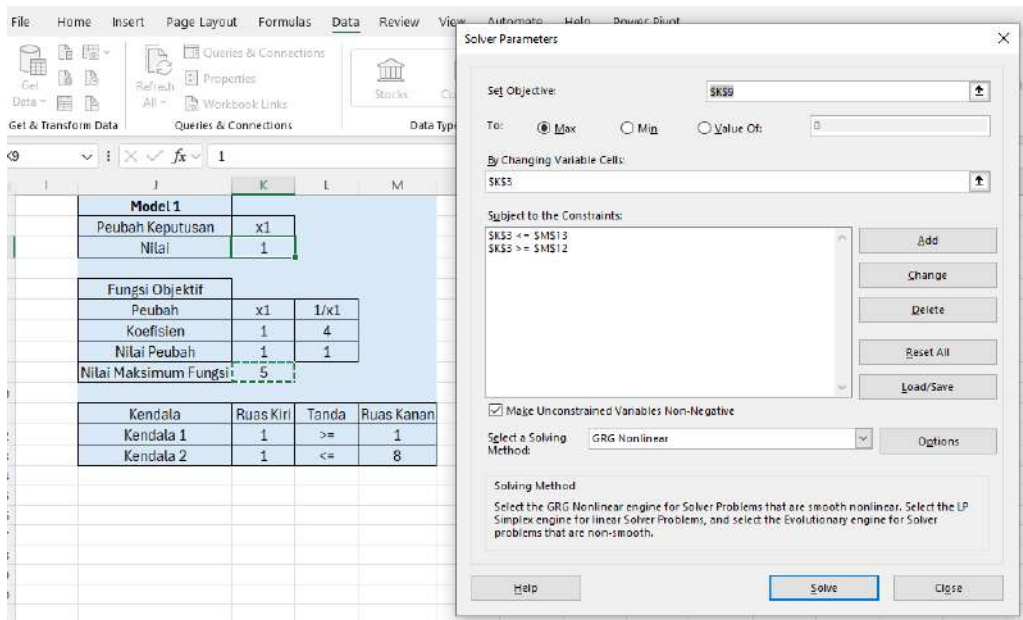
The Options dialog box is configured with the following settings:

- Convergence: 0.0001
- Derivatives:  Forward,  Central
- Multistart:  Use Multistart
- Population Size: 100
- Random Seed: 0
- Require Bounds on Variables

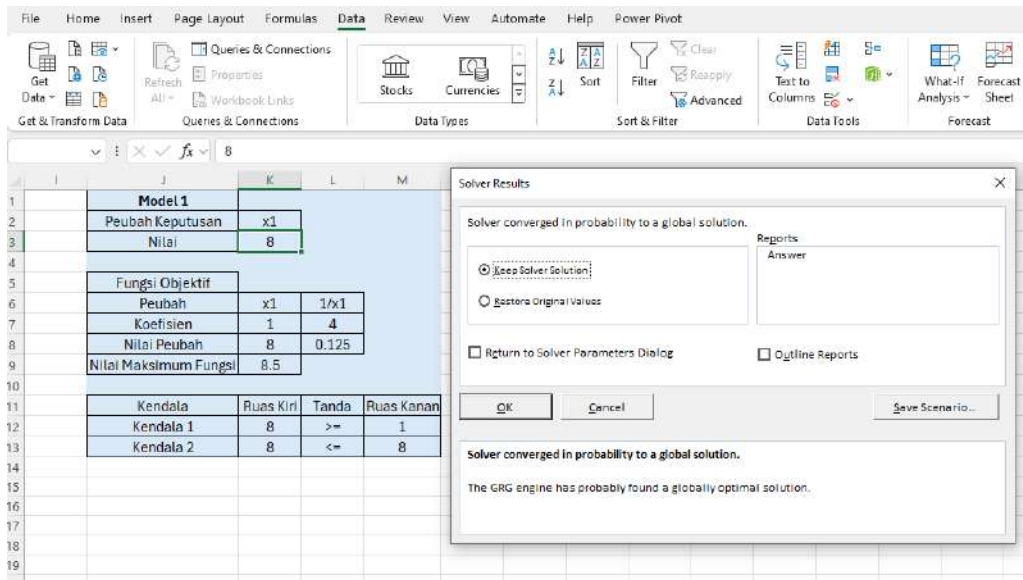
Buttons for OK and Cancel are visible at the bottom of the dialog box.

- Pada Subject to the Constraints, isi batasan atas dan bawah bagi variabel keputusan dengan sel K3 lebih kecil sama dengan M12 dan sel K3 lebih besar sama dengan sel M13.
- Jika semua komponen yang terdapat pada jendela “Solver parameters” sudah terisi dengan benar, maka klik “Solve”. Solver bekerja selama waktu yang dibutuhkan.





- Pada model PTL 1, setelah klik Solve pada jendela Solver Parameters, akan muncul jendela “Solver Result” dan pernyataan “Solver mungkin telah menemukan solusi global” atau “The GRG engine has probably found a globally optimal solution”



- Selanjutnya, pilih Keep Solver Solution.

Berdasarkan spreadsheet setelah solver dijalankan dengan Multistart GRG Nonlinear untuk model 1, terdapat beberapa informasi yang diperoleh sebagai berikut:

- Nilai  $x_1$  (peubah keputusan model 1) berubah dari awalnya 1 menjadi 8
- Nilai maksimum fungsi objektif berubah dari awalnya 5 menjadi 8,5
- Nilai kendala 1 pada ruas kiri model 1 berubah dari awalnya 1 menjadi 8
- Nilai kendala 2 pada ruas kiri model 1 berubah dari awalnya 1 menjadi 8

Jadi, dapat disimpulkan bahwa nilai maksimum fungsi yaitu sebesar 8,5 dengan peubah keputusan sebesar 8 dan nilai minimum fungsi yaitu sebesar 4 dengan peubah keputusan sebesar 2.

### Praktikum 4.3

*Menyelesaikan masalah PTL berkendala dengan banyak variabel menggunakan Solver*

Model PTL:

Peubah Keputusan:

$x_1$  = banyaknya pasien medis yang dirawat

$x_2$  = banyaknya pasien bedah yang dirawat

Fungsi Objektif (dalam ribu dollar) :

$$\text{Max } Z = 13x_1 + 6x_1x_2 + 5x_2 + \frac{1}{x_2}$$

Kendala :

$$2x_1^2 + 4x_2 \leq 90 \text{ (kendala kapasitas keperawatan)}$$

$$x_1 + x_2^3 \leq 75 \text{ (kendala kapasitas x-ray)}$$

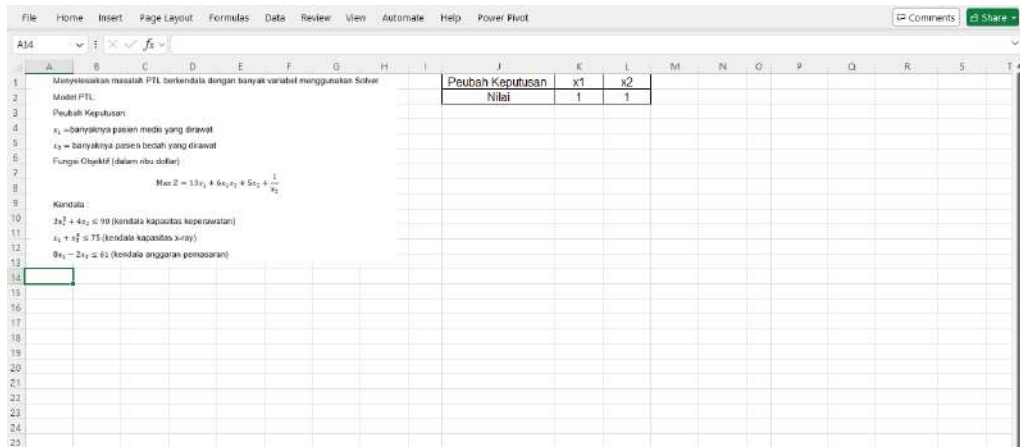
$$8x_1 - 2x_2 \leq 61 \text{ (kendala anggaran pemasaran)}$$

praktikum akan dilakukan dengan langkah-langkah sebagai berikut.

- **Persiapan spreadsheet masalah PTL**

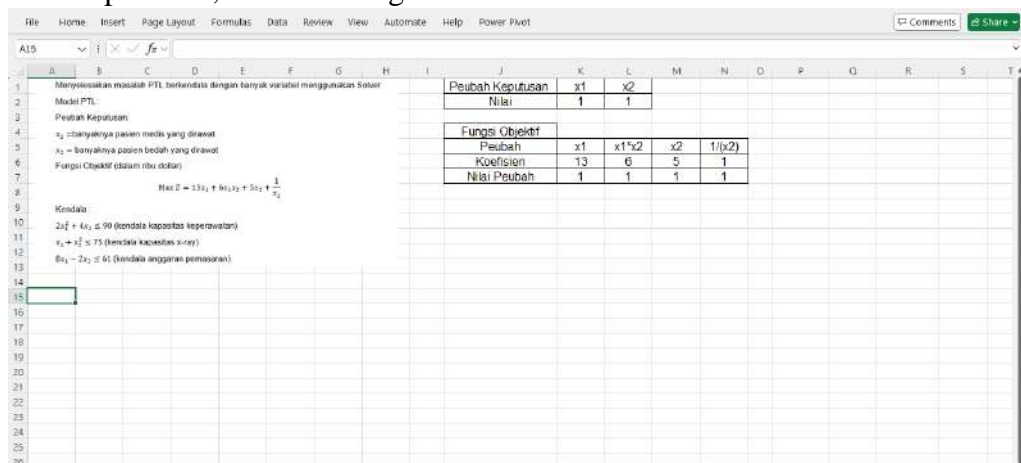
1. Ketikkan pada suatu sel (misal sel J1 pada spreadsheet) dengan teks “Peubah Keputusan”, kemudian pada sel di sebelah kanannya (sel K1 dan

L1) masing-masing ketikkan notasi variabel keputusan  $x_1$  dan  $x_2$ . Lalu, ketikkan pada sel J2 dengan teks “Nilai” dan beri nilai awal masing-masing variabel keputusan pada K2 dan L2 dengan 1 atau sembarang nilai yang terdefinisi.



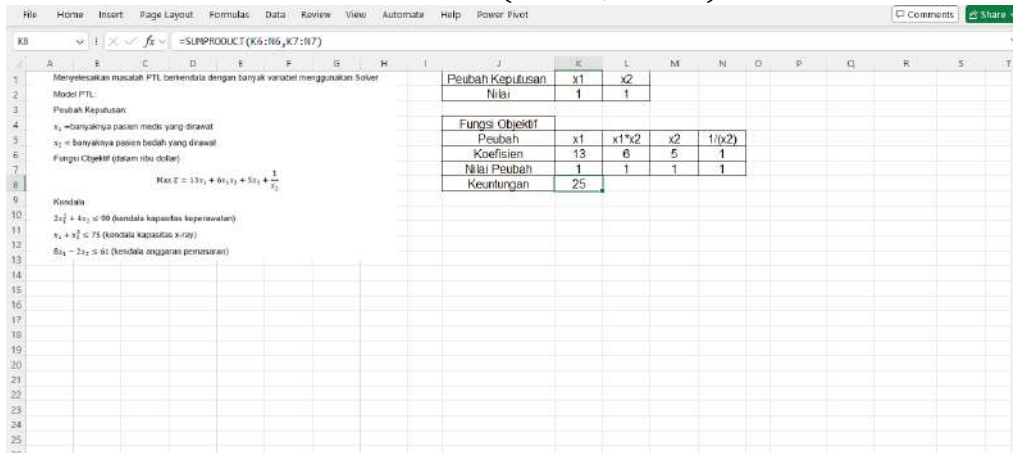
2. Ketikkan teks “Fungsi Objektif” pada sel kosong sebarang (sel J4). Ketikkan teks “Peubah” pada sel kosong sebarang (sel J5), kemudian input peubah-peubah pada fungsi objektif PTL di sebelah kanannya (sel K5 dan seterusnya). ketikkan teks “Koefisien” pada sel kosong sebarang (misal sel J6), kemudian input koefisien tiap suku pada sel kanannya. Selanjutnya, ketikkan teks “Nilai Peubah” pada sel kosong (sel J7) di bawah sel Koefisien. Kemudian ketikkan nilai peubah tiap masing-masing suku terhadap peubah keputusan dengan fungsi-fungsi atau nilai sebagai berikut:

- pada K7, ketikkan fungsi : =K2,
- pada L7, ketikkan fungsi : =K2\*L2,
- pada M7, ketikkan fungsi : =L2, dan
- pada N7, ketikkan fungsi : =1/L2



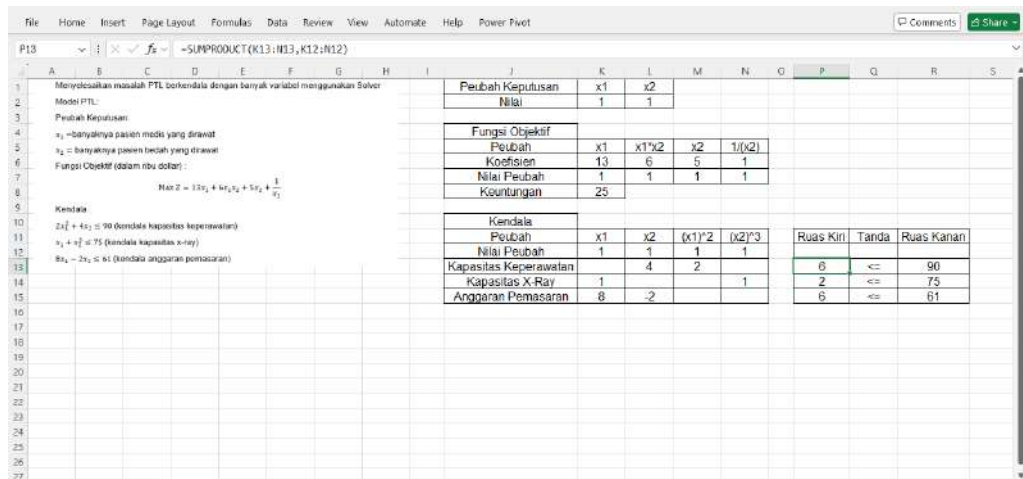
3. Ketikkan Fungsi objektif PTL yang akan dimaksimumkan, misal teks “Keuntungan” pada sel kosong sebarang (sel J8). Lalu, ketikkan nilai keuntungan pada K8 dengan fungsi sebagai berikut:

$$= \text{SUMPRODUCT}(K6:N6, K7:N7)$$

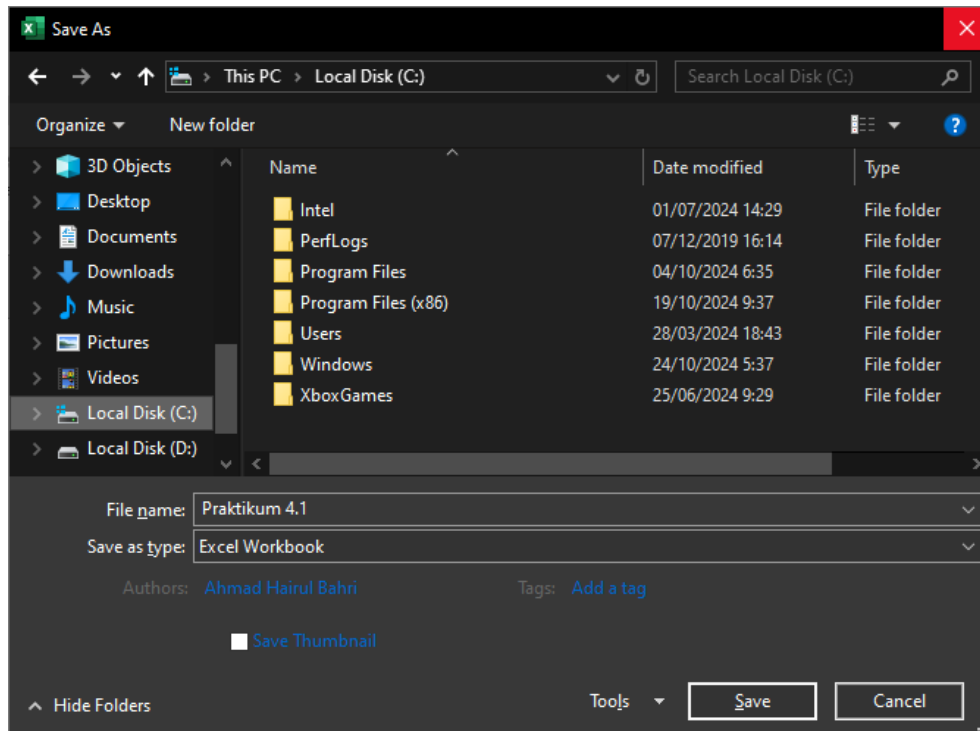


4. Selanjutnya, terdapat tiga kendala, yaitu kendala kapasitas keperawatan, kapasitas X-Ray dan anggaran pemasaran. Ketikkan teks “Kendala” pada sel kosong sebarang (sel J10). Ketikkan teks “Peubah” pada sel kosong sebarang (sel J11), kemudian input peubah-peubah pada kendala di sebelah kanannya (sel K11 dan seterusnya). ketikkan teks “Nilai Peubah” pada sel kosong sebarang (misal sel J12), kemudian input nilai peubah tiap suku pada sel kanannya (sel K12 dan seterusnya). Selanjutnya, ketikkan teks “Kapasitas Keperawatan” pada sel kosong (sel J13), “Kapasitas X-Ray” pada sel kosong (sel J14), dan “Anggaran Pemasaran pada sel kosong (sel J15). Kemudian ketikkan koefisien tiap masing-masing suku terhadap peubah keputusan untuk masing-masing kendala di K13:N15. Apabila suatu kendala tidak memuat peubah tertentu, maka biarkan sel koefisien pada kendala tersebut kosong. Sebagai contoh pada sel K13, tidak terdapat suku dengan peubah  $x_1$  pada kendala kapasitas keperawatan, maka biarkan sel K13 kosong.

Pada sisi kanan, ketikkan teks “Ruas Kiri” pada sel kosong (sel P11), “Tanda” pada sel Q11, dan “Ruas Kanan” pada sel kosong sebarang (sel R11). Selanjutnya, isi kolom ruas kiri dengan fungsi SUMPRODUCT() antara nilai peubah kendala dan koefisien kendala yang dibatasi. Sebagai contoh, untuk kendala kapasitas keperawatan, ruas kiri pada sel P13 diketikkan SUMPRODUCT(K13:N13, K12:N12). Pada kolom tanda, isi tanda pertidaksamaan pada kendala. Selanjutnya, isi kolom ruas kanan dengan nilai batasan masing-masing kendala.



5. Save spreadsheet dengan nama file **Praktikum 4.3.xlsx**.



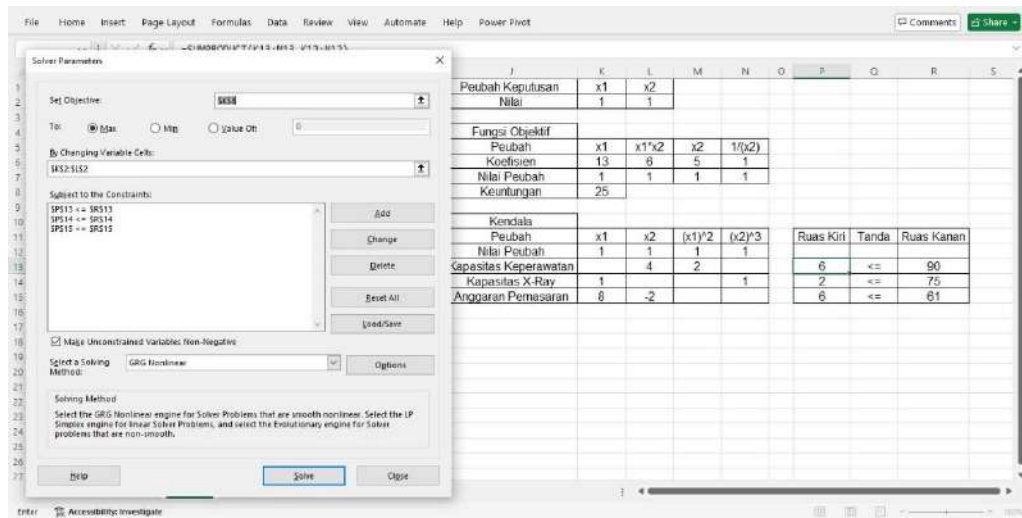
- **Penyelesaian masalah PTL**

6. Dari file Praktikum 4.3.xlsx, klik icon Solver pada tab Data grup Analysis untuk menjalankan Solver.
7. Pada Solve Parameters:

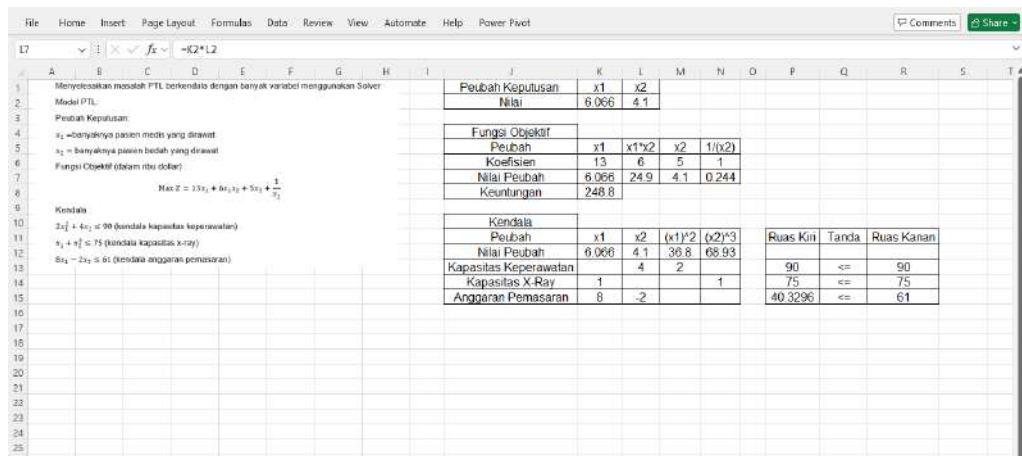
- klik kotak pada sisi teks “Set Objective” dengan sel K8 yang berisi fungsi keuntungan
- klik to Max untuk memaksimumkan fungsi keuntungan
- klik kotak di bawah teks “By Changing Variable Cells” dengan dengan sel K2:L2 yang berisi nilai awal peubah keputusan
- Untuk menambahkan kendala, klik Add pada “Subject to the Constraint”. Isi “Cell Reference” dengan nilai ruas kiri (misal klik sel P13). Untuk sign, sesuaikan dengan tanda pertaksamaan pada kendala. Untuk nilai batasan, isi dengan nilai ruas kanan (misal klik sel R13). Lakukan hal yang sama untuk kendala lainnya
- Beri centang () pada kotak di sebelah kiri teks “Make Unconstrained Variables Non-Negative” agar peubah keputusan tak berkendala menjadi nilai yang tidak negatif
- Pilih metode penyelesaian GRG Nonlinear pada kotak di sebelah kanan teks “Select a Solving Method”. Metode GRG Nonlinear dipilih karena model PTL satu variabel tersebut tidak berkendala dan fungsi halus.
- Jika semua komponen yang terdapat pada jendela “Solver parameters” sudah terisi dengan benar, maka klik “Solve”. Solver bekerja selama waktu yang dibutuhkan.

Pada model, setelah klik Solve pada jendela Solver Parameters, kemudian muncul jendela “Solver Result” dan dibagian bawah ada tulisan yang menyatakan Solver memperoleh solusi optimum yaitu “Solver found a solution. All Constraints and optimality conditions are satisfied”.

Selanjutnya, pilih Keep Solver Solution. Pada opsi Reports, klik Answer. Lalu, klik OK. Anda akan melihat model spreadsheet yang baru dimana nilai beberapa sel berubah dari model spreadsheet yang disiapkan diawal praktikum dan akan tampil sheet baru dengan nama “Answer Report”.



- Tuliskan perubahan-perubahan yang dapat dilihat pada spreadsheet setelah Solver dijalankan!
- Tampilan Spreadsheet setelah Solver dijalankan:



- Tampilan Answer Report setelah Sover dijalankan:



File	Home	Insert	Page Layout	Formulas	Data	Review	View	Automate
B33								
A	B	C	D	E	F	G	H	
1	<b>Microsoft Excel 16.0 Answer Report</b>							
2	Worksheet: [Book1]Sheet1 (4)							
3	Report Created: 11/3/2024 11:05:08 PM							
4	<b>Result: Solver found a solution. All Constraints and optimality conditions are satisfied.</b>							
5	<b>Solver Engine</b>							
6	Engine: GRG Nonlinear							
7	Solution Time: 0.047 Seconds.							
8	Iterations: 6 Subproblems: 0							
9	<b>Solver Options</b>							
10	Max Time Unlimited, Iterations Unlimited, Precision							
11	Convergence , Population Size 0, Derivatives Central							
12	Max Subproblems Unlimited, Max Integer Sols Unlimited, Integer Tolerance %, Assume NonNegative							
13								
14	Objective Cell (Max)							
15	<b>Cell</b>	<b>Name</b>	<b>Original Value</b>	<b>Final Value</b>				
16	\$K\$8	Keuntungan x1	25	248.8456734				
17								
18								
19	Variable Cells							
20	<b>Cell</b>	<b>Name</b>	<b>Original Value</b>	<b>Final Value</b>	<b>Integer</b>			
21	\$K\$2	Nilai x1	1	6.066258709	Contin			
22	\$L\$2	Nilai x2	1	4.100252637	Contin			
23								
24								
25	Constraints							
26	<b>Cell</b>	<b>Name</b>	<b>Cell Value</b>	<b>Formula</b>	<b>Status</b>	<b>Slack</b>		
27	\$P\$13	Kapasitas Keperawatan Ruas Kiri	90	\$P\$13<=\$R\$13	Binding	0		
28	\$P\$14	Kapasitas X-Ray Ruas Kiri	75	\$P\$14<=\$R\$14	Binding	0		
29	\$P\$15	Anggaran Pemasaran Ruas Kiri	40.3295644	\$P\$15<=\$R\$15	Not Binding	20.670436		
30								

Berdasarkan spreadsheet setelah solver dijalankan dan answer report, terdapat beberapa informasi yang diperoleh sebagai berikut:

- Nilai  $x_1$  berubah dari awalnya 1 menjadi 6,066258709
- Nilai  $x_2$  berubah dari awalnya 1 menjadi 4,100252637
- Nilai keuntungan berubah dari 25 menjadi 248,8456734
- Kapasitas Keperawatan berubah dari awalnya 6 menjadi 90
- Kapasitas X-Ray berubah dari awalnya 2 menjadi 75
- Anggaran pemasaran berubah dari awalnya 6 menjadi 40,3295644

Oleh karena itu, dapat disimpulkan bahwa jumlah pasien medis yang dirawat sebanyak 6066 dan jumlah pasien bedah yang dirawat sebanyak 4100 dapat



memaksimalkan keuntungan perawatan di rumah sakit sebesar \$248,8456734 (ribu dollar).

### Latihan

Selesaikan masalah PTL berkendala dengan banyak variabel menggunakan Solver

Model PTL:

Peubah Keputusan:

$x_1$  = banyaknya *toaster oven* yang terjual

$x_2$  = banyaknya *the self-clean toaster oven* yang terjual

Fungsi Objektif:

$$\text{Max } Z = 28x_1 + 21x_2 + 0,25x_2^2$$

Kendala :

$$x_1 + x_2 \leq 1000 \text{ (kendala kapasitas produksi)}$$

$$0,5x_1 + 0,4x_2 \leq 500 \text{ (kendala waktu pengerjaan)}$$

$$x_1, x_2 \geq 0$$

## Lampiran

Is simply dummy text of the printing and typesetting industry.

## » Daftar Pustaka

Is simply dummy text of the printing and typesetting industry.

