

Modul Panduan Sistem Terdistribusi  
Yang Tidak Dipublikasikan

**Penggunaan**  
**GNU Octave untuk Representasi**  
**Data Simulasi Distributed System**

**Berkah I. Santoso, ST, MTI**



UNIVERSITAS  
**BAKRIE**

Program Studi Informatika  
Fakultas Teknik dan Ilmu Komputer  
Universitas Bakrie  
Jakarta

**2024**

## LEMBAR PENGESAHAN MODUL PANDUAN YANG TIDAK DIPUBLIKASIKAN

1. Judul : Modul Panduan Sistem Terdistribusi :  
Penggunaan GNU Octave untuk Representasi  
Data Simulasi Distributed Systems
2. Penulis
  - a. Nama Lengkap : Berkah I. Santoso, ST, MTI
  - b. Jenis Kelamin : Laki Laki
  - c. Pangkat/Golongan/NIDN : Asisten Ahli / III b / 0309068003
  - d. Bidang Keahlian : Distributed System, CloudComputing, Networking
  - e. Program Studi : Informatika
3. E-mail : berkah.santoso@bakrie.ac.id
4. Jangka waktu penulisan : 1 Agustus 2024 – 10 Agustus 2024

Jakarta, 11 Agustus 2024

Menyetujui,

**Ketua Lembaga Pengabdian kepada  
Masyarakat**

**(Prof. Ardiansyah, S.TP., .Si., Ph.D.)**  
NIDN: 0318107501

**Penulis**



**( Berkah I. Santoso, ST, MTI. )**  
NIDN: 0309068003

# Abstrak

Representasi simulasi data sebagai penunjang suatu materi perkuliahan, dalam hal ini *Distributed Systems* merupakan salah satu cara untuk pemenuhan syarat pembelajaran dalam rangka percepatan penerimaan topik materi yang dirasakan sangat efektif bagi mahasiswa. Mahasiswa sebagai peserta pembelajaran sudah dapat dipastikan memerlukan cara alternatif yang menarik untuk dapat menyerap topik materi pada mata kuliah *Distributed System*. Beberapa aplikasi untuk kebutuhan representasi simulasi data baik yang berbentuk *desktop application* seperti Mathematica<sup>®</sup>, Matlab<sup>®</sup>, Maple<sup>®</sup>, GNU Octave, GNUPlot, Sage maupun yang bersifat Software-as-a-Service (SaaS) seperti Sage in cloud, Wolfram Mathematica<sup>®</sup>, Simul8<sup>®</sup> merupakan aplikasi simulasi yang bersifat umum digunakan dalam rangka memberikan hasil simulasi sesuai kebutuhan akademik, organisasi nirlaba atau perusahaan yang memiliki lisensi *proprietary* berbayar dan berlisensi GNU General Public License (GPL) tidak berbayar [1].

Penggunaan aplikasi untuk kebutuhan akademik dapat memanfaatkan GNU Octave yang berlisensi GPL dan memiliki fitur yang lengkap untuk kebutuhan akademik sebagai *breakthrough* terkait ketersediaan aplikasi untuk representasi simulasi data yang bersifat modular, adaptif dan fungsional. Saat ini GNU Octave memiliki sifat *multiplatform* untuk mengurangi keterbatasan pada penggunaan multi *Operating Systems* yang bersifat heterogen. Adapun beberapa tantangan pengguna GNU Octave adalah pada mekanisme pemanfaatan kekayaan fitur GNU Octave untuk menghasilkan representasi simulasi data, mulai dari penggunaan variabel hingga topik terkait simulasi *Distributed Systems*.

Penulis mencoba untuk memberikan panduan penggunaan GNU Octave yang diharapkan dapat bermanfaat bagi para mahasiswa sebagai pengguna aplikasi modern yang membutuhkan aplikasi yang tidak berbayar untuk kebutuhan representasi simulasi data pada materi *Distributed Systems*. GNU Octave dapat menangani berbagai macam kebutuhan representasi simulasi data secara efisien dan mampu melakukan pengelolaan beragam tampilan hasil, walaupun perlu dilakukan pengaturan lebih lanjut terkait pendekatan *scripting* pada *Command Line Interface* (CLI) GNU Octave yang memiliki beban komputasi relatif rendah apabila dibandingkan dengan aplikasi simulasi data lainnya.

# Daftar Isi

<b>1</b>	<b>Pendahuluan</b>	<b>3</b>
1.1	Overview . . . . .	3
1.2	Mengapa GNU Octave ? . . . . .	3
1.3	Start, Quit, Getting Help . . . . .	4
<b>2</b>	<b>Mulai Menggunakan GNU Octave</b>	<b>7</b>
2.1	Variabel, Konstanta dan Tipe Data . . . . .	7
2.2	Ekspresi dan Operator . . . . .	9
2.3	Vektor dan Matriks . . . . .	11
2.4	Plotting . . . . .	15
2.5	Algoritma dan <i>Script</i> . . . . .	16
<b>3</b>	<b>Representasi Data</b>	<b>19</b>
3.1	Simulasi <i>Distributed System</i> . . . . .	19

# Bab 1

## Pendahuluan

### 1.1 Overview

Penggunaan aplikasi GNU is Not UNIX (GNU) Octave pada awalnya dikembangkan sebagai aplikasi penyerta dari materi perkuliahan desain reaktor kimia sehingga mendasari Dr. J.W. Eaton sebagai pemimpin perancang GNU Octave untuk mendesain ulang aplikasi agar dapat memaksimalkan kompleksitas aplikasi dan peningkatan penggunaan aplikasi yang dapat digunakan pada sektor industri maupun sektor akademik. GNU Octave secara khusus didesain untuk memecahkan kebutuhan simulasi komputasi matriks berupa pemecahan masalah persamaan simultan, perhitungan vektor Eigen dan nilai Eigen serta kebutuhan simulasi komputasi lainnya.

Pada permasalahan teknis secara nyata, suatu data dapat direpresentasikan sebagai matriks dan vektor, serta dapat dipandang sebagai kalkulator grafis yang sangat baik, dapat diprogram dan sangat berdaya komputasi. GNU Octave dirancang untuk dapat menyelesaikan permasalahan komputasi numeris dengan melakukan perhitungan nilai matematis pada memori komputer. Kondisi tersebut mengindikasikan bahwa hasil perhitungan yang dilakukan tidak selalu dapat memberikan solusi tepat terhadap setiap permasalahan dan memiliki perbedaan dalam memberikan solusi secara simbolis pada manipulasi aljabar seperti yang dapat dilakukan pada aplikasi Mathematica atau Maple.

### 1.2 Mengapa GNU Octave ?

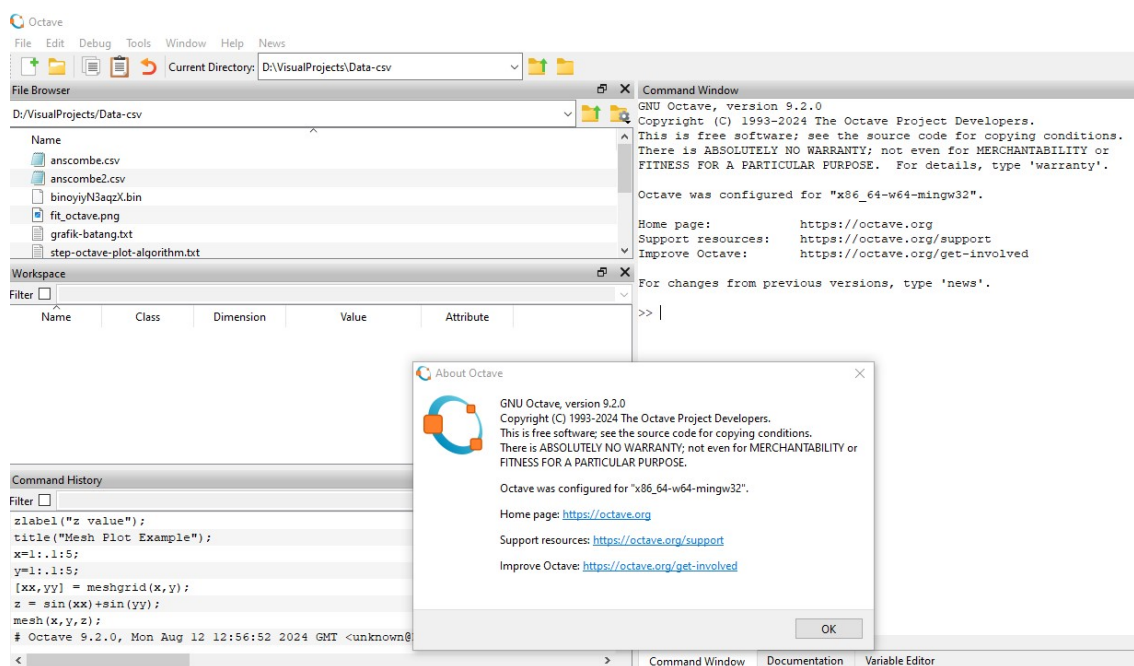
Dewasa ini terdapat bermacam-macam bahasa pemrograman komputer yang dapat menyelesaikan semua permasalahan teknis seperti penggunaan C++, Java, Python. Beberapa bahasa pemrograman tersebut merupakan bahasa pemrograman yang menjadi standar industri untuk membangun aplikasi. Permasalahannya adalah ketika bahasa pemrograman tersebut digunakan untuk membangun aplikasi menggunakan metode purwarupa atau sedang berada pada fase konsep pengembangan, seringkali terbentur kebutuhan terkait kecepatan implementasi dan waktu yang sempit untuk melakukan implementasi suatu algoritma tertentu.

Penggunaan GNU Octave secara khusus dirancang untuk memecahkan masalah sempitnya waktu pemenuhan tersebut dengan menyelesaikan perhitungan segera dan menampilkan hasilnya secepat mungkin, walaupun implementasi akhir dapat dilakukan pada bahasa pemrograman tingkat tinggi. GNU Octave memiliki banyak perangkat bantu untuk memecahkan masalah aljabar linear numeris, mencari nilai akar dari persamaan non-linear, melakukan perhitungan integral terhadap fungsi-fungsi matematika, melakukan manipulasi polinomial, melakukan perhitungan integral atas persamaan differensial dan persamaan aljabar. GNU Octave dapat juga diperluas dan dilakukan kustomisasi melalui fungsi-fungsi yang

didefinisikan oleh pengguna ditulis dalam bahasa GNU Octave atau bahasa pemrograman lainnya seperti modul pada C++, C, Fortran, Python, Java dan lainnya.

Pada saat praktisi ataupun akademisi menggunakan Matlab, maka *script* hasil pekerjaan yang digunakan pada Matlab dapat diterapkan untuk GNU Octave dengan sedikit penyesuaian pada *syntax* berekstensi file *.m*. Aplikasi Matlab ® dari Mathworks ® bersifat fleksibel, dapat diperluas cakupan pemecahan masalah simulasinya, berkinerja tinggi dengan berlangganan biaya lisensi aplikasi *proprietary* untuk dapat menggunakan aplikasi Matlab. Sedangkan GNU Octave dapat dilakukan proses unduh, instalasi dan digunakan tanpa harus berlangganan biaya lisensi pakai aplikasi karena sifat lisensinya yang GNU Public License (GPL), sehingga bersifat gratis untuk penggunaannya.

Pembaca dapat melihat lingkungan kerja GNU Octave yang terinstalasi pada sistem operasi Windows 64bit seperti terlihat pada gambar berikut :



Gambar 1.1: Lingkungan Kerja GNU Octave

### 1.3 Start, Quit, Getting Help

Pengguna GNU Octave dapat memulai menggunakan aplikasi tersebut dengan mengetikkan pada perintah Shell yaitu *octave*, melakukan *double click* pada *Octave.app* atau tindakan apa saja yang dibutuhkan sistem operasi pengguna untuk dapat menjalankan GNU Octave. Setelah GNU Octave mulai dijalankan, maka pengguna akan melihat pada *prompt*:

Listing 1.1: Prompt pada GNU Octave

```
1 >>
```

Apabila pengguna mengalami kesulitan atau masalah pada penggunaan GNU Octave pada *prompt*, maka solusinya adalah dengan melakukan interrupt GNU Octave dengan mengetikkan *Ctrl-C*. Untuk keluar dari *prompt* GNU Octave, pengguna dapat mengetikkan perintah *quit* atau *exit*:

Listing 1.2: Keluar dari Prompt GNU Octave

```
1 >> quit
```

```

2 | or
3 | >> exit

```

Apabila pengguna ingin mendapatkan bantuan atau melihat dokumentasi GNU Octave, pengguna dapat melakukan ketik pada prompt GNU Octave:

Listing 1.3: Mendapatkan bantuan atau melihat dokumentasi pada Prompt GNU Octave

```

1 | >> help
2 | or
3 | >> doc

```

Sedangkan apabila pengguna ingin mendapatkan bantuan secara spesifik terkait perintah yang akan digunakan GNU Octave, pengguna dapat melakukan ketik pada prompt GNU Octave:

Listing 1.4: Mendapatkan bantuan secara spesifik terkait suatu perintah pada Prompt GNU Octave

```

1 | >> help command
2 | contoh:
3 | >> help size
4 | >> help plot
5 | >> help figure
6 | >> help inv

```

Sedangkan apabila pengguna ingin mendapatkan bantuan dari sistem bantuan pada GNU Octave, pengguna dapat melakukan ketik pada prompt GNU Octave:

Listing 1.5: Mendapatkan bantuan dari sistem bantuan pada Prompt GNU Octave

```

1 | >> help help

```

Apabila pengguna ingin keluar dari sistem bantuan pada GNU Octave, pengguna dapat melakukan ketik pada prompt GNU Octave:

Listing 1.6: Keluar dari sistem bantuan pada Prompt GNU Octave

```

1 | >> q

```

Pada GNU Octave, konvensi penamaan yang bersifat *case-sensitive* ditulis menggunakan huruf kecil semua, seperti pada prompt GNU Octave yang memiliki konvensi campuran antara huruf besar dan huruf kecil:

Listing 1.7: Konvensi penamaan yang bersifat case-sensitive pada Prompt GNU Octave

```

1 | >> help round <akan menghasilkan:>
2 | 'round' is a built-in function from the file libinterp/corefcn/mappers.cc
3 | -- Y = round (X)
4 |     Return the integer nearest to X.
5 |     If X is complex, return round (real (X)) + round (imag (X)) * I .
6 |     If there are two nearest integers, return the one further away from zero.
7 |         round ([-2.7, 2.7])
8 |             -3     3
9 |     See also: ceil, floor, fix, roundb.
10 | Additional help for built-in functions and operators is

```

```
11 | available in the online version of the manual. Use the command
12 | 'doc <topic>' to search the manual index.
13 | Help and information about Octave is also available on the WWW
14 | at https://www.octave.org and via the help@octave.org
15 | mailing list.
16 | >> help ROUND
17 | error: help: 'ROUND' not found
18 | >>
```

---



## Bab 2

# Mulai Menggunakan GNU Octave

Beberapa variabel dan tipe data pada GNU Octave dapat berupa beberapa hal berikut ini:

- Matriks (real dan kompleks)
- String (matriks dari karakter)
- Struktur, yang terdiri dari:
  1. Vektor (merupakan matriks dengan satu kolom atau satu baris)
  2. Skalar (merupakan matriks dengan dimensi 1x1)
  3. Integer (juga merupakan tipe data double)
  4. Boolean (yang dapat berupa integer, yaitu not-null=true, 0=false)

Dimana hampir semuanya digambarkan dalam matriks pada GNU Octave, sedangkan pada Matlab memiliki beberapa tipe tambahan berupa kelas-kelas untuk orientasi obyek.

## 2.1 Variabel, Konstanta dan Tipe Data

Pada saat pengguna ingin merepresentasikan suatu matriks, maka pada GNU Octave dapat pengguna ketikkan perintah sebagai berikut :

Listing 2.1: Perintah untuk representasi matriks pada Prompt GNU Octave

```
1 >> A = [8, 2, 1; 3, -1, 4; 7, 6, -5]
2 A =
3
4     8     2     1
5     3    -1     4
6     7     6    -5
7
8 >>
```

Beranjak kepada karakter String, pengguna dapat merepresentasikan karakter String dengan menggunakan tanda petik tunggal (') dan petik ganda (") yang dapat diproses pada GNU Octave [2], seperti terlihat pada listing berikut ini:

Listing 2.2: Perintah untuk representasi karakter String pada Prompt GNU Octave

```

1 >> str = 'Hello World' (menggunakan petik tunggal)
2 str = Hello World
3 >> str = 'Hello World' (menampilkan error pada penggunaan tanda kutip tunggal terbalik
   tinggi)
4 error: parse error:
5   syntax error
6 >>> str = 'Hello World'
7         ^
8 >> str = "Hello World" (menggunakan petik ganda)
9 str = Hello World
10 >>

```

Pada pembuatan struktur untuk merepresentasikan beberapa tipe dari obyek, dapat kita lihat pada syntax GNU Octave berikut ini:

Listing 2.3: Perintah untuk representasi beberapa tipe obyek pada Prompt GNU Octave

```

1 >> data.id = 5;
2 >> data.timestamp = 1265.6983;
3 >> data.name = 'sensor 2 left';
4 >> data
5 data =
6   scalar structure containing the fields:
7     id = 5
8     timestamp = 1265.7
9     name = sensor 2 left
10 >>

```

Pada saat pengguna GNU Octave ingin membuat suatu array dari sebuah struktur, maka dapat kita lihat pada listing berikut ini:

Listing 2.4: Perintah untuk representasi suatu array dari sebuah struktur pada Prompt GNU Octave

```

1 >> data(2).id = 6;
2 >> data(2).timestamp = 1268.9713;
3 >> data(2).name = 'sensor 3 right';
4 >> data
5 data =
6   1x2 struct array containing the fields:
7     id
8     timestamp
9     name
10 >> data(2)
11 ans =
12   scalar structure containing the fields:
13     id = 6
14     timestamp = 1269.0
15     name = sensor 3 right
16 >>

```

Untuk mendapatkan presisi numeris, berupa variabel yang disimpan sebagai angka-angka double presisi pada format floating point IEEE, pengguna dapat melakukan penulisan pada prompt GNU Octave sebagai berikut:

Listing 2.5: Perintah untuk representasi presisi numeris dengan format floating point IEEE pada Prompt GNU Octave

```

1 >> realmin (menampilkan angka floating point positif terkecil)
2 ans = 2.2251e-308
3 >> realmax (menampilkan angka floating point positif terbesar)
4 ans = 1.7977e+308
5 >> eps (menampilkan presisi relatif)
6 ans = 2.2204e-16
7 >>

```

## 2.2 Ekspresi dan Operator

Ekspresi merupakan komponen pembentuk dasar dari suatu pernyataan pada GNU Octave. Ekspresi dapat dibuat dengan menggunakan nilai, variabel yang telah digunakan, juga dengan menggunakan operator, fungsi dalam tanda kurung. Ekspresi akan melakukan evaluasi terhadap suatu nilai dan dapat bertindak sebagai pernyataan atas obyeknya sendiri [3]. Secara umum, pernyataan terdiri dari ekspresi berjumlah satu atau berjumlah lebih yang menerangkan data untuk digunakan.

Pada penggunaan ekspresi, listing berikut ini dapat diterapkan pada GNU Octave:

Listing 2.6: Perintah penggunaan ekspresi pada Prompt GNU Octave

```

1 >> var = 2
2 var = 2
3 >> var += 9
4 var = 11
5 >>

```

Pada penggunaan operator, kita dapat menggunakan 2 (dua) tipe operator yaitu unary(operator yang beroperasi pada nilai tunggal atau operand) dan binary(operator yang beroperasi pada 2 nilai atau operand), listing Increment Operator (penambahan terhadap variabel) berikut ini dapat diterapkan pada GNU Octave:

Listing 2.7: Perintah penggunaan increment dan decrement operator pada Prompt GNU Octave

```

1 >> var = 1
2 var = 1
3 >> ++var
4 ans = 2
5 >> var++
6 ans = 2
7 >> var
8 var = 3
9 >> var--
10 ans = 3
11 >> --var

```

```

12 ans = 1
13 >> var
14 var = 1
15 >>

```

Pada operator aritmatika, dimana dapat kita gunakan ekspresi numeris seperti (+)penambahan, (-)pengurangan, (\*)perkalian, (/)pembagian. Pembaca dapat menggunakan listing berikut pada GNU Octave untuk mencoba operator aritmatika:

Listing 2.8: Perintah penggunaan operator aritmatika pada Prompt GNU Octave

```

1 >> a=[1,2,3]
2 a =
3     1     2     3
4 >> a*a
5 error: operator *: nonconformant arguments (op1 is 1x3, op2 is 1x3)
6 >> a.*a
7 ans =
8     1     4     9
9 >>

```

Pada operator perbandingan, dimana dapat kita gunakan ekspresi numeris seperti (<)lebih kecil, (>)lebih besar, (<=)lebih kecil sama dengan, (>=)lebih besar sama dengan, (!=)tidak sama dengan, (==)sama dengan. Pembaca dapat menggunakan listing berikut pada GNU Octave untuk mencoba operator perbandingan:

Listing 2.9: Perintah penggunaan operator perbandingan pada Prompt GNU Octave

```

1 >> a = 1
2 a = 1
3 >> b = 2
4 b = 2
5 >> a<b
6 ans = 1
7 >> a>b
8 ans = 0
9 >> a<=b
10 ans = 1
11 >> a>=b
12 ans = 0
13 >> a!=b
14 ans = 1
15 >> a==b
16 ans = 0
17 >>

```

Pada Boolean operator, dimana dapat kita gunakan ekspresi logika seperti (&) AND, (|) OR, (!) NOT, (&&) *short circuit* AND, (||) *short circuit* OR. Pembaca dapat menggunakan listing berikut pada GNU Octave untuk mencoba operator Boolean:

Listing 2.10: Perintah penggunaan operator Boolean pada Prompt GNU Octave

```
1 >> a = 1
2 a = 1
3 >> b = 0
4 b = 0
5 >> a & b
6 ans = 0
7 >> c = 1
8 c = 1
9 >> a & c
10 ans = 1
11 >> a | b
12 ans = 1
13 >> b | c
14 ans = 1
15 >> a | c
16 ans = 1
17 >> !a
18 ans = 0
19 >> !b
20 ans = 1
21 >> !c
22 ans = 0
23 >> a && b
24 ans = 0
25 >> a && c
26 ans = 1
27 >> b && c
28 ans = 0
29 >> a || b
30 ans = 1
31 >> a || c
32 ans = 1
33 >> b || c
34 ans = 1
```

## 2.3 Vektor dan Matriks

Vektor dan matriks merupakan representasi yang sangat berguna untuk menggambarkan persamaan linear dan pertidaksamaan. Secara lebih lanjut, vektor dan matriks dibutuhkan untuk merepresentasikan permasalahan pada pemrograman linear. GNU Octave dapat digunakan sebagai bahasa pemrograman ilmiah untuk mendukung penggunaan matriks dan vektor, seperti halnya dengan Matlab. Suatu vektor merupakan kumpulan angka yang tersusun dalam bentuk baris atau bentuk kolom, sedangkan matriks merupakan suatu array angka dengan jumlah baris dan kolom yang lebih besar [3]. Suatu vektor baris dapat dilakukan inisialisasi dengan penulisan angka-angka dalam braket yang dipisahkan oleh koma atau

spasi. Pembaca dapat melihat penulisan listing untuk penggambaran matriks baris, seperti terlihat pada listing berikut:

Listing 2.11: Perintah untuk representasi vektor baris pada Prompt GNU Octave

```

1 >> rowVector1 = [1,2,3,4]
2 rowVector1 =
3     1     2     3     4
4 >> rowVector2 = [5 6 7 8]
5 rowVector2 =
6     5     6     7     8
7 >>

```

Pembaca juga dapat menggunakan fungsi `linspace` untuk membuat vektor baris dengan jarak antar elemen yang sama. Notasinya berupa:  $vectorName = linspace(angkaAwal, angkaAkhir, jumlahElemen)$ , seperti terlihat pada listing berikut:

Listing 2.12: Perintah untuk representasi fungsi `linspace` untuk vektor baris pada Prompt GNU Octave

```

1 >> rowVector3 = linspace(1, 10, 3)
2 rowVector3 =
3     1.0000     5.5000    10.0000
4 >> rowVector3 = linspace(1, 10, 10)
5 rowVector3 =
6     1     2     3     4     5     6     7     8     9    10
7 >> rowVector3 = linspace(1, 12, 15)
8 rowVector3 =
9 Columns 1 through 8:
10     1.0000     1.7857     2.5714     3.3571     4.1429     4.9286     5.7143     6.5000
11 Columns 9 through 15:
12     7.2857     8.0714     8.8571     9.6429    10.4286    11.2143    12.0000
13 >>

```

Pembaca dapat menggunakan vektor kolom yang diinisialisasikan dengan angka didalam braket yang dipisahkan oleh tanda titik koma atau baris baru antar angka.

Listing 2.13: Perintah untuk representasi vektor kolom pada Prompt GNU Octave

```

1 >> colVect1 = [1;2;3;4;5;6;7;8;9;10]
2 colVect1 =
3
4     1
5     2
6     3
7     4
8     5
9     6
10    7
11    8
12    9
13   10

```

```

14
15 >> colVect2 = [1
16 3
17 5
18 7
19 9
20 11
21 13
22 17
23 19
24 ]
25 colVect2 =
26
27     1
28     3
29     5
30     7
31     9
32    11
33    13
34    17
35    19
36
37 >>

```

Suatu matriks dapat diinisialisasikan menggunakan vektor baris dan vektor kolom secara bersamaan. Pembaca dapat melihat listing berikut untuk menggambarkan matriks pada GNU Octave:

Listing 2.14: Perintah untuk representasi matriks pada Prompt GNU Octave

```

1 >> matriks1 = [1,2,3,4;5,6,7,8;9,10,11,12]
2 matriks1 =
3
4     1     2     3     4
5     5     6     7     8
6     9    10    11    12
7
8 >> matriks2 = [13 14 15 16
9 17 18 19 20
10 21 22 23 24]
11 matriks2 =
12
13    13    14    15    16
14    17    18    19    20
15    21    22    23    24
16
17 >>

```

Cara alternatif untuk representasi matriks dengan mengetahui terlebih dahulu koordinat posisi angka-angka anggota matriks tersebut, seperti terlihat pada listing berikut ini:

Listing 2.15: Perintah untuk alternatif representasi matriks pada Prompt GNU Octave

```
1 >> matriks3(1:2,1:2) = 9
2 matriks3 =
3
4     9     9
5     9     9
6
7 >> matriks4(2,2) = 8
8 matriks4 =
9
10    0     0
11    0     8
12
13 >> matriks5(1:3,1:3) = 10
14 matriks5 =
15
16    10    10    10
17    10    10    10
18    10    10    10
19
20 >>
```

Representasi matriks identitas dimana diagonal dari matriks tersebut berisi angka 1, dapat menggunakan fungsi `eye`, seperti terlihat pada listing GNU Octave berikut:

Listing 2.16: Perintah untuk alternatif representasi matriks pada Prompt GNU Octave

```
1 >> matriks6 = eye(3)
2 matriks6 =
3
4 Diagonal Matrix
5
6     1     0     0
7     0     1     0
8     0     0     1
9
10 >> matriks7 = eye(4)
11 matriks7 =
12
13 Diagonal Matrix
14
15     1     0     0     0
16     0     1     0     0
17     0     0     1     0
18     0     0     0     1
```



```
19
20 >>
```

Operasi penambahan dan pengurangan pada matriks dapat pembaca lihat pada listing GNU Octave berikut:

Listing 2.17: Perintah untuk operasi pada matriks di Prompt GNU Octave

```
1 >> matriks1 + matriks2
2 ans =
3
4     14     16     18     20
5     22     24     26     28
6     30     32     34     36
7
8 >> matriks2 - matriks1
9 ans =
10
11     12     12     12     12
12     12     12     12     12
13     12     12     12     12
14
15 >> matriks1*4
16 ans =
17
18     4     8     12     16
19     20    24    28    32
20     36    40    44    48
21
22 >> matriks2/5
23 ans =
24
25     2.6000    2.8000    3.0000    3.2000
26     3.4000    3.6000    3.8000    4.0000
27     4.2000    4.4000    4.6000    4.8000
28
29 >>
```

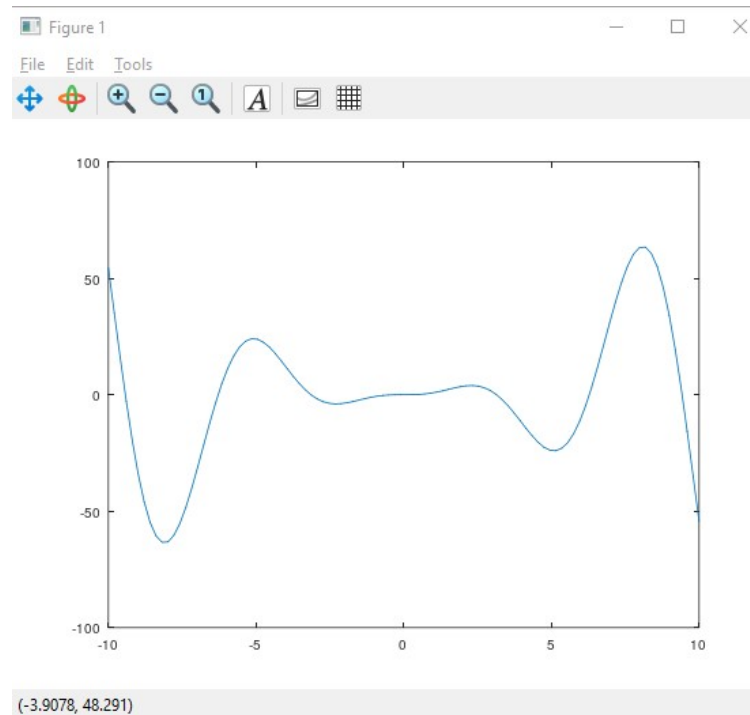
## 2.4 Plotting

Plot pada GNU Octave dapat berupa plot 2 Dimensi (2D) maupun plot 3 Dimensi (3D). Untuk membuat plot 2D pada GNU Octave, pembaca dapat melihat listing berikut:

Listing 2.18: Perintah untuk plotting 2D Sinus pada Prompt GNU Octave

```
1 >> x = linspace(-10, 10, 100);
2 >> plot(x, x.^2.*sin(x))
3 >>
```

Sehingga menghasilkan grafik pada gambar 2.1 berikut ini: Sedangkan untuk grafik cosinus pada GNU



Gambar 2.1: Hasil Plotting 2D sinus pada GNU Octave

Octave, pembaca dapat melihat listing berikut:

Listing 2.19: Perintah untuk plotting 2D Cosinus pada Prompt GNU Octave

```
1 >> x = linspace(-10, 10, 100);
2 >> plot(x, x.^2.*cos(x))
3 >>
```

Sehingga menghasilkan grafik pada gambar 2.2 berikut ini: Untuk menghasilkan plotting 3D, maka pembaca dapat mencoba listing berikut ini pada GNU Octave:

Listing 2.20: Perintah untuk plotting 3D pada Prompt GNU Octave

```
1 >> t = 0:0.1:10*pi;
2 >> r = linspace(0, 1, numel(t));
3 >> z = linspace(0, 1, numel(t));
4 >> plot3(r.*sin(t), r.*cos(t), z);
```

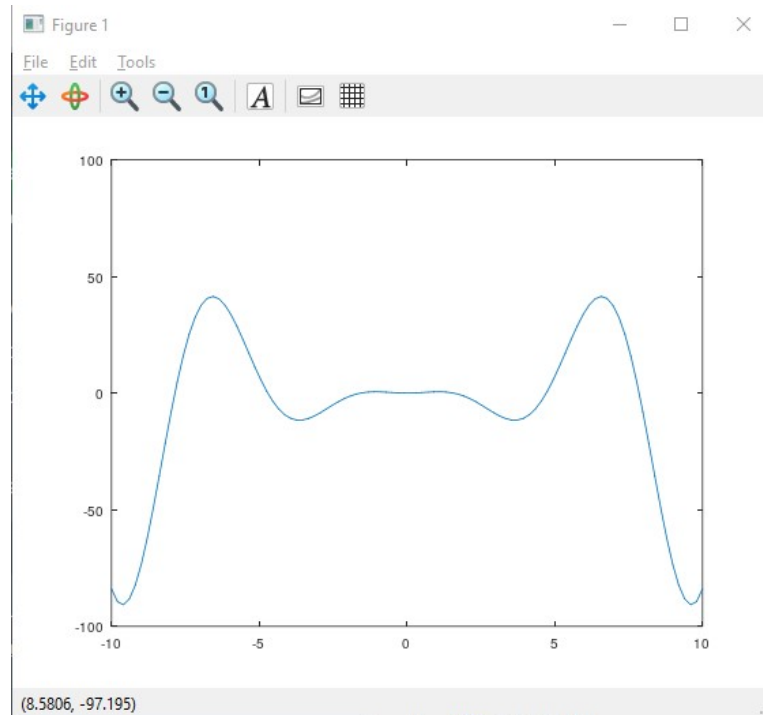
Sehingga menghasilkan grafik 3D Helix pada gambar 2.3 berikut ini:

## 2.5 Algoritma dan *Script*

Untuk dapat menjalankan algoritma pada script GNU Octave, pembaca dapat melakukan penulisan scripting pada file GNU Octave berekstensi .m, misal: areaCircle.m, menggunakan editor default pada aplikasi GNU Octave, dengan contoh sebagai berikut:

Listing 2.21: Scripting untuk menghitung luas lingkaran pada Prompt GNU Octave

```
1 | clc;
```



Gambar 2.2: Hasil Plotting 2D cosinus pada GNU Octave

```

2 clear all;
3 close all;
4 Radius = 5.6;
5 Area = pi*Radius^2;
6 disp(['Area = ' num2str(Area)]);

```

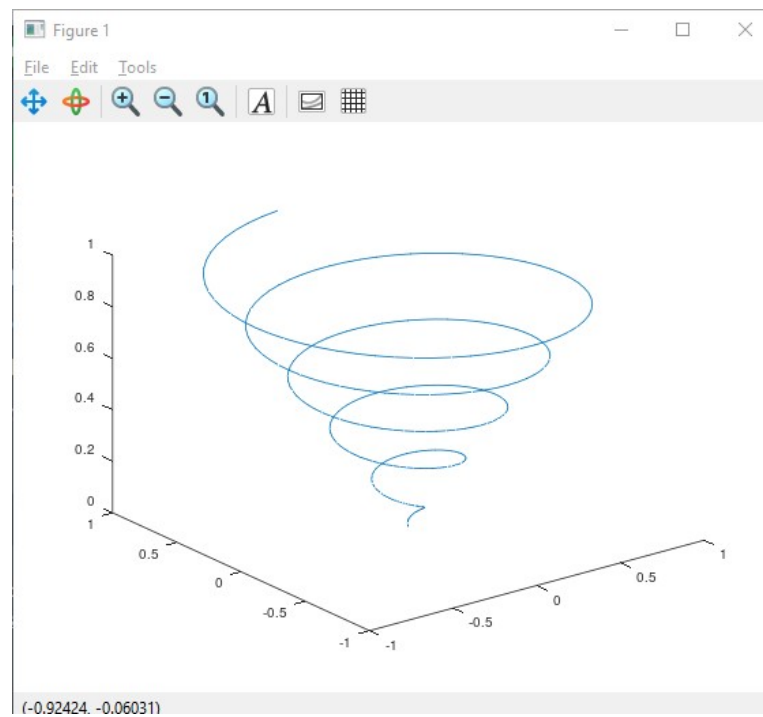
Apabila dijalankan dengan mengetikkan pada prompt GNU Octave sebagai berikut: `runareaCircle` Maka akan menghasilkan luasan pada prompt GNU Octave:

Listing 2.22: Hasil running script untuk menghitung luas lingkaran pada Prompt GNU Octave

```

1 Area =98.5203
2 >>

```



Gambar 2.3: Hasil Plotting 3D Helix pada GNU Octave

## Bab 3

# Representasi Data

### 3.1 Simulasi *Distributed System*

Untuk melakukan simulasi Distributed System, maka pembaca dapat mencoba beberapa script [4] pada listing berikut ini:

Listing 3.1: Perintah untuk simulasi Dist.System shrink factor pada Prompt GNU Octave

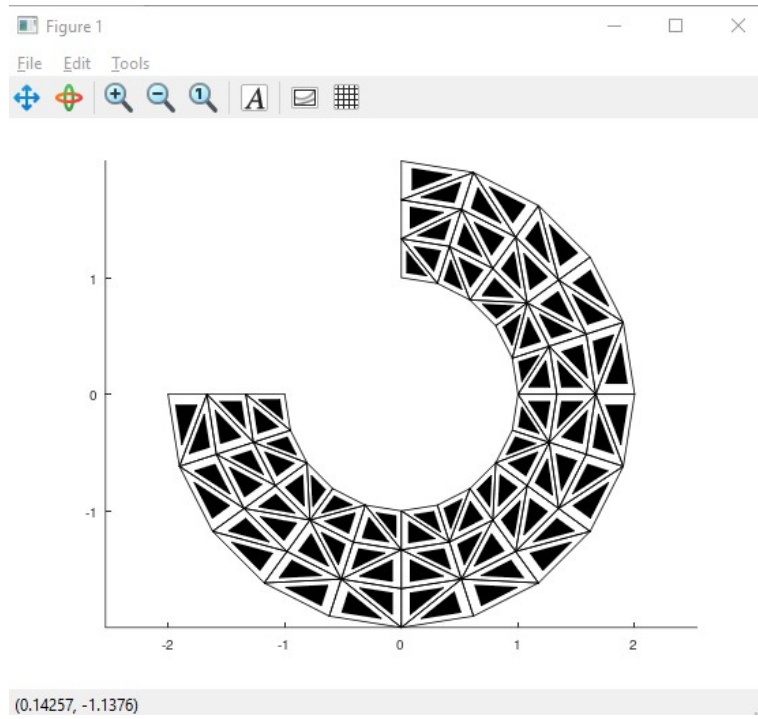
```
1 >> [phi r] = meshgrid (linspace (0, 1.5*pi, 16), linspace (1, 2, 4));
2 >> tri = delaunay (phi(:), r(:));
3 >> v = [r(:).*sin(phi(:)) r(:).*cos(phi(:))];
4 >> clf ()
5 >> p = patch ("Faces", tri, "Vertices", v, "FaceColor", "none");
6 >> fv = shrinkfaces (p);
7 >> patch (fv)
8 >> axis equal
9 >> grid on
```

Sehingga menghasilkan grafik triangulasi 3/4 lingkaran pada gambar 3.1 berikut ini: Untuk mendapatkan hasil penggambaran mesh grid yang digunakan untuk menghasilkan input 2D atau 3D yang telah dilakukan plotting.

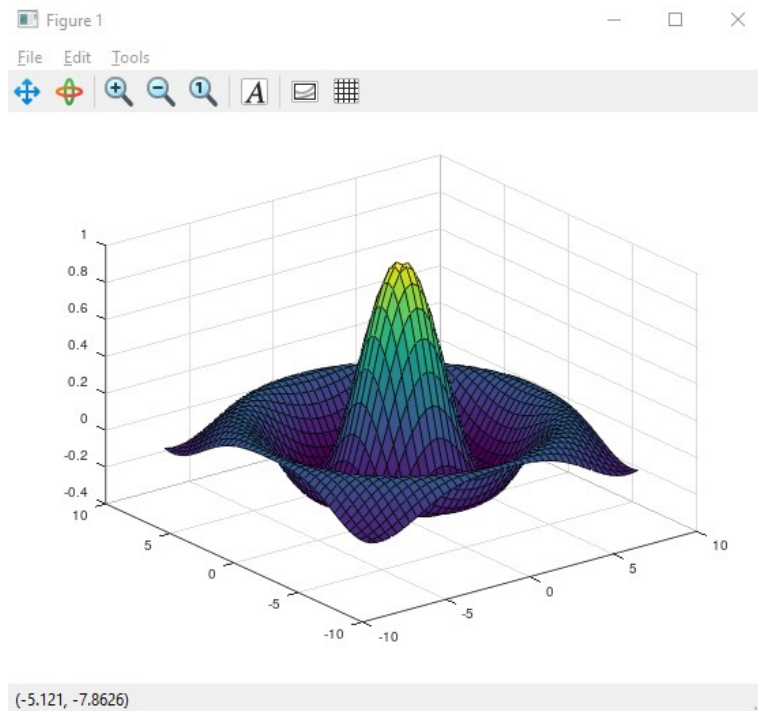
Listing 3.2: Perintah untuk simulasi Dist.System mesh grid pada Prompt GNU Octave

```
1 >> f = @(x,y) sin (sqrt (x.^2 + y.^2)) ./ sqrt (x.^2 + y.^2);
2 >> range = linspace (-8, 8, 41);
3 >> [X, Y] = meshgrid (range, range);
4 >> Z = f (X, Y);
5 >> surf (X, Y, Z);
6 >>
```

Sehingga menghasilkan grafik 3D seperti topi sombrero pada gambar 3.2 berikut ini:



Gambar 3.1: Hasil Shrink Factor untuk  $3/4$  Triangulasi Lingkaran pada GNU Octave



Gambar 3.2: Hasil 3D Plotting terkait mesh grid pada GNU Octave

# Bibliography

- [1] D. o. M. University of Pittsburgh, “Mathematical software,” accessed August 15th, 2024. [Online]. Available: <https://www.mathematics.pitt.edu/mathematical-software> 1
- [2] K. Arras, “Octave/matlab tutorial for social robotics lab,” accessed July 12th, 2024. [Online]. Available: <http://ais.informatik.uni-freiburg.de/teaching/ws11/robotics2/pdfs/rob2-03-octave.pdf> 7
- [3] P. Kumar, “Gnu octave – scientific programming language,” accessed July 20th, 2024. [Online]. Available: <https://techniex.com/gnu-octave-scientific-programming-language/> 9, 11
- [4] J. Lachniet, “Introduction to gnu octave: A brief tutorial for linear algebra and calculus students,” accessed July 18th, 2024. [Online]. Available: <https://www.wcc.vccs.edu/sites/default/files/Introduction-to-GNU-Octave.pdf> 19