

Buku ini mendeskripsikan tentang cara bekerja dengan Mathematica® untuk melakukan kegiatan komputasi baik simbolis dan/atau numeris. Tujuan yang ingin dicapai adalah Anda merasa terbantuan dalam mempelajari/menggunakan perangkat lunak Mathematica® dalam menyelesaikan masalah komputasi yang ditemukan dalam matematika diskrit, kalkulus, geometri, dan aljabar linear tingkat pemula/dasar. Buku ini terdiri dari 8 (delapan) bagian yang setiap bagiannya dilengkapi dengan latihan. Bagian pertama (Menjadi Mahir-1) membahas tentang cara berinteraksi dengan Mathematica®. Bagian kedua (Menjadi Mahir-2) membahas tentang komputasi yang melibatkan fungsi dalam fungsi, diferensiasi fungsi, integrasi fungsi, substitusi nilai peubah fungsi, komputasi persamaan dan sistem persamaan dan matriks. Bagian ketiga (Menjadi Mahir-3) membahas tentang operasi-operasi simbolis melalui pengerjaan visualisasi (geometris) fungsi dalam bentuk eksplisit maupun implisit. Bagian keempat (Menjadi Mahir-4) membahas tentang operasi aritmatika terhadap simbol/lambang. Bagian kelima (Menjadi Mahir-5) membahas tentang penyajian data dalam bentuk list (barisan/himpunan). Bagian keenam (Menjadi Mahir-6) membahas tentang matriks yang tidak lain adalah list dalam list. Bagian ketujuh (Menjadi Mahir-7) membahas tentang solusi sebuah persamaan atau sistem persamaan yang melibatkan turunan fungsi yang dikerjakan secara eksak (simbolis). Bagian kedelapan (Menjadi Mahir-8) membahas tentang solusi sebuah persamaan atau sistem persamaan yang melibatkan turunan fungsi yang dikerjakan melalui hampiran nilai fungsi (numeris).



PENULIS merupakan anak pertama dari tujuh bersaudara yang dilahirkan di kota timah (Dabo Singkep), Kepulauan Riau pada tanggal 13 Februari 1969 dari orang tua bernama La Taka bin La Raha (Buton-Sulawesi Tenggara) dan Nur binti Sahidan (Melayu-Dabo). Penulis mengikuti pendidikan sekolah dasar (SD PN-SDN 004) hingga sekolah menengah (SMPN 1 dan SMAN) di Dabo Singkep- Kepulauan Riau. Pada tahun 1987, penulis melanjutkan pendidikan tingkat S1 di Universitas Riau Pekanbaru-Riau. Di tingkat S1, penulis diterima sebagai mahasiswa di Jurusan Matematika FMIPA Universitas Riau dan memperoleh gelar Sarjana Sains (S.Si) pada tahun 1993. Kemudian pada tahun 1994, penulis mengikuti seleksi sebagai CPNS sebagai tenaga pengajar di Universitas Lampung dan diterima menjadi dosen di Jurusan Matematika FMIPA Universitas Lampung pada bulan April 1994. Pada pertengahan tahun 1998 penulis menerima beasiswa dari program DUE Project Unila-Dikti untuk melanjutkan studi pada program Magister di School Mathematics, Science and Technology Faculty, La Trobe University, Melbourne-Australia. Penulis memperoleh gelar Master of Science (M.Sc.) pada bulan Mei 2001. Penulis sempat bertugas sebagai sekretaris Jurusan Matematika FMIPA Unila periode (2001-2008) sebelum melanjutkan pendidikan di tingkat S3 pada tahun 2011. Penulis mengikuti pendidikan Doktorat pada Kelompok Keahlian Geometri dan Analisis FMIPA-Sekolah Pascasarjana ITB dan meraih gelar Doktor (Dr.) pada Tahun 2017.

Mahir Pemrograman Mathematica® (Tingkat Dasar)

La Zakaria

SERI BUKU MATEMATIKA KOMPUTASI

MAHIR PEMROGRAMAN
MATHEMATICA[®]
(Tingkat Dasar)

Hak cipta pada penulis
Hak penerbitan pada penerbit

Tidak boleh diproduksi sebagian atau seluruhnya dalam bentuk apapun
Tanpa izin tertulis dari pengarang dan/atau penerbit

Kutipan Pasal 72 :

Sanksi pelanggaran Undang-undang Hak Cipta (UU No. 10 Tahun 2012)

1. Barang siapa dengan sengaja dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam Pasal 2 ayat (1) atau Pasal (49) ayat (1) dan ayat (2) dipidana dengan pidana penjara masing-masing paling singkat 1 (satu) bulan dan/atau denda paling sedikit Rp. 1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan atau denda paling banyak Rp. 5.000.000.000,00 (lima miliar rupiah)
2. Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu Ciptaan atau hasil barang hasil pelanggaran Hak Cipta atau Hak Terkait sebagaimana dimaksud ayat (1) dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp. 500.000.000,00 (lima ratus juta rupiah)

SERI BUKU MATEMATIKA KOMPUTASI

MAHIR PEMROGRAMAN
MATHEMATICA[®]
(Tingkat Dasar)

LA ZAKARIA



Perpustakaan Nasional RI:
Katalog Dalam Terbitan (KDT)

**SERI BUKU MATEMATIKA KOMPUTASI
MAHIR PEMROGRAMAN
MATHEMATICA®
(Tingkat Dasar)**

Penulis

LA ZAKARIA

Desain Cover & Layout

PusakaMedia Design

x + 138 hal : 15.5 x 23 cm

Cetakan, Agustus 2020

ISBN: 978-623-6569-16-0

Penerbit

PUSAKA MEDIA

Anggota IKAPI

No. 008/LPU/2020

Jl. Endro Suratmin, Pandawa Raya. No. 100
Korpri Jaya Sukarame Bandar Lampung
082282148711

email : cspusakamedia@yahoo.com

Website : www.pusakamedia.com

Dilarang mengutip atau memperbanyak sebagian
atau seluruh isi buku ini tanpa izin tertulis dari penerbit

KATA PENGANTAR

Puja dan puji syukur kehadirat Allah SWT dan sholawat kepada Rasulullah-Nabiulloh Muhammad SAW karena penulis mendapatkan anugrah ilmu pengetahuan dan ilmu agama sehingga dapat berpartisipasi dalam berbagi ilmu pengetahuan yang dimiliki kepada khalayak ramai melalui buku ini.

Pengalaman yang diperoleh penulis baik belajar maupun mengajar di bidang Matematika, salah satu persoalan yang dihadapi oleh pengguna matematika adalah bagaimana komputasi dapat dilakukan dengan efisien (waktu, ruang, dan biaya) dan efektif (tepat dan akurat). Untuk mencapai efisiensi dan efektifitas yang dimaksud senantiasa memerlukan alat bantu baik perangkat lunak maupun perangkat keras komputer yang mampu melakukannya. Salah satu perangkat lunak yang memiliki kemampuan komputasi yang luar biasa adalah *Mathematica®* milik *Wolfram Research Company*. Buku ini dipersiapkan bagi pengguna yang sedang mengupayakan penyelesaian masalah komputasi menggunakan perangkat lunak *scientific* dengan *branding Mathematica®*. Berbekal pengalaman menggunakan *Mathematica®* untuk proses komputasi dalam menyelesaikan tesis dan disertasi, penulis mencoba menyajikan pengalaman tersebut dalam bentuk buku ini. Buku ini menjadi sebuah kebutuhan bagi pengguna yang memerlukan panduan menggunakan *Mathematica®* sebagai alat komputasi di tingkat pemula yang biasanya berada pada masa-masa awal perkuliahan di prodi ilmu rekayasa dan *basic science*.

Sebagai alat komputasi, *Mathematica®* merupakan sebuah perangkat lunak aplikasi dengan ribuan *tools* terpasang.

Mathematica® berkemampuan tinggi untuk bekerja dengan perhitungan matematis. *Mathematica*® didisain secara akurat untuk benar-benar mengetahui semua standar fungsi dan pengoperasian. *Mathematica*® yang dilengkapi dengan penanganan perhitungan berkelanjutan dan berulang dapat membantu secara dramatis proses perhitungan matematis tanpa adanya kekhawatiran terjadinya kesalahan aritmatika. Hal ini akan membantu pengguna untuk tetap berkonsentrasi pada pemantapan pemahaman konsep dan strategi untuk menyelesaikan persoalan daripada konsentrasi pada proses aritmatika. Akibatnya pengguna dapat menghemat waktu dan biaya untuk berkonsentrasi lebih banyak kepada hal-hal penting lain, misalnya berkenaan dengan pemakaian konsep pada penyelesaian masalah nyata. Dampak ini tentu baik dan berguna bagi guru/pengajar matematika dan/atau *basic science*.

Untuk menghadirkan buku ini dihadapan Anda, ada sejumlah pihak telah turut membantu dalam hal materi dan nonmateri. Oleh karena itu kepada mereka semua diucapkan terima kasih yang setulus-tulusnya. Semoga apa yang telah diperbuat menjadi bermanfaat bagi kehidupan kini dan generasi nanti.

Akhirnya, tanggapan-tanggapan yang bersifat memperbaiki dan menyempurnakan akan sangat diperhatikan demi kemaslahatan ilmu pengetahuan.

Bandar Lampung, Agustus 2020

Penulis

CARA PENGGUNAAN BUKU

Tujuan yang ingin dicapai ketika buku ini hadir di hadapan Anda adalah Anda merasa terbantuan dalam mempelajari/menggunakan perangkat lunak *Mathematica*® dalam melakukan kegiatan komputasi simbolis/numeris dalam setiap upaya menyelesaikan masalah-masalah matematika. Untuk mencapai tujuan itu, Anda perlu mendapatkan informasi bagaimana buku ini bisa efektif dipelajari dan dipahami. Sebagaimana yang penulis alami bahwa mempelajari sebuah bagian dari komputer baik perangkat lunak (*software*) maupun perangkat keras (*hardware*) perangkat tersebut harus ada bersama Anda dan tentunya bersama buku panduannya. Oleh karena itu disarankan bagi pengguna buku ini untuk menghadirkan kedua perangkat tersebut berada bersamanya agar bisa mempraktekan secara langsung apa yang tertulis dalam buku ini. Dengan kata lain buku ini tidak mudah dipahami dan dipelajari dengan cara “menghapal” isinya saja.

Ketika Anda mempraktekkan skrip-skrip yang ada dalam buku ini hendaknya Anda perhatikan hal-hal yang menjadi kekhususan *Mathematica*® baik dalam menulis *command* maupun eksekusi *command*. Dalam buku ini, setiap skrip yang ditulis selalu diawali dengan kata “**In[]:=**”. Anda tidak perlu menuliskannya ke dalam skrip latihan Anda karena skrip ini hanya akan muncul ketika Anda mengeksekusi skrip yang ditulis. Demikian juga dengan kata “**Out[]=**”, kata ini tidak perlu Anda tulis karena kata ini merupakan jawaban langsung dari *Mathematica*® ketika ia menerima perintah eksekusi yang Anda berikan. Terdapat karakter khusus yang perlu berhati-hati penggunaannya, misalnya spasi sebagai pengganti operasi perkalian skalar yang biasa digunakan tanda asterisk “ * “. Kemudian simbol-simbol baku untuk ukuran.

Buku ini dilengkapi dengan latihan, baik latihan dalam bentuk melengkapi skrip maupun latihan membuat skrip dengan *command-command Mathematica®*. Selain itu juga terdapat tanda “(.....)” di bagian tertentu dari skrip-skrip latihan. Tanda itu dimaksudkan meminta Anda mengisi terlebih dahulu sebelum Anda mengeksekusi skrip dengan tujuan agar Anda tau tujuan skrip yang ditulis (jika tanda tersebut berada di sisi kanan skrip) dan Anda tahu output yang diberikan *Mathematica®* ketika skrip tersebut Anda tulis dan eksekusi (jika tanda tersebut berdampingan dengan kata “**Out[]=**”). Selain yang tidak dijelaskan dalam buku ini, Anda dengan leluasa mempelajarinya melalui menu *help-documentation center* yang ada pada *Mathematica®*, bersafari ke laman <https://www.wolfram.com>, atau mengikuti diskusi komunitas *Mathematica®* di laman <https://mathematica.stackexchange.com/>. Dan satu hal lagi penting untuk Anda ketahui bahwa Anda dapat mencoba *platform Mathematica®* secara gratis selama kurang lebih 15 hari melalui laman <https://www.wolfram.com/mathematica/trial/> sebelum Anda memutuskan untuk membeli.

DAFTAR ISI

KATA PENGANTAR.....	v
CARA PENGGUNAAN BUKU.....	vii
DAFTAR ISI.....	ix
MENJADI MAHIR-1 PERANGKAT LUNAK MATHEMATICA®	1
A. Memulai Kerja dengan Mathematica®	1
B. Alat Kelengkapan Mathematica®	4
MENJADI MAHIR-2 KOMPUTASI FUNGSI DAN PERSAMAAN ..	15
A. Menangani Kalkulus dengan Mathematica®	15
B. Objek-objek Mathematica®.....	19
MENJADI MAHIR-3 VISUALISASI FUNGSI/DATA.....	38
A. Arti Penting Visualisasi Fungsi/Data	38
B. Visualisasi Fungsi/Data dengan Mathematica®	41
MENJADI MAHIR-4 OPERASI ARITMATIKA SIMBOLIS.....	49
A. Simbol/Lambang sebagai Tag Objek	49
B. Penjumlahan (Sum) dan Perkalian (Product) suatu Barisan Bilangan	51
C. Mathematica® untuk Matematika Simbolis	57
D. Paket Kalkulus-Analisis Vektor.....	58
E. Mendefinisikan Fungsi dengan Mathematica®	60
F. Membuat Prosedur	61
MENJADI MAHIR-5 LIST DATA.....	66
MENJADI MAHIR-6 MATRIKS (LIST DALAM LIST)	72
A. Pendefinisian Matriks dengan Mathematica®.....	72
B. Operasi Matriks dalam Mathematica®.....	74
C. Aplikasi Matriks (Sistem Persamaan Linear) dalam Mathematica®	81
MENJADI MAHIR-7 KALKULUS DIFERENSIAL: SIMBOLIS.....	84
MENJADI MAHIR-8 KALKULUS DIFERENSIAL: NUMERIS.....	90
DAFTAR PUSTAKA.....	95
INDEKS.....	96

MENJADI MAHIR-1

PENGENALAN PERANGKAT LUNAK

MATHEMATICA[®]

Setelah menyelesaikan kegiatan pada bagian ini Anda akan mampu mengaktifkan laman kerja *Mathematica*[®], bekerja dengan *Mathematica*[®], serta mendokumentasikan hasil kerja ke dalam piranti keluaran (file berektensi ***.nb**). Selain itu Anda juga terampil dalam mengoperasikan dan menggunakan *Mathematica*[®] sebagai alat bantu dalam menyelesaikan masalah-masalah komputasi sederhana yang melibatkan operasi aritmatika dan tampilan grafik/kurva sebuah fungsi.

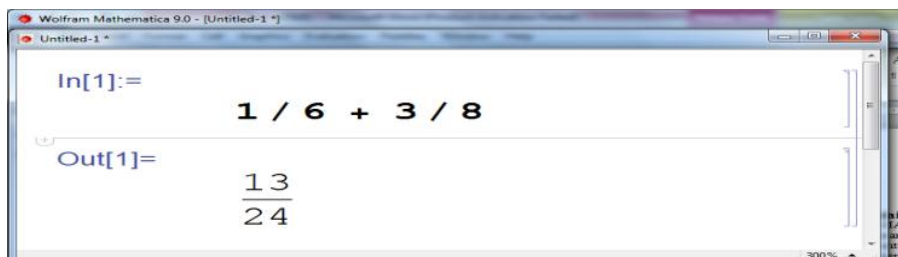
A. Memulai Kerja dengan *Mathematica*[®]

Anda perlu mengenal karakteristik program *Mathematica*[®] sebelum bekerja dengannya. Beberapa hal berikut ini penting untuk Anda ketahui.

1. Penulisan nama fungsi atau *command* terpasang (*default*) yang terdiri satu suku kata dalam *Mathematica*[®] hanya huruf pertama saja berhuruf kapital (*upper case*), misalnya **Sin**[*x*] untuk menyatakan $\sin x$ dan **Plot**[...] untuk menampilkan grafik/kurva sebuah fungsi/data. Sementara itu *command* yang terdiri dari dua suku kata atau lebih, misalnya *Contour* dan *Plot* penulisanya menjadi **ContourPlot**[...]
2. Eksekusi sebuah skrip dilakukan dengan menekan **Shift-Enter** (SE) secara bersamaan (*main button*) atau **Enter** (*numerical button*).

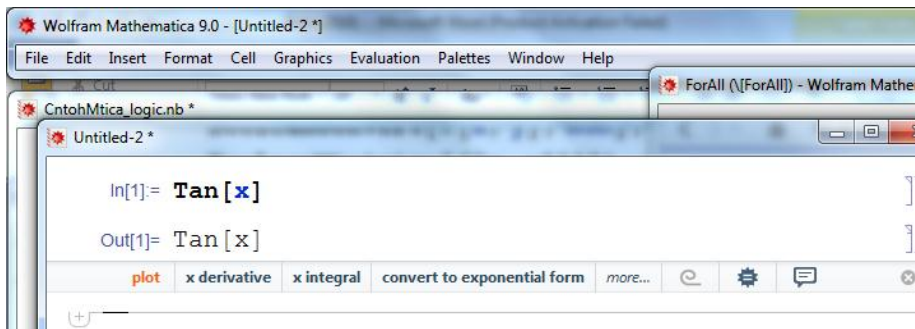
3. Karakter “ ; ” (titik koma/*semicolon*) di akhir sebuah skrip digunakan agar *Mathematica*[®] **tidak menampilkan output** apapun tetapi tetap mengeksekusi dan menyimpannya dalam memori untuk digunakan kembali ketika diperlukan.
4. Jika *Mathematica*[®] membaca intruksi yang diberikan tetapi ia tidak memahaminya, ia selalu merespon dengan menampilkan pesan-pesan kesalahan pada bagian *outputnya*.

Sebagai ilustrasi contoh untuk pemahaman, pada lembar kerja Notebook (.nb), mulailah dengan memasukan ekspresi $1/6 + 3/8$ kemudian tekan tombol SE untuk *running Mathematica*[®] dan diperoleh keluaran sebagaimana tampilan berikut:



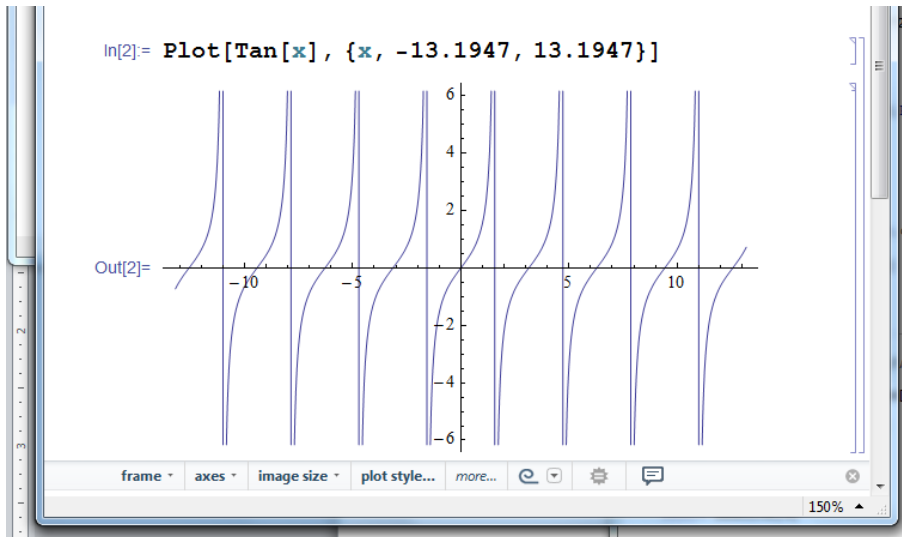
Gambar 1. Tampilan penggunaan operasi penjumlahan di *Mathematica*[®]. Tulisan In[1]:= dan Out[1]= muncul setelah *command* yang ditulis dieksekusi dengan cara menekan tombol **Shift+Enter** pada *keyboard* utama.

Kemudian cobalah dengan menulis skrip **Tan[x]** pada baris berikutnya. Hasil eksekusi seperti tampilan berikut ini.



Gambar 2. Tampilan eksekusi fungsi *Tangen* terpasang di *Mathematica*[®]. Eksikusi yang dilakukan menghasilkan keluaran (Out[1]=) yang sama dengan masukan (In[1]=).

Perhatikan tampilan keluaran *Mathematica*[®] pada Gambar 2 dimana terdapat tampilan submenu pada bagian akhir keluaran. Submenu ini akan segera terlihat setelah keluaran program ditampilkan. Tampilan ini akan hilang ketika *pointer/cursor* Anda menjauh dari baris *output*. Kegunaan submenu ini adalah aksekusi dapat langsung dilakukan oleh program dengan *setting* format terpasang (*default*). Misalnya, untuk menampilkan otomatisasi tampilan grafik/kurva, arahkan *mouse* ke kata “**plot**” lalu klik kiri mause, *Mathematica*[®] akan merespon dan memberikan In[2]-Out[2] sebagaimana tampilan berikut.



Gambar 3. Tampilan eksekusi bentuk geometris (kurva/grafik) fungsi *tangen* terpasang di *Mathematica*[®]. Eksikusi yang dilakukan menghasilkan keluaran (Out[2]=) dengan *setting default* untuk nilai *x* antara -13.1947 dan 13.1947.

Perhatikan tampilan keluaran *Mathematica*[®] pada Gambar 3 yang menampilkan submenu pada bagian akhir keluaran yang berbeda dengan sebelumnya. Isi submenu ini semuanya berkenaan dengan penanganan tampilan grafik/kurva fungsi

B. Alat Kelengkapan *Mathematica*[®]

a. Operator (Aritmatika)

Sebagaimana diketahui bahwa alat komputasi senantiasa menyediakan fasilitas untuk melakukan operasi-operasi aritmatika. Demikian juga dengan *Mathematica*[®] yang menyediakan operator aritmatika terpasang sebagaimana diberikan dalam Tabel 1.

Tabel 1. Notasi operator aritmatika standar pada *Mathematica*[®]

Notasi Matematika	Notasi <i>Mathematica</i> [®]
$x + y$	$x + y$
$x - y$	$x - y$
xy	$x y$ atau $x*y$
x/y	x/y atau $Divide[x,y]$

Beberapa contoh pemakaian operator aritmatika dengan menggunakan *Mathematica*[®] diberikan pada bagian berikut ini.

Catatan:

Tanpa alat bantu kalkulator, Anda dapat melengkapi yang bertanda “(.....)” pada sisi kanan setiap baris terlebih dahulu untuk kemudian bandingkan dengan respon yang diberikan *Mathematica*[®]. Hal semacam ini perlu Anda biasakan karena perilaku *cross-check* itu penting ketika bekerja dengan alat bantu komputer.

- Kalkulasi eksak oleh *Mathematica*[®] untuk bilangan bulat.

In[1]:= $34/55 - 114/27$ (.....)

Out[1]= (.....)

In[2]:= $24/47 \ 94/96$ (.....)

Out[2]= (.....)

In[3]:= $24/47*94/96$ (.....)

Out[3]= (.....)

In[4]:= $(1/2) / (1/2)$ (.....)

Out[4]= (.....)

In[5]:= $1/2/1/2$ (.....)

Out[5]=(.....)

In[6]:= 2^{64} (.....)

Out[6]=(.....)

- Kalkulasi eksak oleh *Mathematica*[®] dengan peubah-peubah.

In[1]:= $x(x+x-y-x)/x$

Out[1]= $x-y$

In[2]:= $\text{Factor}[x^3 - 3x^2y + 3xy^2 - y^3]$

Out[2]= $(x - y)^3$

In[3]:= $\text{Expand}[(x+1)(x-1)^3]$ (.....)

Out[3]=(.....)

In[4]:= $(x^3 - y^3)/(x-y)$

Out[4]=(.....)

In[5]:= $\text{Simplify}[(x^3 - y^3)/(x-y)]$ (.....)

Out[5]=(.....)

Expand, Factor, dan Simplify di atas merupakan nama-nama *command*. Semua nama-nama *command* dimulai dengan huruf kapital (*upper case*). Perhatikan bahwa *command* selalu di sertai dengan simbol kurung siku "[]".

- *Mathematica*[®] dapat juga bekerja dengan hitungan eksak pada sejumlah bilangan khusus.

In[1]:= $(x^3 - y^3) / (x-y) // \text{Simplify}$

Out[1]= $x^2 + xy + y^2$

In[2]:= $\text{Sqrt}[3] \sqrt{4}$ (.....)

Out[2]= (.....)

In[3]:= I^2 (.....)

Out[3]= (.....)

In[4]:= (2+3 I)^10 (.....)
Out[4]= (.....)

In[5]:= Log[Exp[3/5]] (.....)
Out[5]= (.....)

In[6]:= Exp[3/5]/Log (.....)
Out[6]= (.....)

- *Mathematica*[®] dapat melakukan hitungan-hitungan hampiran.

In[1]:= .333333 + .857143 I
Out[1]= 1.19048

In[2]:= Sqrt[2.0] (.....)
Out[2]= (.....)

In[3]:= Sqrt[2.0]^2 (.....)
Out[3]= (.....)

In[4]:= 2.0^64 (.....)
Out[4]= (.....)

b. Pengenal Input-Output

Untuk menandakan sebuah input dari pengguna, *Mathematica*[®] menandainya dengan sebuah *prompt* In[n]:= . Sementara itu, setiap input yang diberikan dan dieksekusi, *Mathematica*[®] memberikan sebuah *label* Out:= [n] yang menandakan sebuah hasil eksekusi (keluaran) dari sebuah input yang diberikan olehnya. Indeks *n* pada *prompt-label* mengindikasikan banyak input-output yang telah dilakukan pengguna dalam satu kegiatan pengoperasian *Mathematica*[®]. Indeks ini berfungsi sebagai penunjuk “alamat” untuk suatu eksekusi yang telah dilakukan. Dengan adanya *prompt-label* pengguna dapat melakukan proses komputasi dengan melibatkan hasil-hasil sebelumnya tanpa mengulangi penulisan skrip secara lengkap.

- **Karakter Penugasan.**

Setiap ekspresi *Mathematica*[®] (numerik, simbol atau gabungan keduanya) dapat diberi nama dengan menggunakan pengenal operator "sama dengan" (=). Nama yang ditandai untuk mengekspresikan suatu numerik atau simbol atau gabungan keduanya, dapat dipanggil kapan saja dan di mana saja dalam *notebook* kerja *Mathematica*[®] ketika ingin digunakan. Perhatikan ilustrasi pemakaiannya berikut ini.

In[1]:= numer=119 * (476+23!) (.....)

Out[1]=(.....)

In[2]:= denom=12²³+1936 (.....)

Out[2]=(.....)

In[3]:= numer/denom (.....)

Out[3]=(.....)

In[4]:= %/numer (.....)

Out[4]=(.....)

In[5]:= %3/%1 (.....)

Out[5]=(.....)

In[6]:= height = 23 feet (.....)

Out[6]=(.....)

In[7]:= width = 14 feet (.....)

Out[7]=(.....)

In[8]:= 123+ (.....)

234

Out[8]=(.....)

In[9]:= 123\ (.....)

+234

Out[9]=(.....)

```
In[10]:= 1+3;2          (.....)
Out[10]= (.....)
```

- **Karakter Pemisah antar Command (;)**

Akhir sebuah *command* yang lengkap dapat disertai sebuah karakter *semicolon* (;) dengan tujuan *command* dieksekusi tapi tidak ditampilkan di *output*. Dalam satu baris dapat dipisahkan dengan menggunakan tanda *semicolon* (;). Perhatikan ilustrasi pemakaiannya berikut ini.

```
In[1]:= 10!
```

```
In[3]:= %1          (angka 1 setelah % menunjukan alamat
Out[3]= 3628800     input yang akan dieksekusi)
```

Selain pemakaian seperti di atas, tanda *semicolon* (;) dipakai untuk menampilkan hasil akhir dari sebuah proses iteratif. Perhatikan penulisan skrip berikut ini.

```
In[1]:= fib[n_]:= Module[ {f},
                                f[1]=f[2]=1;
                                f[i_]:=f[i]=f[i-1]+f[i-2]; f[n]
                                ]
In[2]:= fib[7]
Out[2]= 13
```

Cukup memberikan nilai $n = 7$ untuk mendapatkan nilai ke $\text{fib}[7]$ tanpa meminta program merinci satu persatu berapa nilai $\text{fib}[3]$ hingga $\text{fib}[6]$. Dan bagaimana dengan keluaran potongan program berikut ini?

```
In[3]:= gcd[m0_,n0_]:=          (.....)
                                Module[{m=m0,n=n0},
                                While[n!=0,{m,n}=
                                {n,Mod[m,n]};
                                m ]
In[4]:= gcd[18,21]
Out[4]= (.....)
```


d. Fasilitas Bantuan dan Pencarian

Pengguna pemula tidak perlu khawatir dengan hal-hal baru yang dijumpai ketika bekerja dengan *Mathematica*[®]. Layanan bantuan (*Help*) tersedia dalam perangkat lunak ini. Pengguna cukup menyertakan tanda tanya “?” yang diikuti oleh kata/kalimat yang ingin ditanyakan. Misalnya akan ditanyakan apa itu “Accuracy”. Dengan *Mathematica*[®] cukup ditulis skrip “?Accuracy”. Respon *Mathematica*[®] adalah:

```
In[1]:= ?Accuracy
```

Accuracy[x] gives the number of digits to the right of the decimal point in the number x.

Selain *Help*, fasilitas bantuan yang tersedia pada *Mathematica*[®] Myang berperan penting untuk membuat menjadi terbiasa dan mahir menggunakan *Mathematica*[®] dalah fasilitas pencarian (*searching*). Ada banyak topik yang tersedia dalam *Mathematica*[®] sehingga jika pengguna ingin mengetahui secara rinci (definisi dan contoh) penulisan skrip di *Mathematica*[®] cukup membuka menu *Help* kemudian submenu *Documentation Centre* lihat Gambar 5.).



Gambar 5. Tampilan fasilitas pencarian kata/karakter pada *Mathematica*[®].

e. Konstanta dan Fungsi.

MATHEMATICA[®] memiliki sejumlah *tools* untuk kemudahan komputasi dengannya. Beberapa diantaranya diberikan dalam bentuk konstanta dan fungsi baku/standar yang dapat dikembangkan lebih

kompleks lagi. Beberapa contoh konstanta dan fungsi yang dimaksud diberikan dalam Tabel 2.

Tabel 2. Beberapa Konstanta dan fungsi terpasang dalam *Mathematica*[®]

Notasi Matematika	Notasi <i>Mathematica</i> [®]
e	Exp[1] atau E
π	Pi
i	I atau Sqrt[-1]
$\sin x$	Sin[x]

In[1]:= Log [E]
Out[1]= 1

In[2]:= Sin [Pi]
Out[2]= 0

In[3]:= I²
Out[3]= -1

f. Pemakaian Tanda Pengganti/Penyulih.

Nama suatu peubah dalam *Mathematica*[®] dapat di-On atau di-Off tergantung kebutuhan. *Command* untuk kebutuhan ini adalah "=" dan "Clear[x1, ... , xn]".

In[1]:= x=3
Out[1]= 3

In[2]:= y=5
Out[2]= 5

In[3]:= x =.
Out[3]=

In[4]:= Clear [y]
Out[4]=

Mathematica[®] menyediakan beberapa operator tambahan khusus yang berperan mengganti ganti suatu nilai peubah secara langsung.

Operator yang dimaksud adalah "++", "--", "+=", "-=", "*=", dan "/=" (lihat keterangan operator ini dalam Tabel 3).

Tabel 3. Beberapa Operator Khusus *Mathematica*[®] untuk Penggantian Nilai Peubah Secara Otomatis

Command	Keterangan
x++	penambahan nilai x dengan 1
x--	pengurangan nilai x dengan 1
x += k	penambahan nilai x dengan k
x -= k	pengurangan nilai x dengan k
x *= k	perkalian nilai x dengan k
x /= k	pembagian nilai x dengan k

Penggunaan *command* operator khusus seperti yang diperlihatkan dalam Tabel 3 diberikan dalam beberapa contoh skrip *Mathematica*[®] berikut ini.

In[1]:= x=3 (Memberikan angka 3
 kepada x)
 Out[1]= 3

In[2]:= x++ (.....)
 Out[2]= (.....)

In[3]:= x (.....)
 Out[3]= (.....)

In[4]:= x-- (.....)
 Out[4]= (.....)

In[5]:= x (.....)
 Out[5]= (.....)

In[6]:= x+=2 (.....)
 Out[6]= (.....)

In[7]:= x-=3 (.....)

Out[7]=(.....)

In[8]:= x*=4 (.....)

Out[8]=(.....)

In[9]:= x/=2 (.....)

Out[9]=(.....)

In[10]:= x++

Increment :: rvalue:

x has not been assigned a value.

Sebelum menutup bagian ini sebaiknya Anda mencoba kembali untuk “*Tour of Mathematica®*” melalui materi latihan berikut.

Latihan

1. Buka menu **Help**. Kemudian pilih **Documentation Centre**. Setelah itu ketik kata “REPLACE” dibaris pencarian. Kemudian klik kata “**Replace**”. Salin (*copy*) tulisan/skrip “`Replace[x^2, x^2 -> a + b]`” dan pindahkan (*paste*) ke kertas kerja Anda (*notebook*) kemudian eksekusi dengan cara menekan tombol **Shift-Enter** dan lihat *output*-nya. Berikan komentar anda tentang *output* yang dihasilkan.
2. Gunakan metode pencarian serupa dengan langkah-langkah #1 di atas untuk menelusuri skrip “**ReplaceAll**”, “**StringReplace**”, dan “**Graphs and networks**”. Setelah Anda memperolehnya, buka bagian “**Examples**” dan amati beberapa contoh yang tersedia. Jika Anda menemukan contoh yang “menarik” coba Anda salin (*copy*) untuk kemudian Anda pindahkan (*paste*) di laman *notebook* kerja Anda. Eksekusi apa yang Anda pindahkan dengan menekan **Shift-Enter** pada *keyboard*.
3. Cobalah untuk melakukan operasi aritmatika melalui operator penugasan dengan *command* “:=” dan “=” berikut ini:
In[1]:= x = 1
Out[1]= 1

In[2]:= y := x²+1 (.....)

Out[2]= (.....)

In[3]:= y (.....)

Out[3]= (.....)

In[4]:= x=0 (.....)

Out[4]= (.....)

In[5]:= y (.....)

Out[5]= (.....)

In[6]:= 5 (.....)

Out[6]= (.....)

In[7]:= y:=% (.....)

Out[7]= (.....)

In[8]:= y (.....)

Out[8]= (.....)

In[9]:= 2 (.....)

Out[9]= (.....)

In[10]:= y (.....)

Out[10]= (.....)

Dari beberapa contoh di atas, pernyataan "y:=x" bermakna apa?
Dan bagaimana dengan pernyataan "y=x"?

MENJADI MAHIR-2

KOMPUTASI FUNGSI DAN PERSAMAAN

Bagian ini menyajikan lebih banyak proses komputasi dengan tingkatan penggunaan proses aritmatika dan grafik fungsi yang lebih tinggi. Setelah menyelesaikan bagian ini Anda akan mahir dalam komputasi fungsi dalam fungsi, diferensiasi fungsi, integrasi fungsi, substitusi nilai peubah fungsi, komputasi persamaan dan sistem persamaan dan matriks. Indikator ketercapaian dapat diukur dari kemampuan mampu bekerja dengan *Mathematica*[®] berkenaan dengan operasi-operasi simbolis untuk menyelesaikan persoalan aritmatika dan fungsi-fungsi sederhana yang biasa diberikan materi matematika dasar, seperti kakulus dan pengantar aljabar linear elementer.

A. Menangani Kalkulus dengan *Mathematica*[®].

Ada sejumlah hal menarik di *Mathematica*[®] dimana ia dapat bekerja untuk mengevaluasi integral dan diferensial. Untuk mengintegral, mendiferensial, dan menyelesaikan persamaan diferensial dari sebuah fungsi, masing-masing dapat dilakukan dengan menggunakan *command* "**Integrate**", "**D**", dan "**DSolve**". Dalam bagian berikut ini, ilustrasi pemakaian *command* **Integrate**, **D**, dan **DSolve** diberikan. Untuk memahami arti setiap *command* isilah bagian bertanda "(.....)" terlebih dahulu kemudian bandingkan dengan respon yang diberikan oleh *Mathematica*[®].

- **Diferensiasi Fungsi Satu Peubah**

Dalam Matematika dasar (Kalkulus) simbol turunan/diferensial fungsi satu peubah sering dijumpai. Misalnya turunan pertama fungsi $y = f(x)$ yang dinotasikan dengan $\frac{dy}{dx}$. Komputasi $\frac{dy}{dx}$ dalam

Mathematica[®] menggunakan *command*

$$\mathbf{D[y, x]}$$

Misalnya, $\frac{dy}{dx} = \sin x$, skrip *command* pada *Mathematica*[®] dinyatakan dengan

$$\begin{array}{ll} \text{In[1]:= D[Sin[x],x]} & \text{(Menurunkan fungsi Sin } x \text{ satu kali} \\ \text{Out[1]= Cos [x]} & \text{terhadap } x, \text{ turunan ke-1: Cos } x.) \end{array}$$

Anda juga dapat melakukan kalkulasi $\frac{d^n y}{dx^n}$ menggunakan *command*

$$\mathbf{D[y,\{x, n,\}]}$$

Misalnya, $\frac{d^2 y}{dx^2} = \sin x$, skrip *command* pada *Mathematica*[®] dinyatakan dengan

$$\begin{array}{ll} \text{In[2]:= D[Sin[x],\{x, 2\}] } & \text{(Menurunkan fungsi Sin } x \text{ dua kali} \\ \text{Out[2]= -Sin [x]} & \text{terhadap } x, \text{ turunan ke-1: Cos } x; \text{ turunan} \\ & \text{ke-2: - Sin } x.) \end{array}$$

Untuk mendapatkan hasil komputasi $f'(x)$ dan $f''(x)$ menggunakan *Mathematica*[®] dapat juga menggunakan *command* :

$$\mathbf{F '[x]} \text{ dan } \mathbf{F '' [x]}$$

Misalnya, $\frac{dy}{dx} = F'(x) = \sin' x$, skrip *command* pada *Mathematica*[®] dinyatakan dengan

```
In[3]:= Sin'[x]                (.....)
Out[3]= (.....)
```

```
In[4]:= Sin''[x]             (.....)
Out[4]= (.....)
```

• **Integrasi**

Seperti halnya diferensiasi, komputasi untuk sebuah integrasi fungsi juga dapat dikerjakan dengan *Mathematica*[®]. Komputasi integral fungsi seperti $\int f(x) dx$ dapat dilakukan dengan menggunakan *command* berikut ini.

Integrate[nama fungsi, peubah bebas fungsi]

Misalnya untuk $\int \cos(x) dx$ dan $\int \frac{1}{(x^3 + 1)} dx$, skrip *command* pada

Mathematica[®] dinyatakan dengan :

```
In[1]:= Integrate[Cos[x],x]    (.....)
Out[1]= (.....)
```

```
In[2]:= Integrate[1/(x^3+1),x] (.....)
Out[2]= (.....)
```

Sementara itu untuk komputasi integrasi fungsi tertentu (berbatas) seperti komputasi integral fungsi $\int_a^b f(x) dx$ dapat dilakukan dengan menggunakan skrip berikut ini:

Integrate[f,{x, a, b}]

Pandang integral fungsi $\int_1^e \frac{1}{x} dx$. Skrip *Mathematica*[®] dapat

dinyatakan dengan:

In[3]:= Integrate[1/x, {x,1,E}] (.....)

Out[3]= (.....)

• **Persamaan Diferensial (Biasa)**

Ketika pengguna ingin menyelesaikan sebuah persamaan atau sistem persamaan diferensial (biasa) *Mathematica*[®] menyiapkan fasilitas **DSolve**. Untuk sebuah persamaan diferensial yang dinyatakan dengan format *persamaandiferensial* = 0 dapat dilakukan dengan menggunakan *command*

DSolve[*persamaandiferensial* == 0, f[x], x]

Misalnya akan dilakukan komputasi untuk menyelesaikan persamaan diferensial $f''(x) - f(x) = 0$, skrip *command* pada *Mathematica*[®] dinyatakan dengan :

In[1]:= DSolve[f''[x]-f[x] ==0, f[x],x] (.....)

Out[1]= (.....)

Sementara itu untuk menyelesaikan persamaan diferensial biasa yang disertai dengan nilai awal dapat dilakukan dengan menambah nilai awal yang disyaratkan. Untuk sebuah ekspresi *persamaandiferensial* = 0 dengan fungsi $f(x)$ dan nilai awal $f(a) = b$, dapat dilakukan dengan menggunakan *command*

DSolve[{*persamaandiferensial* == 0, f[a] = b}, f[x], x]

Misalnya untuk integral fungsi $f''(x) - f(x) = 0$ dan nilai awal $f(0) = 3$, skrip *command* pada *MATHEMATICA*[®] dinyatakan dengan :

In[2]:= DSolve[{D[f[x],x] - f[x] ==0, (.....)

f[0] ==3}, f[x],x]

Out[2]= (.....)

B. Objek-objek Mathematica®.

Mathematica® dapat bekerja dengan tipe-tipe objek yang berbeda misalnya fungsi, simbol, bil.eksak, bil. desimal, grafik dan persamaan.

• Fungsi Terpasang (Default)

Penemu Mathematica®, Wolfram, mengerti bahwa pengguna membutuhkan sejumlah fungsi-fungsi standar yang umum dikenal dalam dunia sains. Oleh karena itu pada program Mathematica®, disediakan sejumlah fungsi-fungsi standar. Beberapa diantaranya adalah seperti diberikan dalam Tabel 4.

Tabel 4. Command untuk fungsi versi Mathematica® dan deskripsinya.

Notasi Fungsi (Standar)	Deskripsi
$Sqrt[x]$	Akar kuadrat (\sqrt{x})
$Exp[x]$	Eksponensial (e^x)
$Log[x]$	Logaritma Natural $\log_e(x) = \ln(x)$
$Log[b.x]$	Logaritma dengan basis b : ${}^b \log(x)$
$Sin[x], Cos[x], Tan[x]$	Fungsi trigonometri (Sin, Cos, dan Tg)
$ArcSin[x], ArcCos[x], ArcTan[x]$	Fungsi balikan/inversi trigonometri (Sin^{-1} , Cos^{-1} , dan Tg^{-1})
$n!$	n faktorial ($n \times (n-1) \times (n-2) \times \dots \times 2 \times 1$.)
$Abs[x]$	Nilai Absolut/mutlak
$Round[x]$	Bilangan bulat terdekat
$Mod[n,m]$	n modulo m
$FactorInteger[n]$	Faktor (utama) bilangan asli n beserta kelipatannya (perpangkatan)
$Random[]$	Bilangan Acak antara 0 sampai dengan 1
$Max[x,y,\dots], Min[x,y,\dots]$	Menentukan bilangan terbesar/terkecil dari daftar angka x, y, \dots

Sebagaimana terlihat dalam Tabel 4 bahwa semua argumen fungsi dalam *Mathematica*[®] menggunakan huruf pertama pada setiap kata berupa huruf *kapital* dan diikuti karakter *kurung siku* “[]”. Untuk membuat Anda lebih banyak latihan, berikut ini diberikan sejumlah contoh penggunaan fungsi dalam *Mathematica*[®].

- ✓ Mengkalkulasi nilai logaritma alami dari bilangan real 8.4
 $\text{In}[1]:= \text{Log}[8.4]$
 $\text{Out}[1]= 2.12823$

- ✓ Mengkalkulasi nilai akar kuadrat dari bilangan bulat 16
 $\text{In}[2]:= \text{Sqrt}[16]$
 $\text{Out}[2]= 4$

- ✓ Mengkalkulasi akar kuadrat dari bilalangan asli 2 untuk hasilnya bernilai numerik.
 $\text{In}[3]:= \text{Sqrt}[2] // \text{N}$
 $\text{Out}[3]= 1.41421$

- ✓ Mengkalkulasi akar kuadrat dari bilangan real 2. Perhatikan hasilnya berupa numerik desimal.
 $\text{In}[4]:= \text{Sqrt}[2.]$
 $\text{Out}[4]= 1.41421$

- ✓ Mengkalkulasi akar kuadrat dari bilangan asli 2. Perhatikan hasilnya berupa notasi akar kuadrat dari 2
 $\text{In}[5]:= \text{Sqrt}[2]$
 $\text{Out}[5]= \sqrt{2}$

- ✓ Mengkalkulasi nilai dari 30 faktorial
 $\text{In}[6]:= 30!$
 $\text{Out}[6]= 265252859812191058636308480000000$

- ✓ Mengkalkulasi nilai hampiran numerik untuk 30 faktorial
 $\text{In}[7]:= 30! // \text{N}$
 $\text{Out}[7]= 2.65253 \times 10^{32}$

Selain fungsi, beberapa *command* dalam *Mathematica*[®] merupakan sebuah konstanta yang bernilai khusus sebagaimana diberikan dalam Tabel 5. Simbol/kata yang terdapat dalam Tabel 5 tidak boleh digunakan sebagai peubah sebuah fungsi atau digunakan dengan maksud lain selain yang telah dinyatakan pada kolom kedua Tabel 5.

Tabel 5. Notasi konstanta versi *Mathematica*[®] dan deskripsinya.

Nama Konstanta	Nilai Konstanta
Pi	$\pi \approx 3.14159$
E	$e \approx 2.71828$
Degree	Konversi derajat ($\pi / 180$)
I	Imajiner ($i = \sqrt{-1}$)
Infinity	Ketakberhinggaan (∞)

Perhatikan beberapa skrip yang diperlihatkan berikut ini dimana konstanta-konstanta tetap ditulis dengan huruf kapital di awal namanya.

In[8]:= Pi ^ 2 //N (Mengkalkulasi π^2 untuk nilai *default* numerik)
 Out[8]= 9.8696

In[9]:= Sin [Pi/2] (Mengkalkulasi Sin $\pi/2$)
 Out[9]= 1

In[10]:= Sin [20 Degrees] (Nilai konversi derajat dari Sin 20°)
 Out[10]= 0.34202

In[11]:= Log [E ^ 5] (Mengkalkulasi Ln e^5)
 Out[12]= 5

In[13]:= Log [2, 256] (Mengkalkulasi ${}^2\text{Log } 256$)
 Out[13]= 8

In[14]:= Pi ^ 1 //N (.....)
 Out[14]= (.....)

In[15]:= Sin [Pi/4] (.....)

Out[15]= (.....)

In[16]:= Sin [60 Degrees] (.....)

Out[16]= (.....)

In[17]:= Log [E^4] (.....)

Out[17]= (.....)

In[18]:= Log [10, 1000] (.....)

Out[18]= (.....)

• Penyulihan/Substitusi Nilai Peubah/Symbol

Di dalam *Mathematica*[®] untuk mengganti suatu lambang/symbol dengan suatu nilai/bilangan tertentu menggunakan pasangan karakter *dot-slash* (./).

Tabel 6. Format Penulisan Skrip *Mathematica*[®] untuk substitusi nilai peubah

Format penulisan skrip	Tujuan substitusi
<i>ekspresi (berpeubah x) / . x → nilai peubah</i>	Pemberian nilai untuk peubah x
<i>ekspresi (berpeubah x & y) / {x → nilai peubah1, y → nilai peubah2}</i>	Pemberian nilai secara bersamaan untuk masing-masing peubah x dan y

Deskripsi dari penulisan skrip yang diperlihatkan dalam Tabel 6 dalam bentuk *command Mathematica*[®] diberikan berikut ini.

In[1]:= 1 + 2x / . x → 3 (.....)

Out[1]= (.....)

In[2]:= 1 + x + x^2 / . x → 2 - y (.....)

Out[2]= (.....)

In[3]:= x → 3 + y (.....)

Out[3]= (.....)

In[4]:= $x^2 - 9 / .\%$ (.....)

Out[4]= (.....)

In[5]:= $(x + y)(x - y)^2 / .$
 $\{x \rightarrow 3, y \rightarrow 1 - a\}$ (.....)

Out[5]= (.....)

In[6]:= $x = 3$ (.....)

Out[6]= (.....)

In[7]:= $x^2 - 1$ (.....)

Out[7]= (.....)

In[8]:= $x = 1 + a$ (.....)

Out[8]= (.....)

In[9]:= $x^2 - 1$ (.....)

Out[9]= (.....)

• **Pemasang-Pembebas Nilai Peubah (Simbol)**

Bila diinginkan pemasangan suatu peubah (simbol) dengan dengan suatu nilai tertentu dimana saja pada satu kertas kerja (Notebook) maka digunakan tanda **sama dengan** (=). Sebaliknya untuk membebaskan suatu peubah (simbol) dari suatu nilai tertentu maka digunakan *command* **sama dengan-dot** (=.). Dalam Tabel 6 diperlihatkan format skrip untuk pemasangan dan pembebasan sebuah peubah x dari sebuah nilai *value*.

Tabel 6. Skrip *Mathematica*[®] untuk pemasangan dan pembebasan nilai untuk suatu peubah (simbol)

Format skrip <i>Mathematica</i> ®	Arti skrip
$x = \text{value}$	Pemasangan nilai <i>value</i> untuk peubah x dimana saja pada satu kertas kerja (<i>notebook</i>) sampai dengan diberikan tanda pembebasannya.
$x = .$	Pembebasan peubah x dari sebuah nilai <i>value</i> yang dipasangkan sebelumnya dimana saja pada satu kertas kerja.

In[10]:= $x + 5 - 2x$ (.....)

Out[10]= (.....)

In [11]:= $x =$ (.....)

Out[10]= (.....)

In[12]:= $x + 5 - 2x$ (.....)

Out[12]= (.....)

In[13]:= $t = 1 + x^2$ (.....)

Out[13]= (.....)

In[14]:= $t /. x \rightarrow 2$ (.....)

Out[14]= (.....)

In[15]:= $t /. x \rightarrow 5a$ (.....)

Out[15]= (.....)

In[16]:= $t /. x \rightarrow Pi // N$ (.....)

Out[16]= (.....)

- **Penghimpunan nilai-nilai fungsi (data) dan operatornya**

Melakukan sebuah operasi aritmatika dari beberapa objek dirasakan lebih mudah jika objek-objek itu dikumpulkan dalam sebuah penghimpun. Mathematica® mempunyai *command* tersendiri untuk penghimpun yang dikenal dengan sebutan **List**. Sebuah **List** biasa disebut juga himpunan merupakan sekumpulan objek-objek. Misalkan himpunan {3,7,5} merupakan kumpulan 3 buah objek berupa bilangan asli 3, 7, dan 5. Penulisan/penamaan sekumpulan data tersebut dalam Mathematica® diekspresikan dengan:

List[objek]

Perhatikan dua contoh dalam skrip berikut ini

```
In[1]:= List[a,b,c,d]
```

```
Out[1]= {a,b,c,d}
```

dan

```
In[2]:= FullForm[{a,b,c,d}]
```

```
Out[2]//FullForm= List[a,b,c,d]
```

Misalkan L1 dan L2 adalah dua buah *list*. Beberapa operator yang umum digunakan dalam menangani *list* adalah sebagai berikut:

- | | | |
|------------------------------|---|---|
| Union [L1, L2] | ⇒ | <i>Command</i> untuk mengevaluasi gabungan (<i>Union</i>) dua himpunan L1 dan L2. |
| Intersection [L1, L2] | ⇒ | <i>Command</i> untuk mengevaluasi irisan (<i>Intersection</i>) dua himpunan L1 dan L2. |
| Join [L1, L2] | ⇒ | <i>Command</i> serupa <i>Union</i> hanya saja semua objek ditulis dalam order yang tetap dan adanya pengulangan jika objek tersebut sama. |
| Flatten [[L1, L2]] | ⇒ | <i>Command</i> serupa <i>Joint</i> tetapi dapat berlaku juga untuk <i>list</i> dalam <i>list</i> . |
| L1[[n]] | ⇒ | <i>Command</i> untuk menampilkan elemen ke- n dari sebuah <i>list</i> L1. |

Perhatikan beberapa contoh pemakaian operator *list* berikut ini.

In[1]:= L1={1,1,2,2,x,y,z}

Out[1]= {1, 1, 2, 2, x, y, z}

In[2]:= L2={1,1,3,4,w,x}

Out[2]= {1, 1, 3, 4, w, x}

In[3]:= Intersection[L1,L2]

Out[3]= {1, x}

In[4]:= Flatten[{{a},{b,{c}},{d}}]

Out[4]= {a,b,c,d}

Untuk latihan pemakaian operator lainnya, Anda dapat mencoba contoh-contoh skrip *Mathematica*® berikut ini.

In[1]:= Union[L1,L2] (.....)

Out[1]= (.....)

In[2]:= Intersection[{1,2} {3,4}] (.....)

Out[2]= (.....)

In[3]:= Join[{3,2,1} {1,2,3}] (.....)

Out[3]= (.....)

In[4]:= Union[%] (.....)

Out[4]= (.....)

In[5]:= S={0,Pi/4,Pi/2,3 Pi/4,Pi} (.....)

Out[5]= (.....)

In[6]:= Sin[S] (.....)

Out[6]= (.....)

In[7]:= %[[2]] (.....)

Out[7]= (.....)

In[8]:= k{a,b,c} (.....)

Out[8]= (.....)

In[9]:= {a,b,c}/k (.....)
 Out[9]= (.....)

In[10]:= {a,b,c}+{d,e,f} (.....)
 Out[10]= (.....)

In[11]:= A={S, Sin[S]} (.....)
 Out[11]= (.....)

In[12]:= MatrixForm[A] (.....)
 Out[12]= (.....)

In[13]:= Length[A] (.....)
 Out[13]= (.....)

In[14]:= Dimensions[A] (.....)
 Out[14]= (.....)

In[15]:= A[[1,2]] (.....)
 Out[15]= (.....)

In[16]:= Flatten[A] (.....)
 Out[16]= (.....)

In[17]:= Partition[%,5] (.....)
 Out[17]= (.....)

Hal-hal yang dapat dicatat dari demonstrasi di atas adalah *Mathematica*[®]:

1. Operasi jumlah, tambah, kali, dan bagi dapat diberlakukan secara langsung pada sebuah *list*.
 2. Ekspresi **Matriks** tidak lain adalah sebuah bentuk *list* dalam *list*.
- **Pengulangan (Loops)**
 - ✓ *Command* Pengulangan **Do**

MATHEMATICA[®] menyediakan fasilitas untuk memproses sebuah pengulangan (*loop*) aktivitas komputasi. Salah satu bentuk *command* yang digunakan untuk pengulangan adalah **Do**. Pengulangan **Do** merupakan sebuah pengulangan yang batas sebuah pengulangan ditentukan. Berikut bentuk format pengulangan **Do** dalam *Mathematica*[®]

$$\mathbf{Do}[\text{ekspresi}, \{\text{indeks}, \text{indeksmin}, \text{indeksmax}\}]$$

Dalam format **Do** di atas bagian *ekspresi* merupakan bagian yang akan dilakukan pengulangan. Sementara *indeks* merupakan sebuah peubah bertipe bilangan bulat (*integer*) dengan batas awalnya adalah *indeksmin* dan batas atasnya adalah *indexmax*. Perhatikan contoh skrip dengan *command Do* berikut.

```
In[1]:= Do[Print[i], {i,1,5}];
```

Dalam contoh yang diberikan di atas pengulangan dilakukan untuk sebuah ekspresi **Print** (cetak). Ekspresi **Print** adalah *command* baku dalam *Mathematica*[®] untuk perintah mencetak/menampilkan objek yang dalam contoh ini adalah **i**. Selain menjadi objek bagi **Print**, **i** adalah *indeks* dalam contoh tersebut dengan batas minimal (*indeksmin*) adalah 1 dan batas maksimal (*indeksmax*) adalah 5. Dengan demikian skrip pada input In[1]:= memiliki output Out[1]= adalah

```
Out[1]=
```

```
1  
2  
3  
4  
5
```

✓ *Command* Pengulangan **For**

Bentuk *loops* yang lain pada *Mathematica*[®] adalah **For**. Sama seperti halnya pengulangan dengan *command Do*, pengulangan **For** merupakan pengulangan dengan batas pengulangannya ditentukan. Pengulangan **For** memiliki perbedaan dengan **Do** dalam *Mathematica*[®] hal yang berbeda dengan pengulangan **Do** adalah

ukuran langkah pengulangan (*step size*) pada **For** bisa dimodifikasi sesuai kebutuhan. Berikut format pengulangan **For**.

For[*indeks = indeksmin, indeks (<,=,<=) indeksmax,*
indeks = indeks + ukuranlangkah,
ekspresi]

Jika dilihat skrip format **For** di atas bagian *ekspresi* merupakan bagian yang akan dilakukan pengulangan diletakkan setelah *indeks* dan syarat indeks (*indeksmin, indeksmax*) diberikan. Perhatikan contoh skrip dengan *command For* berikut.

```
In[2]:= For [ i=1, i<=5, i=i+2, Print[" ",i^2]]
```

Skrip pada input In[2]:= memperlihatkan peubah **i** adalah *indeks* dengan batas minimal (*indeksmin*) adalah 1 dan batas maksimal (*indeksmax*) adalah 5 serta ukuran langkah pengulangan adalah 2. Adapun ekspresinya adalah **Print** dengan objeknya adalah **i** dan **i²**. Dengan demikian skrip pada input In[2]:= memiliki output Out[2]= berupa susunan angka dalam baris-kolom seperti berikut ini.

```
Out[2]=  
  1  1  
  3  9  
  5 25
```

✓ *Command* Pengulangan **While**

Pernyataan pengulangan **While** secara umum digunakan untuk permasalahan pengulangan yang belum diketahui secara pasti berapa banyak proses pengulangan yang dibutuhkan agar pengulangan berlangsung sebagaimana yang diharapkan. Dengan kata lain pengulangan **While** biasa digunakan untuk suatu kondisi yang banyaknya pengulangan yang dibutuhkan tidak diketahui secara pasti. Pada pengulangan **While** bila kondisi sudah tidak terpenuhi maka pengulangan diakhiri. Itu artinya selama kondisi masih terpenuhi, pengulangan akan terus dilakukan. Bentuk umum Pernyataan pengulangan **While** adalah sebagai berikut:

While[kondisi,ekspresi]

Berikut ini beberapa contoh skrip *Mathematica*® dengan *command* **While**.

```
In[2]:= n=17;
While[(n=Floor[n/2])!=0,Print[n]];
n
8
4
2
1
Out[2]= 0
```

Catatan: berhati-hati dengan pemakaian simbol “!” dapat bermakna ganda dalam *Mathematica*® karena simbol itu dapat bermakna *factorial* jika berdiri sendiri. Jika diikuti tanda “=” dapat berarti *inequality*.

```
In[2]:= n=1; While[n<7,Print[n];n++];n (.....)
Out[2]= (.....)
```

```
In[3]:= n=1;
While[n<7,Print[n^2+n-1];n++];n (.....)
Out[3]= (.....)
```

✓ Pengulangan Khusus: Sum (*Sigma*) dan Product (*Pi*)

Dua buah bentuk pengulangan khusus yang terpasang pada *Mathematica*® adalah **Sum** dan **Product**. **Sum** dan **Product** merupakan pengulangan serupa **Do** tetapi pengulangan **Sum** dan **Product** masing-masing disertai dengan simbol jumlah “+” dan perkalian “x”. Bentuk umum *command* Jumlah dan Perkalian dalam *Mathematica*® dapat ditulis sbb:

Sum[expr, i, imin, imax]
Product[expr, i, imin, imax]

Misalkan ingin dievaluasi $\sum_{i=1}^{10} i!$ dan $\prod_{i=1}^{10} 2i$, maka dengan

menggunakan *Mathematica*[®] dapat dilakukan dengan skrip berikut ini:

In[1]:= Sum[i!,{i,1,10}] (.....)

Out[1]= (.....)

In[2]:= Product[2i,{i,1,10}] (.....)

Out[2]= (.....)

In[3]:= Sum[i^2,{i,1,10}] (.....)

Out[3]= (.....)

In[4]:= Sum[i^2,{i,1,n}] (.....)

Out[4]= (.....)

- **Mengekspresi suatu bagian dari bagian yang lain**

Ada beberapa ekspresi matematika yang diperoleh dari ekspresi lain. Umumnya ekspresi serupa ini mempunyai bagian-bagian misalnya:

- i. sebuah perkalian mempunyai faktor-faktor,
- ii. sebuah pembagian mempunyai pembilang dan penyebut,
- iii. sebuah daftar mempunyai entry,
- iv. sebuah himpunan mempunyai anggota himpunan, dan lain-lain.

Dengan *Mathematica*[®] pengekspresian bagian tertentu dari suatu bagian dapat dilakukan. Perhatikan skrip dalam demonstrasi berikut ini.

In[1]:= A={{1,2,3},{2,3,4}} (Mendefinisikan sebuah himpunan/list

Out[1]= {{1, 2, 3}, {2, 3, 4}} A dengan dua buah sub himpunan/list)

In[2]:= A[[2]] (Mengambil sub A yang ke dua)

Out[2]= {2, 3, 4}

In[3]:= A[[2,3]] (Mengambil sub A yang ke 2 untuk anggota/elemen ke 3)
Out[3]=4

In[4]:= A[[2,{1,3}]] Mengambil sub A yang ke 2 untuk anggota/elemen ke 1 dan ke 3)
Out[4]= {2, 4}

Bentuk skrip lain seperti penjumlahan dari dua buah factor yang masing-masing factor terdiri dari tiga peubah, katakanlah $S = abc+def$, dengan *Mathematica*[®] dapat diekspresikan menjadi

In[1]:= S=a b c+d e f (.....)
Out[1]= (.....)

In[2]:= S[[2]] (.....)
Out[2]= (.....)

In[3]:= S[[2,3]] (.....)
Out[3]= (.....)

In[4]:= S[[2,{1,3}]] (.....)
Out[4]= (.....)

Sebuah ekspresi pembagian misalnya a / b , *Mathematica*[®] mengartikan sebagai perkalian yaitu $a b^{-1}$.

In[1]:= q=a/b (.....)
Out[1]= $\frac{a}{b}$

In[2]:= FullForm[q] (.....)
Out[2]= FullForm=Times[a, Power[b, -1]]

In[3]:= Numerator[q] (.....)
Out[3]= a

In[4]:= Denominator[q] (.....)

Out[4]= b

In[5]:= p=2-7 x^3+11 x^5 (.....)

Out[5]= 2-7 x³ + 11 x⁵

In[6]:= Coefficient[p,x,3] (.....)

Out[6]= -7

In[7]:= Solve[2 x+3 == 5,x] (.....)

Out[7]= {{x→1}}

In[8]:= s=%[[1,1,2]] (.....)

Out[8]=1

In[9]:= s=x/.%1[[1]] (.....)

Out[9]=1

- **Penyederhanaan dan Pengambilan Bagian dari Suatu Bentuk Aljabar**

Sering dijumpai bentuk aljabar yang memerlukan penyederhanaan karena kebutuhan komputasi simbolis bukan komputasi numeris. *Mathematica*[®] mempunyai sejumlah pilihan *command* khusus yang dapat mengubah bentuk aljabar yang tidak sederhana menjadi sederhana. *Command* yang dimaksud diantaranya adalah:

Expand[*ekspresi*] ⇒ Menulis *ekspresi* dalam bentuk *term* per *term*

Factor[*ekspresi*] ⇒ Menulis sebuah *ekspresi* dalam bentuk faktor-faktor

N[*ekspresi*, *n*] ⇒ Menulis sebuah *ekspresi* dalam bentuk desimal dengan *n* digit

Together[*ekspresi*] ⇒ Menulis sebuah *ekspresi* menjadi bentuk fungsi rasional, $\frac{\text{numer}}{\text{denom}}$

Selain menyederhanakan bentuk aljabar, salah satu kelebihan yang dimiliki *Mathematica*[®] adalah kemampuan bekerja dengan

simbol dan mengoperasikan simbol. *Command* seperti **Coefficient**[], **Exponent**[], dan **Part**[] adalah tiga contoh diantaranya. Penjelasan penggunaan *command* ini diberikan dalam Tabel 7.

Tabel 7. *Command Mathematica*® untuk mengambil bagian tertentu dari sebuah ekspresi matematika simbolis

<i>Command Mathematica</i> ®	Keterangan.
Coefficient [<i>expr</i> , <i>form</i>]	Pengambilan semua koefisien dari <i>form</i> dalam <i>expr</i> .
Exponent [<i>expr</i> , <i>form</i>]	Pengambilan sebuah pangkat/eksponen tertinggi dari <i>form</i> dalam <i>expr</i> .
Part [<i>expr</i> , <i>n</i>] atau <i>expr</i> [[<i>n</i>]]	Pengambilan suku/term ke- <i>n</i> dalam <i>expr</i> .

Dan contoh penggunaan *command* ini adalah sebagai berikut.

$$\text{In}[1]:= e = \text{Expand}[(1+3x+4y^2)^2]$$

$$\text{Out}[1]= 1+6x+9x^2+8y^2+24xy^2+16y^4$$

$$\text{In}[2]:= \text{Coefficient}[e, x]$$

$$\text{Out}[2]= 6+24y^2$$

$$\text{In}[3]:= \text{Exponent}[e, y]$$

$$\text{Out}[3]= 4$$

$$\text{In}[4]:= \text{Part}[e, 4]$$

$$\text{Out}[4]= 8y^2$$

$$\text{In}[5]:= \text{Coefficient}[(1+3x+4y^2)^2, x] \quad (\dots\dots\dots)$$

$$\text{Out}[5]= (\dots\dots\dots)$$

$$\text{In}[6]:= r = (1+x)/(2(2-y)) \quad (\dots\dots\dots)$$

$$\text{Out}[6]= (\dots\dots\dots)$$

In[7]:= Denominator[%] (.....)

Out[7]= (.....)

In[8]:= Denominator[1/x + 2/y] (.....)

Out [8]= (.....)

Sebelum mengakhiri bagian ini, cobalah berlatih dengan menyelesaikan permasalahan berikut agar tercapai tujuan kemahiran yang telah dikemukakan sebelumnya.

Latihan

1. Tulis sejumlah skrip berikut dan beri keterangan tentang output yang diperoleh!
 - a. Do[Print[x],{2}]
 - b. Do[Print[i],{i,10,8,-1}]
 - c. Do[Print[{i,j}],{i,4},{j,i-1}] t=1;
 - d. Do[t*=k;Print[t];If[t>19,Break[]],{k,10}]
 - e. t=67; Do[Print[t];t=Floor[t/2],{3}]
2. Serupa dengan no .1
 - a. For[i=0, i<4, i++, Print[i]]
 - b. For[i=1; t=x, i^2<10, i++,t=t^2+i;Print[t]]
 - c. For[i=1, i<1000, i++, If[i>10,Break[]]];i
3. Serupa dengan no .1
 - a. n=1;While[n<4,Print[n];n++]
 - b. n=1;While[True,If[n>10,Break[]];n++]
 - c. n=1;While[++n<4];n
 - d. n=1;While[n<4,n++];n
4. Serupa dengan no .1
 - a. NestWhile[#/2&,123456,EvenQ]
 - b. NestWhile[Log,100,#>0&]
 - c. NestWhile[Floor[#/2]&,10,UnsameQ,2]
 - d. NestWhile[Floor[#/2]&,10,(Print[{##}];UnsameQ[##])&,All]

5. Misalkan diberikan sepasang barisan bilangan yang dikonstruksi menggunakan formula iteratif, $x_{i+1} = x_i + i$ dan $y_{i+1} = y_i - 2i$, untuk setiap $i \in N$. Untuk masukan (*input*) mula-mula, $x_0 = 1$ dan $y_0 = 2$. Dan untuk $i = 1, 2, 3, 4, 5$ kita mempunyai keluaran (*output*) sebagai berikut:

putaran ke 1 2 dan 0
 putaran ke 2 4 dan -4
 putaran ke 3 7 dan -10
 putaran ke 4 11 dan -18
 putaran ke 5 16 dan -28

Dengan menggunakan perintah pengulangan “Do” dapat dilakukan sebagai berikut.

```

mulai=1;
akhir=5;
x[1] :=1;
y[1] = 2;
Do[print["putaran ke ",i," ",x[i+1]=(x[i]+i),"
dan, y[i+1]=(y[i]-2*i)], {i,mula,akhir}]

```

Sayangnya, program di atas belum lengkap. Lengkapi bagian yang Anda anggap perlu sedemikian sehingga, program yang ditulis memberikan output sebagaimana yang diharapkan.

6. Buatlah program seperti yang ada pada soal no.5. dengan *Mathematica* tetapi menggunakan perintah “For”
7. Entri perintah seperti yang ada pada “In[1]:=” dan “In[2]:=” diiringi dengan menekan Shift-Enter dipenghujung masukan.

```
In[1]:= Plot[Exp[x]^4+2Cos[x],{x,-2,1}]
```

```
In[2]:= {a,b,c}={-1,3,(a+b)/2};i=0;tol=0.00000001;
```

```
f[x_]:=Exp[x]^4+2Cos[x];
```

```

For[i=1,i<15,If[f[a]*f[c]<0,{a,b}={a,c},{a,b}={c,b}];c=(a+b)
/2;Print[i,"
",N[a,8]," ",N[b,8]," ",N[c,8]" ",N[f[c],6];i++];
ClearAll;

```

- Ketika hasil running program berhasil Anda peroleh, modifikasi sintaks pada “In[2]:=” dengan menggunakan perintah “Do”
8. Gunakan penjelasan pada bagian integrasi sebelumnya untuk mengevaluasi integral fungsi $y = e^{2x}$.
 9. Dari skrip yang digunakan dalam soal 1, kembangkan untuk mengevaluasi integral dua kali fungsi $y = e^{2x}$.
 10. Dari skrip yang digunakan dalam soal 2, modifikasi skripnya untuk mengevaluasi integral berbatas $\int_0^1 e^{(2x+1)} dx$.
 11. Gunakan penjelasan pada bagian diferensiasi sebelumnya untuk mengevaluasi persamaan turunan orde 4 dari fungsi $y = e^{2x}$.
 12. Dari skrip yang digunakan dalam soal 1, kembangkan untuk mengevaluasi persamaan deferensial $\frac{d^4y}{dx^4} - \frac{d^2y}{dx^2} + y = 0$.
 13. Dari skrip yang digunakan dalam soal 2, modifikasi skripnya untuk mengevaluasi persamaan deferensial $\frac{d^2y}{dx^2} + y = 0$ dengan nilai awal $y(0) = 1$ dan $y'(0) = 2$.

MENJADI MAHIR-3

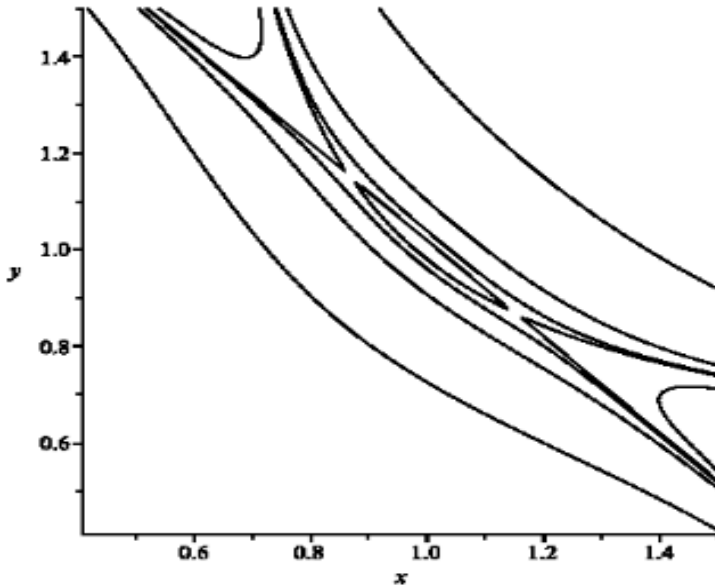
VISUALISASI FUNGSI/DATA

Setelah menyelesaikan materi ini Anda akan memperoleh kemampuan untuk melakukan operasi-operasi simbolis melalui pengerjaan visualiasasi (geometris) satu atau lebih fungsi yang memiliki satu atau lebih peubah baik dalam bentuk eksplisit maupun implisit. Kepahaman Anda dapat dikatakan melebihi 70% jika Anda mampu menyelesaikan beberapa latihan dibagian akhir materi ini sebesar 70% bernilai benar.

A. Arti Penting Visualisasi Fungsi/Data

Ada yang berpendapat bahwa “Sebuah gambar dapat memiliki 1000 informasi.”. Pendapat ini ada benarnya karena dalam sebuah gambar dapat memiliki berbagai macam informasi penting yang dimilikinya. Perhatikan Gambar 6. Visualisasi yang diperlihatkan dalam Gambar 6 bersumber dari sebuah pemetaan berikut ini: (lihat Zakaria dan Tuwankotta, 2016 untuk informasi lengkapnya.)

$$(x,y) \rightarrow \left(\frac{x(1-\mu x)}{y(x-1)}, x \right) \quad (*)$$



Gambar 6. Sebuah visualisasi pemetaan (*) disekitar titik (1,1) untuk sebuah nilai $\mu = 0.495$ menggunakan Mathematica® versi 9.**

** (Disalin dari artikel Zakaria dan Tuwankotta (2016) halaman 184).

Dari visualisasi yang diperlihatkan dalam Gambar 6 terdapat informasi antara lain, pertama titik (1,1) merupakan titik tetap (stasioner/*equilibrium*), kedua titik (1,1) merupakan titik tetap yang stabil, dan yang ketiga pemetaan (*) memiliki bifurkasi *period-doubling* disekitar titik (1,1).

Visualisasi sebuah data pengamatan/survey dalam sebuah kegiatan sensus seringkali dibutuhkan untuk mendapatkan informasi penting darinya. Misalnya hasil survei tentang pendapatan penduduk dalam satu tahun evaluasi dari sebuah populasi yang terdiri dari 100 individu selama 5 (lima) tahun sebagaimana dinyatakan dalam Tabel 8 berikut ini (Subian dkk, 2020).

Tabel 8. Daftar Distribusi Pendapatan 5 (lima) Individu

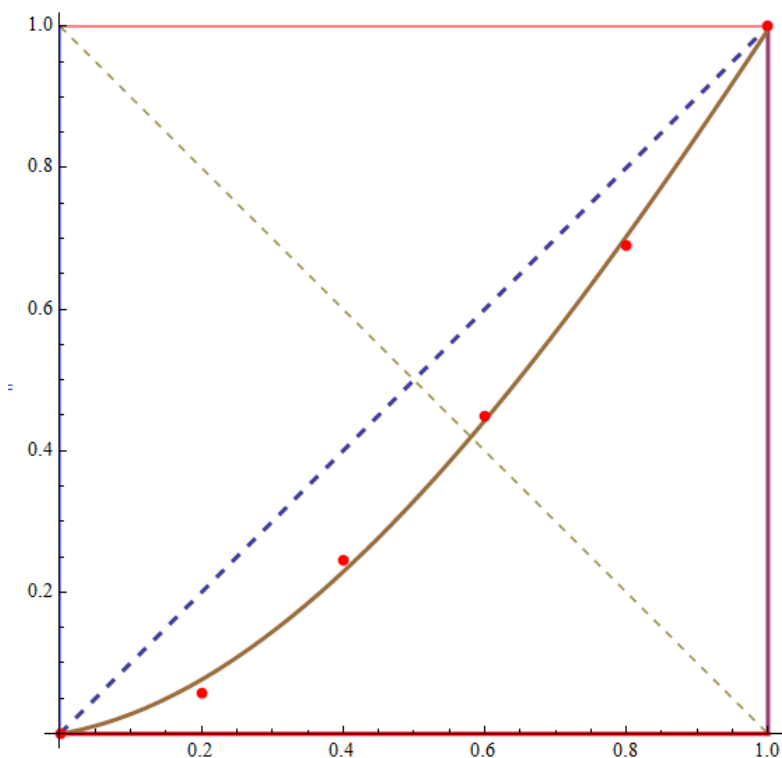
Individu	1	2	3	4	5
Pendapatan (US\$/Tahun)	2,417	7,800	8,489	10,072	12,957

Data ini diinterpretasikan bahwa setiap orang memiliki $(1/100)$ dari total pendapatan, grafik distribusi pendapatan yang mendasarinya adalah garis pemerataan. Asumsikan, distribusi pendapatan dibuat dari tingkatan miskin ke kaya. Ini berarti bahwa individu miskin memiliki kurang dari bagian yang sama dari total pendapatan karena individu yang lebih kaya memiliki lebih dari bagian yang sama. Dengan kata lain individu 1 memiliki US \$ 2,417 /tahun (dia adalah yang termiskin), sementara individu 5 memiliki US \$ 12,957/tahun (dia adalah yang terkaya). Proporsi pendapatan dan individu serta kumulatif masing-masing proporsi dapat dinyatakan sebagaimana diperlihatkan dalam Tabel 9.

Tabel 9. Proporsi pendapatan dan individu serta kumulatif masing-masing proporsi

Distribusi Pendapatan		Proporsi pendapatan yang dimiliki oleh masing-masing individu dan Proporsi populasi yang berkorespondensi dengannya		Proporsi kumulatif pendapatan yang dimiliki oleh masing-masing individu dan Proporsi kumulatif populasi yang berkorespondensi dengannya	
Individu	Pendapatan	Proporsi Pendapatan yang dimiliki masing-masing individu	Proporsi Setiap Individu Pada Total Populasi	Kumulatif Proporsi Pendapatan yang dimiliki masing-masing individu y	Kumulatif Proporsi Setiap Individu Pada Total Populasi x
1	2,417	0.0579	0.2000	0.0579	0.2000
2	7,800	0.1869	0.2000	0.2448	0.4000
3	8,489	0.2034	0.2000	0.4482	0.6000
4	10,072	0.2413	0.2000	0.6895	0.8000
5	12,957	0.3105	0.2000	1.0000	1.0000
Total	41,735	1.0000	1.0000	2.4405	3.0000

Data seperti yang diberikan dalam Tabel 9 akan dapat dipahami dengan baik bila disertai dengan bentuk geometrinya. Karena dari bentuk geometrinya akan dapat dengan mudah diinterpretasikan makna data tersebut. Data dalam Tabel 9 kolom ke-5 memiliki bentuk geometri sebagaimana yang diperlihatkan dalam Gambar 7



Gambar 7. Kurva yang bersumber dari data dalam Tabel 9.

Salah satu interpretasi yang dapat diberikan terhadap bentuk geometri dalam Gambar 7 adalah kurva data (warna coklat) diklaim tidak simetris yang berarti besaran ketidakmerataan pada kelompok masyarakat berpendapatan tinggi lebih rendah daripada besaran ketidakmerataan pada kelompok berpendapatan rendah.

B. Visualisasi Fungsi/Data dengan Mathematica®

Untuk kebutuhan visualisasi data, Mathematica® mempunyai fasilitas untuk menampilkan berbagai macam bentuk grafik/kurva baik jenis fungsi (satu peubah atau lebih yang berbentuk eksplisit atau implisit) maupun dimensi fungsi (dua atau tiga dimensi). Dua *command* Mathematica® yang dapat digunakan untuk visualisasi data adalah

Plot[*expresi1*, {x, a, b}]

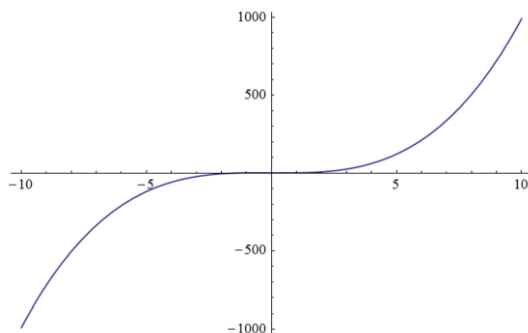
Plot3D[*expresi2*, {x, a, b}, {y, c, d}]

Perhatikan sejumlah input skrip *Mathematica*[®] berikut ini:

- Menvisualisasikan grafik dari *expresi1* berupa sebuah fungsi satu peubah bebas x pada interval $a \leq x \leq b$. Grafik divisualisasi dalam 2 dimensi menggunakan command **Plot**

```
In[1]:= Plot[x^3 - x + 1, {x, -10, 10}]
```

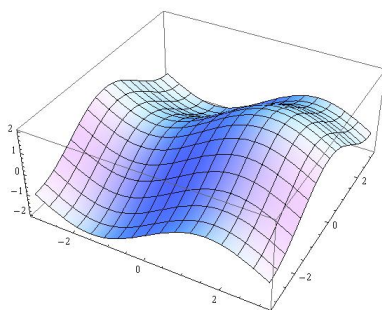
Out[1]=



- Menvisualisasikan grafik dari *expresi1* dengan dua peubah x dan y bebas pada interval $a \leq x \leq b$ dan $c \leq y \leq d$. Grafik divisualisasi dalam 3 dimensi menggunakan command **Plot3D**

```
In[2]:= Plot3D[Sin[x]+Cos[y], {x, -Pi, Pi}, {y, -Pi, Pi}]
```

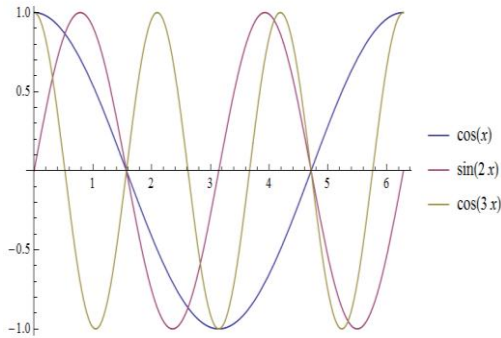
Out[2]=



```
In[3]:= Plot[{Cos[x],Sin[2x],Cos[3x]},  
            {x,0,2Pi},  
            PlotLegends->"Expressions"]
```

(Memplot beberapa grafik fungsi beserta keterangannya (*legend*))

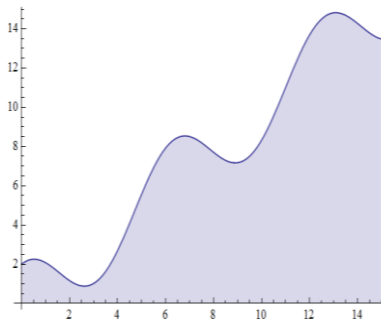
Out[3]=



```
In[4]:= Plot[2Cos[x]+x,{x,0,15},  
            Filling->Bottom]
```

Memplot grafik fungsi dan memberikan arsiran antara kurva dan sumbu-x

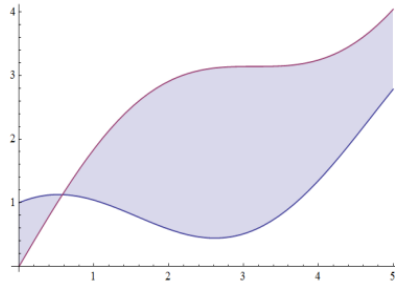
Out[4]=



```
In[6]:= Plot[{Cos[x]+x/2,Sin[x]+x},  
           {x, 0, 5},  
           Filling->{1->{2}}]
```

Memplot grafik fungsi
dan memberikan
arsiran antara
keduanya)

Out[6]=



Selain tampilan grafik fungsi seperti di atas, *Mathematica*® menyediakan banyak fasilitas untuk penanganan grafik fungsi. *Mathematica*® juga dapat mengelola berbagai macam grafik dalam bentuk 2D maupun 3D. Fasilitas *image processing* yang disediakan *Mathematica*® juga dapat bekerja dengan grafik dalam grafik atau gambar dalam gambar. Untuk mengetahui lebih lanjut tentang hal ini, Anda bisa “bersafari” dengan menelusuri *Help browser* yang ada di *Mathematica*® dan/atau bergabung dengan komunitas pengguna *Mathematica*® melalui laman *stackexchange* di internet.

Di akhir bagian ini, seperti biasa Anda diberikan latihan untuk membiasakan diri Anda dengan sejumlah *command* yang telah Anda ketahui sebelumnya.

Latihan

1. *Mathematica*® menyediakan beberapa cara untuk menggambar grafik fungsi satu peubah. Dalam latihan berikut ini, Anda diminta untuk menulis kembali skrip *command* yang ada untuk kemudian dieksekusi dan amati *output* program *Mathematica*® berdasarkan skrip yang diberikan. Seraharusnya *output* berupa grafik fungsi (kurva) yang dihasilkan dari pemplotan biasa, logaritma, parameter, dan polar, jika tidak ada kesalahan input skrip yang Anda buat. Selamat mencoba.

- Eksekusi dan perhatikan *command* **Plot[]** untuk sebuah fungsi $\cos(x)$ dengan batas nilai $0 < x < 2\pi$.

```
In[1]:= Plot[Cos[x],{x,0,2Pi}]
Out [1]= (.....)
```

- Eksekusi dan perhatikan *command* **Plot[]** untuk beberapa fungsi diekspresikan dalam bentuk *list* dengan batas nilai $0 < x < 2\pi$.

```
In[2]:= Plot[{x^2,Sin[x],Sqrt[x]},{x,0,2Pi}]
Out [2]= (.....)
```

- Eksekusi dan perhatikan *command* **Plot[]** terhadap fungsi eksponensial e^{x^3} yang dilakukan melalui penskalaan sumbu vertikal berupa nilai Logaritma dengan batas nilai $1 < x < 5$.

```
In[3]:= LogPlot[E^x^3,{x,1,5}]
Out [3]= (.....)
```

- Eksekusi *command* **Plot[]** dari sebuah fungsi hiperbolik tangen, $\tanh(x)$, yang dilakukan untuk nilai dari Logaritma melalui penskalaan pada sumbu horizontal dengan batas nilai $1 < x < 50$.

```
In[4]:= LogLinearPlot[Tanh[x],{x,1,50}]
Out [4]= (.....)
```

- Eksekusi *command* **Plot[]** dari fungsi $e^{x^{\frac{1}{2}}}$ yang dilakukan dengan penskalaan sumbu vertikal dan horizontal berupa nilai dari log dengan batas nilai $1 < x < 5$.

```
In[5]:= LogLogPlot[E^x^(0.5),{x,1,5}]
Out [5]= (.....)
```


- Melakukan komputasi terhadap persamaan parameter satu peubah yang dinyatakan dalam bentuk $(x, y) = (3 \sin(2t), \cos(t))$ dengan batas nilai $0 < t < 2\pi$ menggunakan *command* **ParametricPlot[]**.

In[6]:= ParametricPlot[{3 Sin[2 t], Cos[t]}, {t, 0, 2Pi}]
 Out [6]= (.....)

- Melakukan komputasi terhadap persamaan parameter satu peubah yang dinyatakan dalam bentuk $(x, y, z) = (2 \cos(u), 3 \sin(u), u + \sin(2u))$ dengan batas nilai $-2\pi < u < 2\pi$ menggunakan *command* **ParametricPlot3D[]**.

In[7]:= ParametricPlot3D[{2 Cos[u], 3 Sin[u], u+Sin[2 u]}, {u, -2Pi, 2Pi}]
 Out [7]= (.....)

- Melakukan komputasi fungsi $\cos^2(t)$ dengan batas nilai $0 < t < 2\pi$ dalam koordinat polar dan memplotkannya menggunakan *command* **ContourPlot[]**:

In[8]:= PolarPlot[Cos[t]^2, {t, 0, 2Pi}]
 Out [8]= (.....)

- Melakukan komputasi sebuah persamaan dalam bentuk implisit untuk mendapatkan kurva *contour* dengan menggunakan *command* **ContourPlot[]**:

In[9]:= ContourPlot[x^3+x y^2+y==0, {x, -1, 1}, {y, -10, 10}]
 Out [9]= (.....)

2. Selain menggambar grafik sebuah fungsi, *Mathematica*® menyediakan beberapa cara untuk menggambar data hasil observasi berupa diagram (histogram, pie-diagram, atau lainnya). Beberapa latihan berikut ini memberikan Anda pengalaman untuk menangani pengelolaan diagram dari suatu data. Salin skrip pada **In:=**[] pindahkan ke *notebook Mathematica*® lalu eksekusi. Perhatikan Outputnya.

- Label diberikan pada elemen data yang bersesuaian.
 In:=[10] BarChart[{1,2,3,4},ChartStyle->"Pastel",
 ChartLegends->{"Mary","Bob","Susan","John"}]
 Out [10]= (.....)
- *Command* "None" digunakan untuk menghilangkan sebuah elemen data dari *legend*, dan ia ditempatkan pada label data yang dimaksud.
 In:=[11] BarChart[{1,2,3,4},ChartStyle->"Pastel",
 ChartLegends->{None,"Bob",None,"John"}]
 Out [11]= (.....)
- Kelompok data dapat ditampilkan dalam *legend*nya.
 In:=[12] BarChart[{{1,2,3},{4,5,6}},ChartLegends->
 {"Group A","Group B"},None},ChartStyle->
 {"Pastel",None}]
 Out [12]= (.....)
- Penggunaan *command* "**ChartLabels**" untuk membuat pengulangan *labels* pada elemen data dalam setiap grup.
 In:=[13] BarChart[{{1,2,3},{4,5,6}},ChartLegends->
 {"Group A","Group B"},None},ChartStyle->
 {"Pastel",None},ChartLabels->{"r1","r2","r3"}]
 Out [13]= (.....)
- Penggunaan *command* "**LegendAppearance**" untuk mengubah tampilan *labels* dalam *legend*.
 In:=[14] BarChart[{{1,2,3},{4,5,6}},ChartLegends->
 {"Group A","Group B"},None},ChartStyle->
 {"Pastel",None},ChartLabels->

```

{"r1","r2","r3"},LegendAppearance->"Row"
Out [14]= (.....)

```

- Alternatif lainnya, penggunaan “**Placed[]**” untuk menempatkan *legend* berseberangan dari bagian bawah diagram.

```

In:=[15] BarChart[{{1,2,3},{4,5,6}},ChartLegends->
  {Placed[{"Group A","Group B"},Bottom],None},
  ChartStyle->{"Pastel",None},ChartLabels->
  {"r1","r2","r3"}]
Out [15]= (.....)

```

- Command “**ChartLegends**” juga dapat dipakai untuk diagram 3D.

```

In:=[16] BarChart3D[{{1,2,3},{4,5,6}},ChartLegends->
  {Placed[{"Group A","Group B"},Bottom],None},
  ChartStyle->{"Pastel",None},ChartLabels->
  {"r1","r2","r3"}]
Out [16]= (.....)

```

- Command “**Legended[]**” dapat digunakan untuk memberikan keterangan *legend* secara langsung untuk elemen spesifik data pada diagram yang dibuat.

```

In:=[17] PieChart[{1,Legended[2,"Bob"]},3,
  Legended[4,"John"]}]
  BarChart[{1,Legended[2,"Bob"]},3,
  Legended[4,"John"]},ChartStyle->"Pastel"]
Out [17]= (.....)

```

- Command “**Legended[]**” dapat bekerja pada pembuatan diagram fungsi 3D.

```

In:=[18] Row[{PieChart3D[{1,Legended[2,"Bob"]},
  3,Legended[4,"John"]}]},
  {BarChart3D[{1,Legended[2,"Bob"]},
  3,Legended[4,"John"]},ChartStyle->"Pastel"]}]]
Out [18]= (.....)

```

MENJADI MAHIR-4

OPERASI ARITMATIKA SIMBOLIS

Komputasi numerik yang melibatkan simbol-simbol sering dijumpai dalam bidang fisika atau kimia dan juga dalam bidang matematika (Kalkulus). Ukuran satuan metriks, operator sebuah transformasi fungsi, dan pengelolaan deret dalam bentuk penjumlahan (**Sum**) atau dalam bentuk perkalian (**Product**) merupakan beberapa contoh diantaranya. Dalam bagian ini Anda akan diperkenalkan beberapa teknis menggunakan *Mathematica*[®] untuk mengelola simbol yang terlibat dalam sebuah komputasi numerik. Setelah menyelesaikan bagian ini Anda akan dapat melakukan komputasi numerik berkenaan dengan keterlibatan satuan metriks dalam sebuah komputasi, melakukan sejumlah operasi aritmatika terhadap suatu deret (**Sum** dan **Product**), melakukan transformasi (*Laplace*) yang tersedia berupa paket dalam *Mathematica*[®].

A. Simbol/Lambang sebagai Tag Objek

Sering dijumpai untuk penanganan komputasi dalam ilmu fisika/kimia menggunakan sistem metrik (ukuran satuan), misalnya *meter*, *feet*, *kilogram*, dan lain sebagainya. *Mathematica*[®] menyediakan fasilitas untuk melakukan penulisan sistem metriks secara langsung. Untuk melakukan hal ini *command* yang dapat digunakan dalam *Mathematica*[®] adalah **Quantity**[] yang format penulisan skripnya adalah sebagai berikut.

Quantity[ukuran, "satuan"]

Berikut ini beberapa contoh skrip *Mathematica*[®] untuk penanganan simbol satuan metrik yang tersedia. Perhatikan operasi aritmatika yang dilibatkan.

In [1] := Quantity[29.25,"Kilometers"] Mendefinisikan nilai 19.25 dalam satuan metrik kilometer-km)
 Out [1]= 29.25 km

In [2] := Quantity[29.25,"Kilometers"]+ Melakukan operasi aritmatika
 Quantity[23,"Miles"] penjumlahan terhadap dua nilai numerik dengan satuan yang berbeda SI dan *Imperial*).
 Out [2]= 41.1751 mi

In[3]:= UnitConvert[41.1751 mi,"Metric"] Melakukan konversi satuan dari *imperial* ke SI.
 Out[3]= 66.265 km

In [4] := Quantity[30,"Feet"]/ (.....)
 Quantity[3.3,"Seconds"]+
 Quantity[19,"Meters"/"Seconds"]
 Out [5]= (.....)

In [6] := N[Quantity[155, "Pounds"]/ (.....)
 Quantity[120, "Kilograms"]]
 Out [6]= (.....)

In [7] := Quantity[24,"Miles"/"Gallons"]* (.....)
 Quantity[16.2,"Gallons"/
 ndependentUnit["tank"]]*
 Quantity[1/8,IndependentUnit["tank"]]
 Out [7]= (.....)

Mathematica[®] dapat juga menyelesaikan tugasnya walau *command* **Quantity**[] tidak disertai dalam skrip sebagaimana contoh sebelumnya. Perhatikan peragaan berikut ini:

In [1] := 12 meters Out [1]= 12 meters	Mendefinisikan sebuah nilai ukuran yang disertai <i>tag</i> tanpa peubah
In [2] := % + 5.3 meters Out [2]= 17.3 meters	Melakukan komputasi dengan melibatkan terhadap hasil akhir Out=[1].
In [3] := %/(25 second)	Melakukan komputasi dengan melibatkan
Out [3]= $\frac{0.692 \text{ Meters}}{\text{Second}}$	hasil akhir Out=[2] dibagi dengan nilai numerik bersatuan waktu (<i>second</i>) sehingga diperoleh satuan untuk kecepatan/ <i>speed</i>
In [4] := % /.meters → 3.28084feet Out [4]= (.....)	(.....)
In [5] := 12 meters/second Out [5]= (.....)	(.....)
In [6] := Convert[%,Mile/Hour] Out [6]= (.....)	(.....)
In [7]= Convert[3 Kilometers/Hour, Inch/Minute] Out [7]= (.....)	(.....)

B. Penjumlahan (*Sum*) dan Perkalian (*Product*) suatu Barisan Bilangan

Pandang sebuah himpunan beranggotakan bilangan berhingga atau takberhingga yang biasa disebut barisan bilangan, misalnya $A = \left\{1, \frac{1}{2}, \frac{1}{3}, \dots, \frac{1}{n+1}\right\}$ dan $B = \left\{1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \dots\right\}$. Menjumlahkan setiap anggota himpunan B atau mengalikan setiap anggota himpunan A yang dalam notasi matematikanya masing-masing dapat dinyatakan

dalam bentuk $J = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5}, \dots$ dan $K = 1 \times \frac{1}{2} \times \frac{1}{3} \times \dots \times \frac{1}{n+1}$.

Dalam notasi sederhananya masing-masing dapat dinyatakan dalam bentuk **Sum** (Σ) dan **Product** (Π). Dengan kata lain J dan K masing-masing dapat ditulis sebagai $J = \sum_{i=1}^{\infty} \frac{1}{i}$ dan $K = \prod_{i=1}^{n+1} \frac{1}{i}$. Sisi

kanan dari J dan K biasa disebut **Deret**. Untuk melakukan komputasi dengan nilai perubahan indeks i yang relatif besar maka proses untuk mendapatkan nilai numerik dari deret J dan K memerlukan alat agar pekerjaan menjadi efisien dan efektif. *Mathematica*[®] mampu menjumlahkan dan mengalikan suatu deret konvergen. *Command* yang digunakan adalah **Sum[]** dan **Product[]**. Perhatikan skrip yang menggunakan *command* **Sum[]** dan **Product[]** berikut ini:

In [1] := Sum[xⁱ/i, {i, 1, 7}]

$$\text{Out [1]} = x + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{4} + \frac{x^5}{5} + \frac{x^6}{6} + \frac{x^7}{7}$$

Mendefinisikan sebuah penjumlahan berhingga fungsi polinomial x^i/i .

In [2] := Product[xⁱ/i, {i, 1, 7}]

$$\text{Out [2]} = x \times \frac{x^2}{2} \times \frac{x^3}{3} \times \frac{x^4}{4} \times \frac{x^5}{5} \times \frac{x^6}{6} \times \frac{x^7}{7}$$

Mendefinisikan sebuah perkalian berhingga fungsi polinomial x^i/i .

In [3] := Sum[(x+1)ⁱ/i, {i, 1, 5, 2}] (.....)

Out [3]=(.....)

In [4] := product[x+i, {i, 1, 4}] (.....)

Out [4]= (.....)

In [5]:= Expand [Product[x+i,{i,3}]] Kalkulasi perkalian
 Out [5]= 6+11x+6x²+x³ (x+1)(x+2)(x+3)
 dan mengekspansikan hasilnya.

In [6]:= Expand[Product[x+i,{i,4}]] (.....)
 Out [6]= (.....)

In [7] :=
 polinomx[n_] := Expand [
 Product[x+i,{i,1,n}]] Mendefinisikan sebuah fungsi
 bernama *polinomx* dengan
 Out [7]= (.....) peubah bebas n.

In [8]:= polinomx[5] (.....)
 Out [8]= (.....)

Format baku skrip *Mathematica*® dalam pemakaian *command* **Sum** dan **Product** untuk penjumlahan dan perkalian dalam bentuk deret diberikan dalam Tabel 10.

Tabel 10. *Command Mathematica*® untuk penjumlahan dan perkalian elemen suatu barisan (deret) .

Notasi Matematika	Notasi Mathematica®
$\sum_{i=i \min}^{i \max} f$	Sum [<i>f</i> , { <i>i</i> , <i>i min</i> , <i>i max</i> }] Sum [<i>f</i> , { <i>i</i> , <i>i min</i> , <i>i max</i> , <i>di</i> }]
$\sum_{i=i \min}^{i \max} \sum_{j=j \min}^{j \max} f$	Sum [<i>f</i> , { <i>i</i> , <i>i min</i> , <i>i max</i> , <i>j</i> , <i>j min</i> , <i>j max</i> }]
$\prod_{i=i \min}^{i \max} f$	Product [<i>f</i> , { <i>i</i> , <i>i min</i> , <i>i max</i> }]

Berikut ini diberikan beberapa contoh skrip *Mathematica*® dalam pemakaian *command* **Sum** dan **Product**.

In [5] := Sum[i^2,{i,1,n}] (.....)

Out [5] := (.....)

In [6] := Sum[1/i^4,{i,1,Infinity}] (.....)

Out [6] := (.....)

In [7] := Sum[1/(i^4+2),{i,1,Infinity}] (.....)

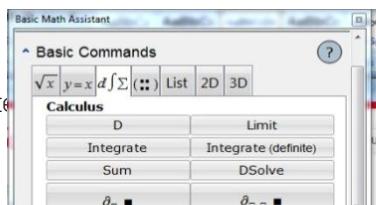
Out [7]= (.....)

In [8] := Expand[%] (.....)

Out [8]= (.....)

Anda dapat memanfaatkan fasilitas *Palettes* untuk menulis simbol **Sum** dan **Product** (atau simbol lainnya yang tersedia di *Mathematica*(@). Caranya buka menu *Palettes* kemudian pilih *Basic Math Assistant* dan *Basic command* akan tampil submenu dalam sebuah *window* tersendiri (lihat Gambar 8).

Cobalah dengan menggunakan submenu sebagaimana yang diperlihatkan dalam Gambar 8 untuk beberapa skrip *Mathematica*® yang telah dikemukakan di atas (input In[2]:= hingga In[8]:=). Kemudian coba eksekusi dan lihat outputnya, apakah bersesuaian dengan Out[2]= hingga Out[8]=? Jika tidak, ulangi lagi hingga Anda menemukan kesamaan hasilnya.



Gambar 8. Tampilan Submenu Palettes-Basic Math Assistant-Basic Command dan Type Setting

Dalam hal khusus, sebagaimana diketahui bahwa bentuk **Sum[]** yang telah diperlihatkan sebelumnya menghasilkan bentuk penjumlahan dalam pola tertentu yang dalam Kalkulus atau pemakaiannya biasa disebut dengan **deret/series**, misalnya deret Geometri atau Aritmatika yang suku-sukunya berupa bilangan real atau deret Taylor dan McLaurin yang suku-sukunya berupa peubah/variabel. *Mathematica*® menyediakan *command* khusus untuk deret yaitu **Series[]**. Skrip yang melibatkan *command* fasilitas **Series[]** adalah sebagai berikut:

Series[ekspresi,{peubah,batas_bawah,batas_atas}]

Beberapa pemakaian **Series[]** diberikan dalam penjelsan berikut ini.

- ✓ **Series[]** dipakai dalam deret pangkat/deret kuasa dari sebuah fungsi dipersekitaran titik $x = 0$. Deret serupa ini biasa disebut dengan deret **Taylor** satu peubah untuk x disekitar nol atau biasa disebut juga deret **McLaurin**.

In [1]:= Series[2 Exp[x],{x,0,5}]

Out [1]= $2 + 2x + x^2 + \frac{x^3}{3} + \frac{x^4}{12} + \frac{x^5}{60} + O[x]^6$

- ✓ Suku sisa (residu), jika tidak diinginkan bisa dihilangkan dengan menggunakan *command* **Normal[]**.

In [2]:= Normal[%]

$$\text{Out [2]} = 2 + 2x + x^2 + \frac{x^3}{3} + \frac{x^4}{12} + \frac{x^5}{60}$$

- ✓ **Series[]** dipakai dalam deret pangkat/deret kuasa dari sebuah fungsi dipersekitaran titik tertentu, misalnya $x = a$.

In [3]:= Series[f[x],{x,a,4}]

$$\begin{aligned} \text{Out [3]} = & f[a] + f'[a](x-a) + \frac{1}{2}f''[a](x-a)^2 + \\ & \frac{1}{6}f^{(3)}[a](x-a)^3 + \frac{1}{24}f^{(4)}[a](x-a)^4 + O[x-a]^5 \end{aligned}$$

- ✓ **Series[]** dapat digunakan dalam sebarang operasi aritmatika dan tetap mempertahankan bentuk suku persuku.

In [4]:= Series[Cos[2t]/t,{t,0,5}]

$$\text{Out [4]} = \frac{1}{t} - 2t + \frac{2t^3}{3} - \frac{4t^5}{45} + O[t]^6$$

In [5]:= 1/(1+%)+%^2

$$\text{Out [5]} = \frac{1}{t^2} - 4 + t + \frac{13t^2}{3} + 3t^3 - \frac{353t^4}{45} + O[t]^5$$

- ✓ **Series[]** dapat digunakan bersamaan dengan pangkat pecahan dan logaritma

In [6]:= Series[Sqrt[Cos[2 t]],{t,0,5}]

$$\text{Out [6]} = 1 - t^2 - \frac{t^4}{6} + O[t]^6$$

In [7]:= Series[Log[Cos[2 t]],{t,0,5}]

$$\text{Out [7]} = -2t^2 - \frac{4t^4}{3} + O[t]^6$$

- ✓ Pemakaian pada simbol untuk suatu parameter

In [8] := Series[(1+x)^n,{x,0,3}]

$$\text{Out [8]} = 1 + nx + \frac{1}{2}(-1+n)nx^2 + \frac{1}{6}(-2+n)(-1+n)nx^3 + O[x]^4$$

C. **Mathematica[®]** untuk Matematika Simbolis

Sebagai perangkat lunak *engineering*, *Mathematica*[®] dipersiapkan untuk menangani masalah-masalah khusus seperti Transformasi *Fourier* dan Transformasi *Laplace* yang biasa digunakan dalam bidang teknik/rekayasa. Salah satu sifat transformasi adalah ia bekerja antar simbol dengan tujuan agar mudah disederhanakan bentuknya dan sederhana pula dalam menyelesaikan model matematisnya. Dalam bagian ini diperlihatkan penanganan simbolis dengan menggunakan *Mathematica*[®] melalui transformasi *Laplace*, *Fouries*, dan beberapa simbol berkenaan dengan analisis vektor dalam Kalkulus. *Command-command* yang dimaksud dijelaskan dalam bagian berikut ini.

- **Transformasi *Laplace***

Salah satu transformasi yang relatif sering dipakai dalam menangani persamaan/sistem persamaan diferensial adalah transformasi *Laplace*. *Command Mathematica*[®] berkenaan dengan transformasi ini dijelaskan dalam Tabel 9.

Tabel 9. *Command Mathematica*[®] untuk pengaktifan paket transformasi *Laplace* dan inversinya.

<i>Command Mathematica</i> [®]	Keterangan
LaplaceTransform [<i>expr</i> , <i>t</i> , <i>s</i>]	Transformasi <i>Laplace</i> dari <i>expr</i> dengan peubah bebas <i>t</i> .
InverseLaplaceTransform [<i>expr</i> , <i>s</i> , <i>t</i>]	Invers dari Transformasi <i>Laplace</i> dengan peubah bebas <i>s</i> yang bersesuaian dengan peubah <i>t</i> .

Skrip untuk pemakaian *command* yang diperlihatkan dalam Tabel 9, beberapa diantaranya diberikan dalam barisan berikut ini.

In [1]:= << Calculus `LaplaceTransform`

In [2]:= LaplaceTransform[t^3 Exp[a, t], t, s]

$$\text{Out [2]} = \frac{6}{(-a+s)^4}$$

In [3]:= InverseLaplaceTransform [%, s, t]

$$\text{Out [3]} = E^{at} t^3$$

- **Transformasi Fourier**

Transformasi lain yang relatif sering dipakai terutama dalam menganalisa trend data yang melibatkan deret waktu adalah transformasi *Fourier*. *Command Mathematica*® berkenaan dengan transformasi ini dijelaskan dalam Tabel 10.

Tabel 10. *Command Mathematica*® untuk pengaktifan paket transformasi *Fourier* dan inversinya.

<i>Command Mathematica</i> ®	Keterangan
FourierTransform [<i>expr</i> , <i>t</i> , <i>s</i>]	Transformasi <i>Fourier</i> dari <i>expr</i> dengan peubah bebas <i>t</i> .
InverseFourierTransform [<i>expr</i> , <i>s</i> , <i>t</i>]	Invers dari Transformasi <i>Fourier</i> dengan peubah bebas <i>s</i> yang bersesuaian dengan peubah <i>t</i> .

Skip untuk pemakaian *command* yang diperlihatkan dalam Tabel 10, beberapa diantaranya diberikan dalam bagian berikut ini.

In [1]:= <<Calculus`FourierTransform`

In [2]:= FourierTransform [t^2 Sin[t], t, w]

Out [2]= $-I \pi (\text{DiracDelta}[1-w] - \text{DiracDelta}[1+w])$

D. Paket Kalkulus-Analisis Vektor

Topik analisis vektor dalam kalkulus peubah banyak memerlukan proses komputasi yang tidak sederhana. Selain menggunakan konsep geometri juga melibatkan konsep persamaan diferensial. Oleh karena itu, komputasi secara simbolis memerlukan tingkat konsentrasi pada bentuk persamaan matematika lebih tinggi dibandingkan dengan konsentrasi pada hasil operasi aritmatika yang dihasilkan oleh persamaan tersebut. Oleh karena itu, proses penyederhaan persamaan matematika dan menyelesaikannya dapat menggunakan alat komputasi yang dapat menangani simbol-simbol seperti halnya *Mathematica*®. Beberapa *command Mathematica*®

berkenaan dengan kalkulus-analisis vektor ini dijelaskan dalam Tabel 11.

Tabel 11. *Command Mathematica*[®] untuk pengaktifan paket *Kalkulus-Analisis Vektor*.

<i>Command Mathematica</i> [®]	Keterangan
<code><< Calculus `VectorAnalysis`</code>	Mengaktifkan paket <i>Kalkulus Analisis Vektor</i>
<code>SetCoordinates[System[Names]]</code>	Menset sistem koordinat yang akan digunakan (Kartesian atau Bola atau lainnya).
<code>Grad [f]</code>	Menghitung <i>Gradient</i> dari f yang dinotasikan dengan ∇f
<code>Div [f]</code>	Menghitung <i>Divergent</i> dari f yang dinotasikan dengan $\nabla \cdot f$
<code>Curl [f]</code>	Menghitung <i>Curl</i> dari f yang dinotasikan dengan $\nabla \times f$
<code>Laplacian [f]</code>	Menghitung <i>Laplacian</i> dari f yang dinotasikan dengan $\nabla^2 f$

Skrip untuk pemakaian *command* yang diperlihatkan dalam Tabel 11, beberapa diantaranya diberikan dalam ilustrasi berikut ini.

```
In [1]:= <<Calculus `VectorAnalysis`      Mengaktifkan paket Calculus
                                         `VectorAnalysis`
In [2]:= SetCoordinates[Spherical [      Menspesifikasikan sebuah
r,theta,phi ] ]                          koordinat
                                         dengan nama  $r$ ,  $\theta$ , dan  $\phi$ 
Out [2]= Spherical [r,theta,phi ]
In [3]:= Grad[r ^ 2Sin[theta]] Mengevaluasi gradient  $r^2 \sin \theta$ 
Out [3]= {2rSin[theta],rCos[theta],0}
```

E. Mendefinisikan Fungsi dengan *Mathematica*[®]

Sebagaimana diketahui bahwa hubungan fungsional yang ditemukan dalam model-model matematika tidak selalu dalam bentuk standar. *Mathematica*[®] menyediakan banyak bentuk fungsi standard dan sebagian fungsi-fungsi nonstandar dalam sebuah model matematika yang populer, fungsi Bessel dan fungsi *Gamma* misalnya. Untuk fungsi yang tidak tersedia, *Mathematica*[®] dapat membantu mendisain fungsi-fungsi yang dimaksud karena *Mathematica*[®] mampu bekerja dengan simbol-simbol. Sebagai contoh, *Mathematica*[®] menyediakan **Sqr[x]** untuk menyatakan sebuah fungsi kuadrat $f(x) = x^2$. Akan tetapi jika dibutuhkan pengantian nilai x berupa simbol $(a + 1)$ atau simbol-simbol bentuk lain perlu pendefinisian lanjutan. Perhatikan skrip berikut ini.

In [1]:= f[x_]:=x^2	Mendefinisikan sebuah fungsi $f(x) = x^2$
In [2]:= f[a+1]	Melakukan komputasi simbolis terhadap
Out [2]= (1+a) ²	fungsi $f(x) = x^2$ untuk suatu $x = a + 1$.
In [3]:= f[4]	Melakukan komputasi numeris pada
Out [3]= 16	$f(x) = x^2$ untuk suatu $x = 4$.
In [4]:= f[3x+x^2]	(.....)
Out [4]= (.....)	

Perhatikan skrip pada In[4]:= di atas. Bentuk ini merupakan pendefinisian fungsi komposisi $f \circ g(x)$ dengan $g(x) = 3x + x^2$ dan $f(x) = x^2$. Perhatikan juga skrip berikut ini.

In [5]:= Expand [f[(x+1+y)]]	Mengkalkulasi nilai $f(x) = x^2$ untuk nilai
Out [5]= 1+2x+x ² +2y+2xy+y ²	$x = x + 1 + y$ dan mengekspansikannya.
In [6]:= Clear [f]	Membebaskan semua definisi-definisi
	untuk f .

Untuk notasi fungsi dapat dibuat sesuai kebutuhan Akan tetapi perlu menjadi perhatian natosi tidak dibenarkan sama dengan

command baku yang telah didefinisikan oleh Mathematica®, misalnya ingin dibuat notasi fungsi untuk rata-rata. Notasi **ratarata** atau **Ratarata** atau **mean** masih dibolehkan. Akan tetapi **Mean** tidak dibolehkan karena ia *command* baku dalam Mathematica®. Notasi fungsi dalam ilustrasi berikut masih dibolehkan.

$$hump(x, xmax) = \frac{(x - xmax)^2}{xmax}$$

Perhatikan bahwa notasi fungsi adalah *hump* dan peubah-peubahnya adalah *x* dan *xmax*. Skrip Mathematica® untuk fungsi *hump(x, xmax)* adalah sebagai berikut:

```
In [7]:= hump[x_,xmax_]      Mendefinisikan fungsi dua peubah
                               (x - xmax)^2 / xmax  hump(x, xmax).
In [8]:= 2+hump[x,3.5]      Penjumlahan yang melibatkan fungsi
                               hump untuk x = x dan xmax = 0.35
Out [8]= 2+0.285714(-3.5+x)^2
In [9]:= Clear[hump]       Membebaskan fungsi hump dari
                               seluruh nilai yang telah ada sebelum.

In [10]:= hump1[y_,ymax_] = (.....)
                               (y - ymax)^4
In [11]:= 2+hump1[t,2.35]  (.....)
Out [11]= (.....)

In [12]= Clear[hump1]     (.....)
```

F. Membuat Prosedur

Dalam kegiatan pemrograman prosedur memiliki sifat seperti sebuah fungsi tetapi berbeda peranan. Jika fungsi merupakan modul program yang memberikan atau mengembalikan sebuah nilai dari tipe tertentu, sebaliknya prosedur tidak demikian. Perhatikan pendefinisian fungsi **polinomx** dan **polinomy** dalam skrip di In[1]:= dan In[4] := berikut ini.

In[1]:= polinomx[n_]:=Expand[
 Product[
 n. (x+1)^i,{i,1,n}]
]
]

Mendefinisikan sebuah fungsi
 polinomx dengan peubah bebas

In[2]:= polinomx[3] Memberikan nilai n = 3 pada polinomx
 Out[2]= 1+6x + 15x² + 20x³ + 15x⁴ + 6x⁵ + x⁶

In[3]:=Coefficient[polinomx[3],x^4] Menampilkan nilai koefisien
 untuk x⁴

Out[3]=15 dari fungsi polinomx[3]

In[4]:= polinomy[m_]:=Expand[
 Product[
 (x^2+i),{i,1,m}]
 (.....)

In [5]:= polinomy[4] (.....)

Out [5]= (.....)

In [6]:= Coefficient[polinomy[4],x^4] (.....)

Out [6]= (.....)

Command-command yang diberikan dalam skrip In][4] :=, In[5] :=, dan In[6]:= di atas dapat digabungkan menjadi sebuah prosedur. Prosedur yang dimaksud dalam Mathematica® dinyatakan dalam bentuk command **Module**[] dengan format skripnya adalah sebagai berikut

Namaprosedur[peubah0_] := **Module**[{peubah}, ekspresi_{peubah}]

Berikut adalah sebuah contoh penggunaan prosedur yang bersesuaian dengan skrip In][4] :=, In[5] :=, dan In[6]:= yang dimaksudkan di awal bagian ini.

In[7]:= coefpoly [n0_,i0_] :=
 Module[{n=n0,i=i0},
 (.....)]

```

t=Expand[Product[(x^2+i),{i,1,n}]];
Print[t]; Coefficient[t,x^i];
Out[7]= (.....)

In[8]:= coefpoly[4,4] (.....)
Out[8]= (.....)

```

Module[] dapat juga digunakan untuk memudahkan proses komputasi yang sering dijumpai dalam bentuk sebuah bentuk fungsi yang melibatkan fungsi didalamnya yang biasa dikenal dengan sebutan komposisi fungsi. Salah satu bentuk operasi aritmatika terhadap komposisi fungsi dapat dijumpai dalam proses iterasi diskrit (persamaan diskrit), misalnya barisan bilangan *Fibonacci*. Menangani proses iterasi umumnya dilakukan dengan cara membuat sebuah fungsi yang prosesnya diulang melalui fungsi itu sendiri (*self iteration function*) sesuai kebutuhan.

Dua contoh penggunaan **Module[]** disajikan berikut ini:

✓ **Prosedur Tukar Posisi (switchover)**

Penggunaan **Module[]** untuk pemrograman dinamik dengan sebuah *list* lokal.

```

In[1]:= switchover [S_,P_]:=Module[
  {T=Table[0,{Length[S]},p],
  Table[p=P[[i]];Print[T];
  T[[p]]=S[[i]},{i,Length[S]}];
  T]
  k={4,2,3,1}
  switchover [{"a","b","c","d"},k]
  Clear[a,b,c,d]

```

Out[1]={d,b,c,a}

Dapat dicatat bahwa *k* menyatakan bentuk permutasi sebarang. Untuk bentuk yang lain bisa dibuat skrip “*k* = RandomSample[Range[4]]” dengan tujuan membangkitkan bentuk permutasi acak lainnya.

✓ **Prosedur Barisan Bilangan Fibonacci**

Penggunaan **Module[]** untuk pemrograman dinamik dengan sebuah fungsi lokal.

```

In[2]:= Fibonacci [n_]:=
Module[{f},
f[1]=f[2]=1;
f[i_]:=f[i]=f[i-1]+f[i-2];
f[n]
]
Fibonacci[6]
Out[2]=8

```

Bagian ini akan diakhiri dengan sebuah soal penentuan bentuk persamaan regresi sederhana dengan dua buah himpunan data yang beranggotakan masing-masing n buah data. Teorinya adalah sebagai berikut.

Pandang sebuah fungsi $g(x)$ dalam bentuk:

$$g(x) = a + b x. \quad (*)$$

Model/persamaan linear (*) merupakan bentuk umum dari sebuah regresi sederhana. Konstanta a dapat ditentukan nilainya melalui rumus berikut:

$$a = \frac{1}{n} \sum y_i - \frac{1}{n} \sum x_i b \quad (**)$$

Sementara itu nilai koefisien b dapat ditentukan melalui rumus berikut:

$$b = \frac{n \sum x_i^2 y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2} \quad (***)$$

Dengan mensubsitusi nilai a dan nilai b yang diperoleh masing-masing dari persamaan (**) dan persamaan (***) maka fungsi $g(x)$ di (*) dapat ditentukan bentuknya. Pada latihan di bawah ini gunakan *command-command* yang sudah didapati pada materi sebelumnya untuk mengkonstruksi persamaan (*) sebagai bentuk persamaan regresi linear dari data yang diberikan.

Latihan

Diberikan pasangan data (x,y) sebanyak 10 buah pasangan berikut ini:

x	1	2	3	4	5	6	7	8	9	10
y	4	6	8	10	14	16	20	22	24	28

Tulis skrip *Mathematica*® untuk mendapatkan persamaan (*) melalui nilai a menggunakan rumus (**) dan nilai b menggunakan rumus (***)

Catatan: Gunakan *command* **Sum[]** dan **Total[]** dalam sebuah **Module[]** yang relatif mudah Anda selesaikan melalui skrip-skrip yang sudah diberikan sebelumnya.

MENJADI MAHIR-5

LIST DATA

Bekerja dengan matematika yang melibatkan himpunan (barisan) adalah hal yang biasa dilakukan bagi ilmuwan matematika dan *basic science*. Sekumpulan data yang memiliki hubungan fungsional dapat disajikan dalam bentuk himpunan/barisan. Menangani data dengan jumlah yang besar tidak ada pilihan lain selain menggunakan alat komputasi. Pada bagian ini akan diperlihatkan teknis menggunakan *Mathematica*® untuk menangani permasalahan operasi aritmatika dalam suatu program sederhana yang melibatkan pengulangan (*loop*) pada barisan bilangan (data). Setelah mengikuti kegiatan ini Anda akan mampu bekerja dengan *Mathematica*® berkenaan dengan penggunaan (*loop*) dalam bentuk *list* data melalui *command* **Table**[] dengan atau tanpa operasi-operasi simbolis.

Pada bagian akhir materi sebelumnya, Anda diberikan pasangan data sebanyak sepuluh data ($n = 10$) yang disajikan dalam bentuk *list* (himpunan) bilangan atau barisan bilangan. Untuk mendefinisikan data dalam sebuah *list* data, *Mathematica*® menyediakan *command* khusus yang diberi nama **Table**. Dengan *command* ini, Anda bisa mengerjakan atau melakukan operasi-operasi himpunan dan mengolah serta menyimpan data pengamatan. Beberapa variasi *command* **Table**[] yang terdapat dalam *Mathematica*® diberikan dalam Tabel 12.

Tabel 12. *Command Mathematica*[®] untuk mendefinisikan/ menyajikan data dalam bentuk *list* (himpunan).

Command	Keterangan
Table [$f, \{i, i_{\max}\}$]	Mengkonstruksi sebuah <i>list</i> dengan anggota bercirikan ekspresi f sebanyak i_{\max} buah anggota <i>list</i> .
Table [$f, \{i, i_{\max}\}$]	Mengkonstruksi sebuah <i>list</i> dengan anggota bercirikan ekspresi f sebanyak pergerakan i hingga i_{\max} .
Table [$f, \{i, i_{\min}, i_{\max}\}$]	Mengkonstruksi sebuah <i>list</i> dengan anggota bercirikan ekspresi f yang pergerakan i dimulai dari i_{\min} hingga i_{\max} .
Table [$f, \{i, i_{\min}, i_{\max}, di\}$]	Mengkonstruksi sebuah <i>list</i> dengan anggota bercirikan ekspresi f yang pergerakan i dimulai dari i_{\min} hingga i_{\max} dengan beda/selisih ukuran langkah (<i>step size</i>) sebesar di .
Table [$f, \{i, i_{\max}\}, \{j, j_{\max}\}$]	Mengkonstruksi sebuah <i>list</i> di dalam <i>list</i> dengan anggota bercirikan ekspresi f yang pergerakan i sebanyak i_{\max} dan pergerakan j sebanyak j_{\max} . Bentuk <i>list</i> ini juga dikenal sebagai bentuk matriks i_{\max} baris dan j_{\max} kolom.

Ilustrasi penggunaan *command* **Table**[] yang terdapat dalam Tabel 12 diberikan dalam sejumlah skrip berikut ini.

b. Mengkonstruksi sebuah *list* (himpunan) dengan anggota *list* merupakan nilai dari suatu fungsi kuadrat hingga suku ke enam.

In [1] := **Table**[$i^2, \{i, 6\}$]

Out [1] := {1, 4, 9, 16, 25, 36}

c. Mengkonstruksi sebuah *list* (himpunan) dengan anggota *list*

merupakan nilai dari suatu fungsi $\sin \frac{n}{5}$ untuk $n = 1, \dots, 4$.

In [2] := **Table**[**Sin**[$n/5$], { $n, 0, 4$ }] (.....)

Out [2] := {0, **Sin**[$\frac{1}{5}$], **Sin**[$\frac{2}{5}$], **Sin**[$\frac{3}{5}$], **Sin**[$\frac{4}{5}$]}

- d. Mengkosntruksi sebuah *list* (himpunan) dengan anggota *list* merupakan nilai numerik dari hasil terakhir sebelumnya.
 In [3] := N[%]
 Out [3] := (.....)
- e. Mengkosntruksi sebuah *list* (himpunan) dengan anggota *list* merupakan semua nilai fungsi $x^i + 2i$ untuk $i = 1, \dots, 5$.
 In [4] := Table[x^i + 2i, {i, 5}]
 Out [4] := (.....)
- f. Mengkosntruksi sebuah *list* (himpunan) dengan anggota *list* merupakan hasil evaluasi perkalian fungsi $x^i + 2i$ untuk $i = 1, \dots, 5$.
 In [5] := Product[x^i + 2i, {i, 5}]
 Out [5] := $(2 + x)(4 + x^2)(6 + x^3)(8 + x^4)(10 + x^5)$
- g. Mengkosntruksi sebuah *list* (himpunan) dengan anggota *list* merupakan semua nilai fungsi \sqrt{x} untuk $x = 0, 0,25, 0,5, 0,75, 1$.
 In [6] := Table[Sqrt[x], {x, 0, 1, 0.25}]
 Out [6] := (.....)
- h. Mengkosntruksi sebuah *list* (himpunan) dengan anggota *list* merupakan semua nilai fungsi $(\sqrt{x})^2 + 3$ untuk $x = 0, 0,25, 0,5, 0,75, 1$.
 In [7] := %^2 + 3 (.....)
 Out [7] := (.....)
- i. Mengkosntruksi sebuah *list* (himpunan) dengan anggota *list* disusun dalam bentuk tabel dengan isi tabel tersebut berupa semua nilai yang ada pada Out[7] di atas.
 In [8] := %//TableForm
 Out [8]// (.....)

j. Mengkonstruksi sebuah list (himpunan) dengan anggota list semua nilai fungsi $x^i + y^j$ untuk $i = 1, 2, 3$ dan $j = 1, 2$. Bentuk ini juga dikenal sebagai bentuk matriks berukuran 3 baris dan 2 kolom.

In [9] := Table[x^i + y^j, {i, 3}, {j, 2}] (.....)

Out [9] := (.....)

k. Mengkonstruksi sebuah list (himpunan) dengan anggota list berupa semua bilangan random antara 0 dan 1 untuk empat suku pertama.

In[10]:= Table[Random[],{4}]

Out [10] := {0.0560708,0.6303,0.359894,0.871377}

Setelah mengkonstruksi sebuah list/himpunan biasanya ditindaklanjuti dengan mengolah list tersebut melalui operasi-operasi aritmatika atau logika. *Mathematica*® menyediakan banyak *command* yang dapat digunakan untuk menangani operasi aritmatika atau logika terhadap satu atau lebih list/himpunan. Salah satu upaya untuk mengelola list/himpunan adalah mengambil beberapa anggota list. Oleh karena itu, bila diinginkan pengambilan bagian tertentu dalam suatu daftar atau tabel dapat dilakukan dengan menggunakan *command* sebagaimana berikut dalam Tabel 13.

Tabel 13. *Command Mathematica*® untuk mengambil sebagian anggota list/himpunan.

Command	Keterangan
$t[[i]]$ atau Part [t, i]	Mengambil/memunculkan suku/bagian ke- i dari ekspresi t . Amati perbedaan kedua <i>command</i> .
$t[[{i_1, i_2, \dots}]]$ atau Part [$t, \{i_1, i_2, \dots\}$]
$t[[i, j, \dots]]$ atau Part [t, i, j, \dots]

Mengekspresikan m dengan sebuah list bersarang/*Nestlist* (list dalam list) untuk suatu fungsi $i - j$ dengan $i = 1, 2$ dan $j = 1, 2$.

In [11]:= $m = \text{Table}[i-j, \{i, 2\}, \{j, 2\}]$ Mendefinisikan m merupakan list dalam Out [11]= $\{\{0, -1\}, \{1, 0\}\}$ list yang dapat dicaca sebagai sebuah matriks dengan baris- i dan kolom- j

In [12]:= $m[[1]]$ Memunculkan *term*/suku ke-1 dari m
atau Out [12]= $\{0, -1\}$ dapat diartikan sebagai entri baris ke 1 dari sebuah matriks m .

In [13]:= $\%[[2]]$ Memunculkan *term* ke-2 dari $m[[1]]$ atau entri matriks m_{12} (baris ke-1 kolom ke-2).

Out [13]= -1
In [14]:= $m[[1, 2]]$ Alternatif lain untuk memunculkan *term* ke-2 dari $m[[1]]$.
Out [14]= -1

In [15]:= $\text{TableForm}[m]$ Menampilkan m dalam bentuk serupa matriks.
Out [15]// $\text{TableForm} =$

0 -1
1 0

In[16]:= $\text{Table}[i j^2 k^3, \{i, 2\}, \{j, 2\}, \{k, 2\}]$ Mengevaluasi sebuah larik/array (matriks) berukuran $2 \times 2 \times 2$ yang dimulai dengan daftar paling "dalam" dengan fungsi berbentuk ij^2k^3 .

Out[16]= $\{\{\{1, 8\}, \{4, 32\}\}, \{\{2, 16\}, \{8, 64\}\}\}$

Keluaran Out[16] dapat diperiksa dengan kalkulasi sebagai berikut:

Untuk nilai-nilai $i=1, j=1$, dan $k=1$ diperoleh $ij^2k^3 = 1$. Dengan cara yang sama, untuk nilai $i=1, j=1$, dan $k=2$ diperoleh $ij^2k^3 = 8$. Demikian seterusnya sehingga untuk $i=1, j=2$, dan $k=1$ diperoleh $ij^2k^3 = 4$.

Untuk mendapatkan dalam bentuk format penulisan matriks input In[16]:= ditulis dengan skrip berikut ini:

```
In[17]:= %16//MatrixForm
```

```
Out[16] //MatrixForm=
```

$$\begin{pmatrix} \begin{pmatrix} 1 \\ 8 \end{pmatrix} & \begin{pmatrix} 4 \\ 32 \end{pmatrix} \\ \begin{pmatrix} 2 \\ 16 \end{pmatrix} & \begin{pmatrix} 8 \\ 64 \end{pmatrix} \end{pmatrix}$$

MENJADI MAHIR-6

MATRIKS (*LIST DALAM LIST*)

Bagian ini akan difokuskan pada topik matriks. Matriks dapat dipandang sebagai *list* dalam *list* sebagaimana yang telah dikemukakan dalam bagian sebelumnya. Matriks memainkan peranan penting dalam membuat kesederhanaan dalam menyelesaikan dan menganalisa sebuah sistem. Sebagaimana diketahui bahwa salah satu peranan matematika dalam kehidupan adalah menyelesaikan permasalahan-permasalahan kehidupan dunia nyata melalui model-model matematika yang dapat dideskripsikan dalam bentuk matriks. Setelah menyelesaikan bagian ini Anda akan terbiasa dengan persoalan yang melibatkan konsep matriks dan melakukan operasi-operasi matriks secara terampil dengan bantuan *Mathematica*[®]. Indikator keberhasilan Anda dalam bagian ini adalah kemampuan Anda bekerja dengan *Mathematica*[®] dalam menyelesaikan persoalan yang melibatkan sistem persamaan linear menggunakan konsep matriks (*list dalam list*).

A. Pendefinisian Matriks dengan *Mathematica*[®]

Pandang sebuah permasalahan sederhana yang realtif sering ditemukan dalam pembelajaran matematika di tingkat sekolah menengah. Misalnya, harga 5 pensil dan 2 buku adalah Rp 26000 sedangkan harga 2 pensil dan 4 buku Rp 38000. Jika harga pensil dinyatakan dengan x dan harga buku dinyatakan dengan y , maka dalam bentuk kalimat matematis dapat dinyatakan dengan $5x+2y = 26000$ dan $2x+4y = 38000$. Kalimat matematis ini dapat disusun dalam bentuk *list* berikut ini

$$\{5x + 2y = 26000, 2x + 4y = 38000\}.$$

Jika dibuat *list* baru di dalam *list* tersebut maka *list* tersebut menjadi

- i. untuk sisi sebelah kiri tanda sama dengan (*lhs/left hand side*) :
 $\{\{5,2\},\{2,4\}\}$ dan
- ii. untuk sisi sebelah kanan tanda sama dengan (*rhs/right hand side*):
 $\{26000,38000\}$.

List pada i dan ii dalam skrip *Mathematica*® untuk *lhs* dan *rhs* dapat dinyatakan sebagai berikut:

```
In[1]:= lhs={{5,2},{2,4}};
      rhs={26000,38000};
```

Jika digunakan *command* **MatrixForm**, sebagaimana skrip berikut ini,

```
In[2]:= lhs//MatrixForm
      rhs//MatrixForm
```

akan diperoleh penyajian *list* dalam bentuk matriks sebagai berikut:

```
Out[2]//MatrixForm=
```

$$\begin{pmatrix} 5 & 2 \\ 2 & 4 \end{pmatrix}$$

```
Out[3]//MatrixForm=
```

$$\begin{pmatrix} 26000 \\ 38000 \end{pmatrix}$$

Dari input yang diberikan dalam `In[1]:=` kemudian diikuti dengan `In[2]:=` yang disertai `Out[2]=` memperlihatkan bahwa *Mathematica*® dapat mendefinisikan *list* dalam *list* sebagai sebuah matriks. Pada prinsipnya *Mathematica*® menyediakan banyak cara menulis *list* untuk dikonversi kedalam matriks. Beberapa diantaranya diberikan berikut ini.

`In[1]:= m34=Table[2 i-j,{i,4},{j,3}]` Mendefinisikan m_{34} dengan sebuah *list* hasil kalkulasi $2i-j$, utk $i=1..4$ dan $j=1..3$.

```
Out[1]= {{1,0,-1},{3,2,1},{5,4,3},{7,6,5}}
```

`In[2]:= m34//MatrixForm` Mengkonversi m_{34} kedalam matriks.

```
Out[2]//MatrxForm=
```

$$\begin{pmatrix} 1 & 0 & -1 \\ 3 & 2 & 1 \\ 5 & 4 & 3 \\ 7 & 6 & 5 \end{pmatrix}$$

In[3]:= listfungsi=Array[f,{2,3},{0,4}]

Mendefinisikan sebuah fungsi dalam bentuk *list* dalam *list* yang terdiri dari 2 *list* untuk baris (angka 0 dan

Out[3]= $\left\{ \left\{ f[0,4], f[0,5], f[0,6] \right\}, \left\{ f[1,4], f[1,5], f[1,6] \right\} \right\}$

1) dan 3 *list* untuk kolom (angka 4,5,6) dengan entri-entri berupa fungsi $f[\text{baris},\text{kolom}]$. In[4]:= listfungsi//MatrixForm Mengkonversi *listfungsi* kedalam matriks.

Out[4]//MatrixForm=
$$\begin{pmatrix} f[0,4] & f[0,5] & f[0,6] \\ f[1,4] & f[1,5] & f[1,6] \end{pmatrix}$$

In[4]:= Array[0&,{3,3}] (.....)

Out[4]= (.....)

In[5]:= Boole[Array[Greater,{4,4}]] (.....)
//MatrixForm

Out[4]//MatrixForm=(.....)

In[6]:= m23=Array[Subscript[a, ##] (.....)
&,{2,3}]

Out[4]= (.....)

In[7]:= MatrixForm[m23] (.....)

Out[4]//MatrixForm=(.....)

B. Operasi Matriks dalam Mathematica®

Mathematica® menangani matriks numerik dan simbolis, secara otomatis mengakses sejumlah besar algoritma yang sangat efisien. *Mathematica*® menggunakan algoritma *State-of-The-Art* untuk bekerja dengan matriks padat dan jarang, dan

2. Operasi Fundamental Matriks

Terdapat sejumlah operasi yang berlaku pada matriks. Beberapa diantaranya dijelaskan dalam bagian berikut ini. Pandang sebuah matriks A berukuran $n \times n$ yang didefinisikan dengan

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = A = [a_{ij}], \quad i = 1, 2, \dots, n; \quad j = 1, 2, \dots, n$$

- Matriks *transpose* A , A^T , adalah sebuah matriks yang didefinisikan dengan

$$A^T = [a_{ji}]$$

Contoh:

Misal $A = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}$. Matriks $A^T = [a_{ji}]$, $i = 1, 2; j = 1, 2$ adalah

$$\begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}^T = \begin{bmatrix} -1 & 2 \\ 3 & 2 \end{bmatrix}$$

- Determinan matriks A adalah jumlah dari semua hasil kali elementer bertanda (hasil kali n buah unsur A tanpa ada pengambilan unsur dari baris/kolom yang sama). Kata “bertanda” dalam konteks ini adalah bertanda negatif jika invers permutasi (bilangan yang lebih besar mendahului bilangan yang lebih kecil dalam urutan permutasi) pada indeks kolom dari perkalian unsur bernilai ganjil dan bertanda positif jika invers permutasi pada indeks kolom dari perkalian unsur bernilai 0 atau genap.

Contoh:

Misal $A = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}$. Determinan matriks A ,

$$\det A = \det \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix} = (-1)(2) - (3)(2) = -8$$

Perhatikan bahwa tanda negatif pada perkalian elementer (3)(2) adalah negatif karena invers permutasi kolom dari perkalian unsur a_{12} a_{21} adalah ganjil. Ketentuan ini berlaku karena permutasi $\{2,1\}$ memiliki invers permutasi bernilai 1. Dengan kata lain terdapat 1 kejadian dimana bilangan yang lebih besar mendahului bilangan yang lebih kecil (2 mendahului 1).

- Minor matriks A , M_A , merupakan sebuah matriks dengan entri/elemen m_{ij} berupa nilai determinan yang diperoleh dengan cara menghilangkan baris- i dan kolom- j .

Contoh:

Misal $A = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}$. Matriks minor, $M_A = [m_{ij}]$, $i = 1, 2; j = 1, 2$,

adalah sebuah matriks dengan

$$m_{11} = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix} = 2, \quad m_{12} = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix} = 2, \quad m_{21} = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix} = 3, \text{ dan}$$

$$m_{22} = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix} = -1. \text{ Sehingga minor A adalah}$$

$$M_A = \begin{bmatrix} 2 & 2 \\ 3 & -1 \end{bmatrix}.$$

- Matriks *adjoint* A merupakan sebuah matriks yang didefinisikan dengan $adj_A = [\eta_{ij}]^T$, $i = 1, 2, \dots, n; j = 1, 2, \dots, n$

dimana η_{ij} merupakan matriks minor-kofaktor yang didefinisikan dengan $\eta_{ij} = (-1)^{i+j} |a_{(n-i+1)(n-j+1)}|$

Contoh:

Misal $A = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}$. Matriks *adjoint* A adalah

$$adj_A = [\eta_{ij}]^T, \quad i = 1, 2; \quad j = 1, 2$$

dimana

$$\begin{bmatrix} \eta_{11} = (-1)^{1+1} |2| = 2 & \eta_{12} = (-1)^{1+2} |2| = -2 \\ \eta_{21} = (-1)^{2+1} |3| = -3 & \eta_{22} = (-1)^{2+2} |-1| = -1 \end{bmatrix}^T = \begin{bmatrix} 2 & -2 \\ -3 & -1 \end{bmatrix}^T = \begin{bmatrix} 2 & -3 \\ -2 & -1 \end{bmatrix}$$

- Matriks invers A , A^{-1} , merupakan matriks yang didefinisikan dengan

$$A^{-1} = \frac{adj_A}{\det A}$$

Contoh:

Misal $A = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}$. Dengan menggunakan hasil-hasil sebelumnya,

matriks invers A dapat diperoleh yaitu

$$A^{-1} = \frac{adj_A}{\det A} = \frac{\begin{bmatrix} 2 & -3 \\ -2 & -1 \end{bmatrix}}{(-8)} = \begin{bmatrix} \frac{-1}{4} & \frac{3}{8} \\ \frac{1}{4} & \frac{1}{8} \end{bmatrix}.$$

- Trace matriks A adalah jumlah elemen/entri diagonal utama dari matriks A .

Contoh:

Misal $A = \begin{bmatrix} -1 & 3 \\ 2 & 2 \end{bmatrix}$. Trace matriks A dikalkulasi dengan cara

$$Tr(A) = \sum_{i=j=1}^2 m_{ij} = (-1) + 2 = 1$$

- Rank matriks A merupakan sebuah nilai skalar yang menunjukkan banyaknya baris/kolom yang bebas linear dalam sebuah matriks. Rank matriks dapat digunakan untuk mengetahui apakah suatu matriks itu singular atau nonsingular. Misalkan A matriks berukuran $n \times n$. Matriks A dikatakan nonsingular apabila $\text{rank}(A) = n$, sebaliknya matriks A adalah singular jika $\text{rank}(A) < n$. Untuk mendapatkan rank sebuah matriks diperlukan transformasi elementer dengan mengubah sebanyak mungkin baris atau kolom menjadi vektor nol. Hal ini dilakukan karena matriks yang ekuivalen mempunyai rank matriks yang sama. Perlu dicatat bahwa matriks yang terdiri dari 2 baris, jika baris yang satu kelipatan dari baris yang lainnya, maka rank matriks = 1.

Contoh:

Misal $A = \begin{bmatrix} 2 & 3 & 1 \\ 2 & 1 & 2 \\ 4 & 4 & 3 \end{bmatrix}$. Operasi aritmatika elementer pada baris

(OBE) akan digunakan untuk mendapatkan sebanyak mungkin baris yang memiliki semua elemennya nol, misalnya

$$\begin{bmatrix} 2 & 3 & 1 \\ 2 & 1 & 2 \\ 4 & 4 & 3 \end{bmatrix} \xrightarrow{\begin{matrix} (-1)\{2,3,1\} + \{2,1,2\} \\ (-2)\{2,3,1\} + \{4,4,3\} \end{matrix}} \begin{bmatrix} 2 & 3 & 1 \\ 0 & -2 & 1 \\ 0 & -2 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 2 & 3 & 1 \\ 0 & -2 & 1 \\ 0 & -2 & 1 \end{bmatrix} \xrightarrow{(-1)\{0,-2,1\} + \{0,-2,1\}} \begin{bmatrix} 2 & 3 & 1 \\ 0 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Hasil akhir OBE yang dilakukan membentuk sebuah matriks dengan 2 buah baris yang tidak semuanya bernilai 0. Oleh karena itu matriks A memiliki $\text{Rank}(A) = 2$ berdasarkan definisi yang diberikan.

Beberapa operasi aritmatika yang telah dikemukakan di atas juga dapat dilakukan dengan menggunakan *Mathematica*®. *Command-command* yang digunakan dalam skrip *Mathematica*® terkait dengan operasi yang telah dibahas sebelumnya diberikan dalam Tabel 14.

Tabel 14. *Command Mathematica*® untuk Operasi Matriks A (Standar)

Command	Keterangan
<i>Transpose</i> [A]	Mempertukarkan baris-kolom matriks A menjadi kolom-baris.
<i>Det</i> [A]	Menentukan nilai Determinan matriks A
<i>Minors</i> [A]	Menentukan bentuk minor matriks A
<i>Inverse</i> [A]	*Menentukan Balikan (<i>inverse</i>) Matriks A
<i>Tr</i> [A]	Menentukan <i>Trace</i> (jumlah entri diagonal utama) matriks A
<i>MatrixRank</i> [A]	Range (banyaknya baris atau kolom yang bebas linear) suatu matriks A

*Dengan asumsi matriks A berukuran $n \times n$ dan *nonsingular* (determinan $A \neq 0$)

Perhatikan beberapa contoh skrip *Mathematica*® berikut ini kemudian bandingkan hasilnya (output) dengan hasil yang dikalkulasi sebelumnya. Untuk ukuran matriks yang lebih besar dapat digunakan dengan mudah.

In[1]:= m1={{-1,3},{2,2}}; Mendefinisikan m_1 sebagai sebuah list dalam list (2 list dalam 1 list).
 In[2]:= m1//MatrixForm Konversi list kedalam bentuk matriks berukuran 2x2

$$\text{Out}[2]= \begin{pmatrix} -1 & 3 \\ 2 & 2 \end{pmatrix}$$

In[3]:= Transpose[m1]//MatrixForm (.....)
 Out[3]= (.....)

In[4]:= Det[m1] (.....)
 Out[4]= (.....)

In[5]:= Minors[m1]//MatrixForm (.....)
 Out[5]= (.....)

In[6]:= Inverse[m1]//MatrixForm (.....)
 Out[6]= (.....)

In[7]:= Tr[m1] (.....)
 Out[7]= (.....)

In[8]:= MatrixRank[m1] (.....)
 Out[8]= (.....)

In[9]:= m2={{2,3,1},{2,1,2},{4,4,3}}; Mendefinisikan m_1 sebagai sebuah list dalam list (3 list dalam 1 list)..
 In[10]:= m2//MatrixForm Konversi list kedalam bentuk matriks berukuran 3x3

$$\text{Out}[10]= \begin{pmatrix} 2 & 3 & 1 \\ 2 & 1 & 2 \\ 4 & 4 & 3 \end{pmatrix}$$

In[11]:= Transpose[m2]//MatrixForm (.....)
 Out[11]= (.....)

In[12]:= Det[m2] (.....)
 Out[12]= (.....)

In[13]:= Minors[m2]//MatrixForm (.....)
Out[13]= (.....)

In[14]:= Inverse[m2]//MatrixForm (.....)
Out[14]= (.....)

In[15]:= Tr[m2] (.....)
Out[15]= (.....)

In[16]:= MatrixRank[m2] (.....)
Out[16]= (.....)

C. Aplikasi Matriks (Sistem Persamaan Linear) dalam Mathematica®

Tinjau kembali permasalahan pembelian pensil-buku yang dikemukakan di awal bagian ini. Kalimat matematis berkenaan dengan kasus tersebut adalah

$$5x + 2y = 26000 \text{ dan } 2x + 4y = 38000$$

atau dalam bentuk *list* dinyatakan dalam bentuk

$$\{5x + 2y = 26000, 2x + 4y = 38000\}.$$

Dapat diperiksa bahwa jika nilai $x = 1750$ dan $y = 8625$ kalimat matematika pada *lhs* dan *rhs* bernilai benar. Itu artinya bahwa nilai $x = 1750$ dan $y = 8625$ merupakan solusi bagi permasalahan pensil-buku yang dihadapi. Dalam bagian berikut ini akan diperlihatkan penggunaan *command Mathematica®* dalam menyelesaikan sistem persamaan linear buku-pensil sebelumnya.

In[1]:= m3={{5,2},{2,4}}; Mendefinisikan m_3 sebagai sebuah $a=26000$; $b=38000$; matriks berukuran 2×2 .

In[2]:= m3.{x,y}=={a,b} Menggunakan *command* perkalian titik (*dot product*) untuk memeriksa kesesuaian masalah dengan format skrip Mathematica®.

Out[2]= {5x + 2y, 2x + 4y} == {26000, 38000}

In[3]:= Solve[%,{x,y}] Menggunakan *command Solve[]* untuk

mendapatkan solusi sistem persamaan linear secara langsung.

Out[3]= {{x->1750,y->8625}}

In[4]:= m3//MatrixForm Konversi list m_3 kedalam bentuk matriks

$$\text{Out[4]} = \begin{pmatrix} 5 & 2 \\ 2 & 4 \end{pmatrix}$$

In[5]:= LinearSolve[m3,{x,y}] Menggunakan *command* **LinearSolve**[] untuk mendapatkan solusi sistem persamaan linear.

Out[5]= {1750,8625}

In[6]:= Inverse[m3].{x,y} Menggunakan *command* **Inverse**[] dan perkalian titik *dot* terhadap sebuah matriks koefisien m_3 untuk mendapatkan solusi sistem persamaan linear.

Out[5]= {1750,8625}

Skrip yang telah diberikan sebelumnya dapat digunakan untuk peubah yang lebih tinggi. Silahkan Anda gunakan kembali untuk menyelesaikan sistem persamaan linear dalam latihan berikut ini.

Latihan

1. Tuliskan skrip *Mathematica*® untuk menyelesaikan sebuah sistem persamaan linear berikut menggunakan **Solve**[], **Inverse**[], dan **LinearSolve**[].

$$x - 2y + 3z = 1$$

$$x + y + z = -1$$

$$x + 2y - z = 2$$

2. Tuliskan skrip *Mathematica*® untuk menyelesaikan sebuah sistem persamaan linear berikut ini dengan nilai a , b , dan c bervariasi dari $[-2,2]$.

Catatan: Gunakan **Solve**[], **Inverse**[], dan **LinearSolve**[]

$$x - 2y + 3z = a$$

$$x + y + z = b$$

$$x + 2y - z = c$$

3. Tuliskan skrip *Mathematica*® untuk menyelesaikan sebuah sistem persamaan linear berikut ini dengan nilai a_1, a_2, a_3, a_4 , dan a_5 bervariasi dari $[-2,2]$.

Catatan: Gunakan **Solve[]**, **Inverse[]**, dan **LinearSolve[]**

$$\begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} \\ \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} & \frac{1}{9} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{pmatrix}$$

MENJADI MAHIR-7

KALKULUS DIFERENSIAL: SIMBOLIS

Sistem persamaan linear yang telah dibahas dalam bagian sebelumnya merupakan salah satu jenis sistem persamaan yang dikenal dalam matematika. Sistem ini dipelajari di tingkat awal masa perkuliahan di PT. Sistem persamaan jenis lain yang penting dipelajari dan dikuasai dengan baik adalah **sistem persamaan diferensial**. Sistem ini terdiri dari persamaan-persamaan yang melibatkan turunan fungsi. Jika turunan fungsi yang terlibat dalam sebuah persamaan dalam sistem tersebut adalah turunan biasa (turunan terhadap satu peubah bebas) persamaan tersebut dinamakan **persamaan diferensial biasa**. Sementara itu jika turunan fungsi yang terlibat adalah turunan parsial (turunan terhadap dua atau lebih peubah bebas) maka persamaan tersebut dinamakan **persamaan diferensial parsial**. Dalam bagian ini Anda akan diperkenalkan dengan sebuah persamaan diferensial biasa dan sistem persamaan diferensial biasa serta cara menyelesaikannya menggunakan *Mathematica*® melalui sebuah *command* **DSolve**[]. Indikator keberhasilan Anda dalam bagian ini diperoleh dari hasil latihan-latihan yang diberikan dengan tingkat ketercapaian jawaban soal latihan benar lebih dari 80%.

Pandang sebuah persamaan diferensial yang dinyatakan dalam bentuk:

$$y' = f(x, y) \tag{1}$$

dimana f adalah sebuah fungsi kontinu dari x dan y pada domain D dalam bidang- xy ; dan misalkan (x_0, y_0) adalah sebuah titik pada D .

Anda dapat menggunakan *command Mathematica*® **DSolve[]** untuk menemukan solusi simbolis dari persamaan diferensial biasa tertentu dari (α). Beberapa penjelasan berikut ini dapat Anda coba pada lembaran *notebook Mathematica*® Anda.

- **DSolve[]** melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa untuk mendapatkan fungsi solusi dalam bentuk $y[x]$

In[1]:= DSolve[y'[x]+y[x]==1,y[x],x]

Out[1]= {{y[x]→1+e^{-x} C[1]}}

- **DSolve[]** melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa untuk mendapatkan fungsi solusi dalam bentuk y

In[3]:= DSolve[y'[x]+y[x]==1,y,x]

Out[3]= {{y→Function[{x},1+e^{-x} C[1]]}}

Perbedaan dalam dua ilustrasi di atas terletak pada saat akan memverifikasi solusi melalui perintah *replace* “/.”. Perhatikan entri berikut.

- Simbol “%” dalam entri berikut adalah “ $y[x]→1+e^{-x} C[1]$ ”

In[2]:= y[x]+2y'[x]+y[0] /.%

Out[2]= {1+e^{-x} C[1]+y[0]+2y'[x]}

- Simbol “%” dalam entri berikut adalah “ $y→Function[{x},1+e^{-x} C[1]]$ ”

In[4]:= y[x]+2y'[x]+y[0] /.%

Out[4]= {2+C[1]-e^{-x} C[1]}

- **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan dua atau lebih persamaan diferensial biasa secara bersamaan (sistem persamaan diferensial biasa) untuk mendapatkan fungsi-fungsi solusi.

In[6]:= DSolve[{y[x] == -z'[x], z[x] == -y'[x]},{y,z},x]


```
Out[6]= {{z→Function[{x}, (1/2)e-x (1+e2x)C[1]- (1/2)e-x (-1+e2x)C[2]],
y→Function[{x},-(1/2) e-x (-1+e2x) C[1]+ (1/2) e-x (1+e2x)C[2]]}}
```

- **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa yang dilengkapi dengan nilai awal (Masalah Nilai Awal) untuk mendapatkan fungsi solusi.

```
In[8]:= DSolve[{y'[x] = a y[x], y[0] = 1}, y[x], x]
```

```
Out[8]= {{y[x] → eax}}
```

- **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa orde tinggi yang dilengkapi dengan nilai awal (Masalah Nilai Awal) untuk mendapatkan fungsi solusi.

```
In[10]:= DSolve[{y''''[x] = y[x], y[0] = y'[0] = 0}, y[x], x]
```

```
Out[10]= {{y[x] → e-x ( C[3]+ e2x C[3] - e2x C[4] - 2ex C[3] Cos[x] + ex C[4] Cos[x] + ex C[4] Sin[x] )}}
```

- **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa orde satu homogen untuk mendapatkan fungsi solusi.

```
In[11]:= DSolve[y'[x] - x y[x] = 0, y[x], x]
```

```
Out[11]= {{y[x] → e $\frac{x^2}{2}$  C[1]}}
```

- **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa orde satu non homogen untuk mendapatkan fungsi solusi.

```
In[12]:= DSolve[y'[x] - x y[x] = x, y[x], x]
```

```
Out[12]= {{y[x] → -1+ e $\frac{x^2}{2}$  C[1]}}
```

- **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa Bernoulli untuk mendapatkan fungsi solusi.

In[23]:= DSolve[y'[x] - x y[x]^3 + y[x] = 0, y[x], x]

Out[23]= {{y[x] → $-\frac{\sqrt{2}}{\sqrt{1+2x+2e^{2x}C[1]}}$ }, {y[x] → $\frac{\sqrt{2}}{\sqrt{1+2x+2e^{2x}C[1]}}$ }}

Mathematica[®], melalui **D** dan **DSolve[]**, dapat digunakan dalam menyelesaikan persamaan diferensial parsial yang melibatkan dua atau lebih peubah.

- Penggunaan operator **D** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial parsial orde satu homogen untuk mendapatkan fungsi solusi.

In[30]:= (D[# ,x1] + D[# ,x2] + D[# ,x3]) & [y[x1,x2,x3]] = 0

Out[30]= $y^{(0,0,1)}[x1,x2,x3] + y^{(0,1,0)}[x1,x2,x3] + y^{(0,0,1)}[x1,x2,x3]$
= 0

- Penggunaan operator **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial parsial orde satu homogen untuk mendapatkan fungsi solusi.

In[30]:= (D[# ,x1] + D[# ,x2] + D[# ,x3]) & [y[x1,x2,x3]] = 0

Out[30]= $y^{(0,0,1)}[x1,x2,x3] + y^{(0,1,0)}[x1,x2,x3] + y^{(0,0,1)}[x1,x2,x3]$
= 0

- Penggunaan operator **DSolve[]** dapat melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial parsial orde satu non homogen untuk mendapatkan fungsi solusi.

In[35]:= DSolve[D[y[x1, x2],x1]+D[y[x1, x2],x2] = Exp[y[x1, x2]],
y[x1,x2],{x1,x2}]

Out[35]= {{y[x1, x2] → $-\text{Log}[-x_1 - C[1][-x_1 + x_2]]$ }}

Terhadap beberapa skrip yang telah diberikan sebelumnya Anda dapat menjadikannya sebagai pedoman untuk mengerjakan soal latihan yang akan diberikan berikut ini.

Latihan

- Gunakan *Mathematica*[®] untuk menyelesaikan persamaan diferensial berikut.
 - $\text{DSolve}[y'[x]+y[x]= a \text{Sin}[x],y[x],x]$
 - $\text{DSolve}[\{y'[x]+y[x]= a \text{Sin}[x],y[0]= 0\},y[x],x]$
 - $\text{DSolve}[\{y'[x]+y[x]= a \text{Sin}[x],y[0]= 0\},y,x]$
 - $\text{FullSimplify}[y''[x]+y[x]^2/.\%]$
- Gunakan *Mathematica*[®] untuk menyelesaikan persamaan diferensial berikut.

Terhadap output yang dikeluarkan oleh *Mathematica* untuk masing-masing soal no. 1 di atas, berikan keterangan/penjelasan tentang input yang diberikan dan output yang diperoleh.
- Gunakan *Mathematica*[®] untuk menyelesaikan persamaan diferensial berikut.
 - $\text{DSolve}[\{y'[x] = a y[x],y[0] = 1\},y[x],x]$
 - $\text{DSolve}[\{y''[x]-x y[x] = 0,y[0] = 1, y[9] = 1\},y,x]$
 - $\text{DSolve}[y''[x]+4y'[x]+5y[x] = 0,y[x],x]$
 - $\text{DSolve}[x^2y''[x] +4x y'[x]+7y[x] = 0,y[x],x]$
 - $\text{DSolve}[x y''[x]+2 y'[x]-x y[x] = \text{Sin}[x],y,x]$
 - $\text{DSolve}[y'''[x]+4y'[x] = 5y[x],y,x]$
- Terhadap output yang dikeluarkan oleh *Mathematica* untuk masing-masing soal no. 3 di atas, berikan keterangan/penjelasan tentang input yang diberikan dan output yang diperoleh.
- Latihan berikut ini akan memotivasi Anda untuk mengetahui bagaimana *Mathematica*[®] dapat membantu Anda dalam mendapatkan solusi dari persamaan diferensial yang digunakan dalam beberapa aplikasi. Tuliskan entri yang berada disebelah kanan notasi “In[...] := “ lalu eksekusi dengan menekan Shift-Enter. Amati hasilnya disebelah kanan notasi “Out[...] := “. Jika Outputnya benar, cobalah menggantikan entri yang berada disebelah kanan notasi “In[...] := “ dengan soal-soal yang Anda buat sendiri!

- a. Penyelesaian sebuah persamaan logistik (Persamaan Riccati):

```
In[1]:= DSolve[{y'[x] == y[x](1-y[x]/27),y[0]==a},y,x]/Quiet
Out[1]= {{y→Function[{x},(27 a E^x)/(27-a+a E^x)]}}
```

- b. Melakukan *Plotting* (menggambarkan kurva/grafik) fungsi solusi untuk sejumlah nilai awal yang berbeda-beda. In[2]:= Plot[Evaluate[y[x]/.%/.{a→1/13},{a→1/2},{a→4}],{x,0,18}]

- c. Penyelesaian sebuah persamaan pendulum linear dengan redaman dan memplot fungsi solusinya dapat dikerjakan dengan *Mathematica*® melalui input berikut:

```
In[1]:= DSolve[{y''[x]+y[x]==0,y[0]==1,y'[0]==1/3},y ,x]
Out[1]= {{y→Function[{x},1/3 (3 Cos[x]+Sin[x])]}}
In[2]:= Plot[y[x]/.%, {x,0,17}]
```

- d. Menyelesaikan sebuah persamaan pendulum linear dengan redaman dan memplot fungsi solusi:

```
In[1]:= DSolve[{y''[x]+3y'[x]+40 y[x]==0,y[0]==1,y'[0]==1/3},y,x]
Out[1]= {{y→Function[{x},1/453 E^(-3 x/2) (453 Cos[(Sqrt[151] x)/2]
+11 Sqrt[151] Sin[(Sqrt[151] x)/2])]}}
In[2]:= Plot[y[x]/.%, {x, 0,4}, PlotRange → All]
```

- e. Menyelesaikan sebuah sistem persamaan diferensial dan memplot potret fase:

```
In[1]:= DSolve[{x'[t]==-2y[t]+3x[t],y'[t]==15x[t]-y[t],x[0]==
a,y[0]==b},{x,y}, t];
In[2]:= ParametricPlot[Table[{x[t], y[t]}/.%,{a→1/(13+m), b→
1/(15+m)}, {m,0,20,7}], {t,-2,2}, Evaluated→True]
```

Menyelesaikan sebuah persamaan diferensial biasa koefisien peubah (lihat entri pada baris In[1]:=) untuk kemudian menemukan solusi dalam bentuk deret pangkat ketika solusi eksaknya diketahui (In[2]:=):

```
In[1]:= DSolve[{y'[x]+Exp[x]y[x]==1,y[0]==3},y ,x]
Out[1]= {{y→Function[{x},E^-E^x (3 E-ExpIntegralEi[1]+
ExpIntegralEi[E^x])]}}
In[2]:= Series[y[x]/.%[[1]],{x,0,7}]
Out[2]= 3-2 x-x^2/2+x^3/3+x^4/6-x^5/120-(11 x^6)/
360-x^7/105+O [x]^8
```

MENJADI MAHIR-8

KALKULUS DIFERENSIAL: NUMERIS

Command *Mathematica*® **`DSolve`**[] yang telah dikemukakan pada bagian sebelumnya memiliki versi numeriknya yaitu **`NDSolve`**[]. Dengan menggunakan *command* ini solusi persamaan diferensial atau sistem persamaan diferensial biasa yang ingin didapat akan diperoleh dalam bentuk nilai-nilai numerik. Setelah menyelesaikan bagian ini Anda akan terampil dalam mengoperasikan dan menggunakan perintah **`NDSolve`**[] pada program *Mathematica*® sebagai alat bantu dalam membuat program-program sederhana untuk menyelesaikan persamaan atau sistem persamaan diferensial secara numerik.

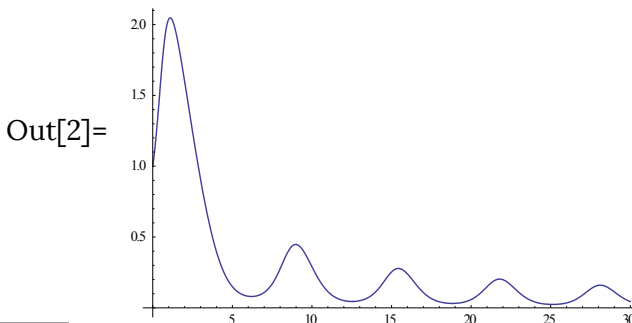
Sebagai catatan, materi latihan ini bersumber dari *Mathematica*® *Tutorial* pada *software Mathematica*® v.9.

- **`NDSolve`** melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa untuk mendapatkan nilai-nilai fungsi solusi numerik dalam bentuk barisan bilangan.

```
In[1]:= solusi=NDSolve[{y'[x]==y[x]Sin[x+y[x]],y[1]=1},y,{x,0,30}]
```

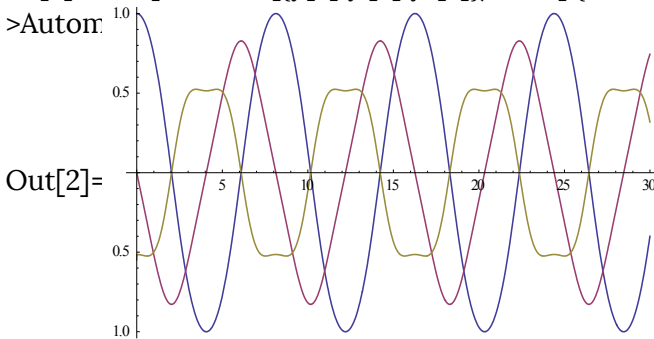
```
Out[1]= {{y->InterpolatingFunction[{{0.,30.}},<>]}}
```

```
In[2]:= Plot[Evaluate[y[x]/.solusi],{x,0,30},PlotRange->All]
```



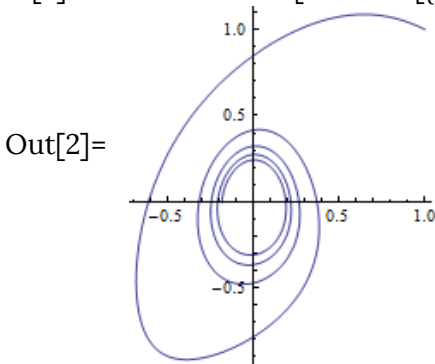
- **NDSolve** melaksanakan perintah untuk menyelesaikan sebuah persamaan diferensial biasa orde dua nonlinear untuk mendapatkan nilai-nilai fungsi solusinya

```
In[1]:=
sol2=NDSolve[{{y''[x]+Sin[Cos[y[x]]]y[x]==0,y[0]==1,y'[0]==0},
  y,{x,0,30}]
Out[1]= {{y->InterpolatingFunction[{{0.,30.}},<>]}}
In[2]:=Plot[Evaluate[{{y[x],y'[x],y''[x]}/.sol2},{x,0,30}],PlotStyle->Autom
```



- Solusi numerik untuk sistem persamaan diferensial biasa.

```
In[1]:= s=NDSolve[{{x'[t]==-y[t]-x[t]^2,y'[t]==2x[t]-y[t]^3,
  x[0]==y[0]==1},{x,y},{t,0,20}]
Out[1]={{x->InterpolatingFunction[{{0.,20.}},<>],
  y->InterpolatingFunction[{{0.,20.}},<>]}}
In[2]:= ParametricPlot[Evaluate[{{x[t],y[t]}/.s},{t,0,20}]
```

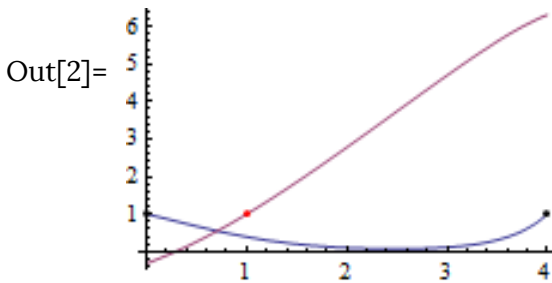


- Masalah nilai Batas dari sebuah pemetaan Non Linear

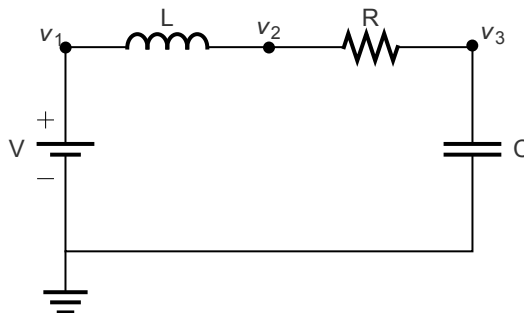
```

In[1]:= sol = NDSolve[{x'[t]==y[t], y'[t]==2-x[t], x[0]==x[4]==1,
    y[1]==1}, {x, y}, t]
Out[1]= {{x->InterpolatingFunction[{{0.,4.}},<>],
    y->InterpolatingFunction[{{0.,4.}},<>]}}
In[2]:= Plot[Evaluate[{x[t], y[t]} /. First[sol]], {t,0,4},
    Epilog->{Point[{0,1}], Point[{4,1}], Red, Point[{1,1}]}]

```



- Lakukan simulasi respon sebuah sirkuit RLC pada sebuah langkah voltase v_1 pada waktu $t < 0.001$. Penggunaan hukum-hukum komponen dan hukum Kirchoff untuk koneksinya:



Gambar 8. Diagram sebuah sirkuit RLC berdasarkan hukum Kirchoff untuk koneksinya.

Dengan *Mathematica*[®] dapat dilakukan dengan cara berikut.

Definisikan komponen dan hubungan, yaitu

```

In[1]:= components={r iR[t]==vR[t], l iL'[t]==vL[t], iC[t]==c vC'[t]};
connections={vR[t]+vL[t]+vC[t]-v1[t]==0, iL[t]==iR[t],
iR[t]==iC[t]};

```

Simulasi untuk sebuah langkah respon:

a. Kondisi awal (nilai awal dan nilai parameter) :

```
In[2]:=
ic={iR[0]==0,iL[0]==0,iC[0]==0,vR[0]==0,vL[0]==0,vC[0]==0};
params={r->5,l->10^-2,c->10^-4};
```

b. Ukuran langkah (iterasi):

```
In[3]:= v1[t_]:=UnitStep[t-.01];
```

c. Solusi Numerik:

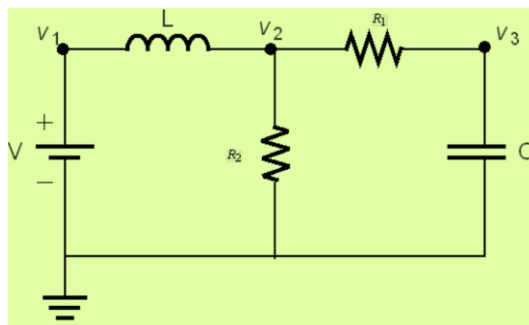
```
In[4]:=
sol=NDSolve[{components,connections,ic}/.params,vC,{t,0,.03}];
```

d. Grafik solusi:

```
In[5]:= Plot[Evaluate[{v1[t],vC[t]}/.sol],{t,0,.03},Filling->{1->0},
Ticks->{Range[0.01,0.03,0.01],Automatic},PlotRange->{0,1.5}]
```



- Lakukan simulasi respon sebuah sirkuit RLC pada sebuah langkah voltase v_1 pada waktu $t < 0.001$ sebagaimana latihan sebelumnya. Perhatikan Gambar 9.



Gambar 9. Diagram sebuah sirkuit RLC berdasarkan hukum Kirchoff untuk koneksinya.

Model matematis berkenaan dengan hubungan komponen dan koneksi sirkuit dalam Gambar 9 dapat dijelaskan berikudini:

Komponen :

$$\begin{cases} r_1 i R_1(t) = v R_1(t) \\ l i L'(t) = v L(t) \\ i C(t) = c v C'(t) \\ r_2 i R_2(t) = v R_2(t) \end{cases}$$

Koneksi:

$$\begin{cases} v L(t) + v R_2(t) = v_1(t) \\ v R_1(t) + v C(t) = v R_2(t) \\ i L(t) = i R_1(t) + i R_2(t) \\ i R_1(t) = i C(t) \end{cases}$$

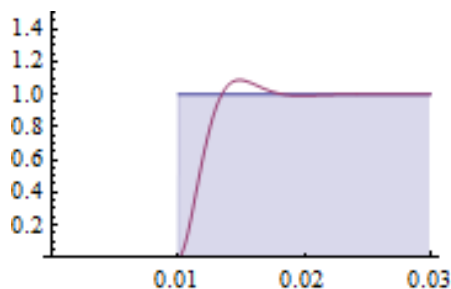
Nilai Awal:

$$\begin{cases} i R_1(0) = 0, i R_2(0) = 0, \\ i L(0) = 0, i C(0) = 0, \\ v R_1(0) = 0, v R_2(0) = 0, \\ v L(0) = 0, v C(0) = 0. \end{cases}$$

Nilai Parameter:

$$\begin{cases} r_1 \rightarrow 5, \\ r_2 \rightarrow 10, \\ l \rightarrow 10^{-2}, \\ c \rightarrow 10^{-4}. \end{cases}$$

Solusi numerik model sirkuit ini, secara grafik seperti gambar berikut.



Gambar 10. Diagram solusi model matematika yang dikonstruksi berdasarkan diagram dalam Gambar 9.

Dengan menggunakan *Mathematica*[®], tunjukan!

DAFTAR PUSTAKA

- Abell M.L & Braselton, J.P, (1993). *Differential equations with Mathematica*, ISBN. 0120415399.
- Kinzel W., Reents, G., Clajus, M. & Clajus, B. F. (1997). *Physics by Computer: Programming Physical Problems Using Mathematica and C*, Springer, ISBN: 354062743X,9783540627432.
- Kulenovic, M.R.S., & Merino, O. (2002). *Discrete Dynamical Systems and Difference Equations with Mathematica*, Chapman and Hall/CRC, Boca Raton, Florida. USA, ISBN: 1584882875.
- Shifrin L., (2008). *Mathematica programming: an advanced Introduction, Part I: the core language*. Wolfram Media, Inc., 2008.
- Subian S., Zakaria L., Ruby T., Mariska Y., & Muharramah U. (2020). Modifikasi Kurva Lorenz dan Komputasinya dengan Menggunakan Mathematica®, *Desimal: Jurnal Matematika* Vol 3 No 2 pp 99-108
- Stephen W., (2003). *The Mathematica Book, Fifth Edition*, Publisher: Wolfram Media, Inc., ISBN: 1579550223.
- Zakaria L., & Tuwankotta, J.M. (2016). Dynamics and Bifurcations in a Two-Dimensional Maps Derived From a Generalized $\Delta\Delta$ -sine-Gordon Equation, *Far East Journal of Dynamical Systems*, 28(3), pp 165-194.
- <http://reference.wolfram.com/Mathematica/ref/menuitem/WolframWebsite.html>.

INDEKS

A

Absolut, 19
ArcCos[x], 74
aritmatika, 1, 4, 14, 15, 25, 50, 55, 56,
58, 63, 66, 69, 79
Aritmatika, 4, 55
Array, 74

B

Bessel, 59
bil.eksak, 19

C

clear, 11, 12, 60, 61, 63
Clear, 11, 12, 60, 61, 63
coefficient, 33, 34, 62
Coefficient, 33, 34, 62
command, 1, 5, 7, 9, 11, 12, 14, 15, 16,
17, 18, 19, 21, 22, 23, 25, 28, 29, 30,
33, 34, 41, 42, 44, 45, 46, 47, 48,
49, 50, 52, 53, 54, 55, 57, 58, 59,
60, 62, 64, 65, 66, 67, 69, 73, 79,
81, 82, 84, 85, 90
Cos[x], 17, 42, 43, 44, 86, 89
curl, 59
Curl, 59

D

data, 1, 25, 38, 39, 40, 41, 46, 47, 48,
57, 64, 65, 66, 67
Degree, 21
Denominator, 33
Det, 79, 80, 81

diferensial, 15, 16, 18, 57, 58, 84, 85,
86, 87, 88, 89, 90, 91
Differential equations, 95
Div, 59, 79, 80, 81
Divergence, 95
Do, 28, 30, 35, 36, 37
DSolve, 15, 18, 84, 85, 86, 87, 88, 89,
90

E

eksak, 4, 5, 19
ekspresi, 2, 8, 18, 28, 29, 30, 31, 32,
33, 34, 55, 67, 69
Enter, 1, 2, 13, 14, 36, 88
Equations, 95
EulerGamma, 42
evaluate, 89, 90, 91, 92, 93
exp, 6, 11, 36, 55, 87, 89
Exp[x], 36, 55, 89
Expand, 5, 33, 52, 53, 60, 61, 62
exponent, 34Exponent, 34
expr, 31, 34, 57, 58
Expressions, 42

F

f[x], 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13,
15, 16, 17, 18, 19, 20, 21, 22, 24, 25,
26, 27, 28, 30, 31, 32, 33, 34, 36,
41, 42, 44, 46, 49, 50, 52, 53, 54,
55, 56, 57, 58, 59, 60, 62, 65, 66,
67, 69, 72, 73, 75, 79, 81, 82, 83, 84,
85, 87, 88, 89, 90, 92, 94, 95
f[x_], 36
factor, 5, 32, 33
faktorial, 19, 20
Filling, 43, 93
flatten, 25, 26, 27

For, 28, 29, 35, 36, 37
form, 34
Fraction, 36, 55, 89
FullSimplify, 88
function, 63, 85, 86, 89
Fundamental, 76
Fundamental matrix, 43, 93
fungsi, 1, 2, 3, 11, 15, 16, 17, 18, 19, 20,
21, 25, 34, 37, 38, 41, 42, 43, 44, 45,
46, 48, 49, 52, 53, 55, 59, 60, 61,
62, 63, 64, 66, 67, 68, 69, 70, 74,
84, 85, 86, 87, 89, 90, 91
fungsi komposisi, 60
fungsi kuadrat, 60, 67

G

GoldenRatio, 25, 26, 27
Grad, 59
Gradient, 59
grafik, 1, 3, 15, 19, 25, 38, 40, 41, 42,
43, 44, 46, 89, 93, 94

H

help, 10, 13, 44
hump, 61

I

Infinity, 21
Information, 21
Input, 6, 21
integral, 15, 17, 18, 37
Integrate, 15, 17
InterpolatingFunction, **90, 91, 92**
Intersection, 25, 26
invers, 57, 58, 76, 77, 78
Inverse, 79, 80, 81, 82, 83
InverseLaplaceTransform, 57

J

Join, 25, 26
jumlah, 27, 30, 66, 76, 78, 79

K

kalkulasi, 4, 5, 7, 16, 52, 70, 73
kalkulasi eksak, 4

kalkulasi eksak, 5
Kalkulasi, 4, 5, 7, 52
Kalkulasi Eksak, 4
koefisien, 34, 62, 64, 82, 89
Konstanta, 11, 21, 64
konvergen, 52

L

label, 6, 46
LaplaceTransform, 57
laplacian, 59
length, 27, 63
LinearSolve, 82, 83
list, 25, 26, 27, 32, 44, 63, 66, 67, 68,
69, 70, 72, 73, 74, 80, 81, 82
log, 6, 11, 20, 21, 22, 36, 45, 56, 87
Loop, 28, 66
Loops, 28

M

main button, 1
Mathematica, 1, 2, 3, 4, 5, 6, 7, 8, 10,
11, 12, 13, 15, 16, 17, 18, 19, 20, 21,
22, 24, 25, 26, 27, 28, 30, 31, 32,
33, 34, 36, 39, 41, 42, 44, 46, 49,
50, 52, 53, 54, 55, 56, 57, 58, 59,
60, 62, 65, 66, 67, 69, 72, 73, 75,
79, 81, 82, 83, 84, 85, 87, 88, 89,
90, 92, 94, 95
MatrixForm, 27, 71, 73, 74, 75, 80, 81,
82
Mean, 60
Mod, 10
Module, 9, 62, 63, 64, 65

N

NDSolve, 90, 91, 92, 93
Normal, 55
notasi, 4, 19, 20, 21, 51, 53, 60, 61, 88
numerical button, 1
numerik, 8, 20, 21, 49, 50, 51, 52, 68,
75, 90, 91, 93, 94
numeris, 33, 60

O

operator, 4, 8, 12, 14, 25, 26, 49, 75, 87

P

Palettes, 54
ParametricPlot, 45, 89, 91
Part, 34, 95
Partition, 27
pemrograman, 63, 64
Penggantian, 12
pengulangan, 25, 28, 29, 30, 36, 47, 66
Pengulangan, 29
perkalian, 12, 30, 31, 32, 49, 51, 52, 53, 68, 75, 76, 77, 81, 82
Perkalian, 30, 51, 75
permutasi, 63, 76, 77
Pernyataan, 29
persamaan, 15, 18, 19, 37, 45, 46, 57, 58, 63, 64, 65, 72, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91
plot, 1, 3, 36, 41, 42, 43, 44, 45, 89, 90, 91, 92, 93
Plotrange, 89, 90, 93
PlotStyle, 91
Point, 92
PolarPlot, 46
product, 30, 31, 49, 51, 52, 53, 54, 61, 62, 75, 81
prompt, 6

R

Random, 69
RandomInteger, 75
Range, 63, 79, 93
Rank, 78, 79
ReplaceAll, 13
running, 2, 37

S

semicolon, 2, 9

Series, 55, 56, 89
Setcoordinates, 59
simbol, 5, 7, 8, 16, 19, 21, 22, 23, 24, 30, 34, 49, 50, 54, 56, 58, 60, 85
Simplify, 5
Sin, 1, 11, 16, 17, 19, 21, 22, 26, 27, 42, 43, 44, 45, 46, 86, 88, 89, 90, 91
Sin[x], 1, 11, 16, 42, 43, 44, 86, 88, 89
sistem metrik, 49
solusi eksak, 89
solve, 33, 82, 83
Spherical, 59
sqrt, 6, 11, 20, 44, 56, 89
statement, 2
Sum, 30, 31, 49, 51, 52, 53, 54, 55, 65

T

Table, 63, 66, 67, 69, 70, 73, 89
TableForm, 70
Tan, 2
Tan[x], 2
tanda persen, 7
Tanh[x], 45
test, 29
together, 34
Transformasi Fourier, 56, 57, 58
transformasi laplace, 56, 57, 58, 59
Transpose, 79, 80

U

Umum, 30
Union, 25, 26
upper case, 1, 5

V

value, 13, 23, 24
VectorAnalysis, 59

W

While, 9, 29, 30, 35