

UJIAN AKHIR SEMESTER
PRAKTIKUM DESAIN DAN ANALISIS ALGORITMA

Asisten Laboratorium: Petrus Marcellino Tampubolon



Oleh Kelompok 3:

Sutriyaningsih	211401007
Zaki Afwanul Hilmi Lubis	211401010
Yessica Angelita Situmorang	211401025

PROGRAM STUDI ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA

2022

Algoritma

Sorting Dinamis

- a. Berikut ini merupakan tampilan awal pada output sorting dinamis.

```
C:\Users\LENOVO\Documents\SEMESTER 3\Lab DAA\UTS_Dynamic Sorted.exe
List Kosong
-----
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan:
```

Akan ditampilkan keterangan list kosong yang artinya belum ada data yang di-inputkan. Dapat dilihat juga terdapat lima menu pilihan yang terdiri dari input data, cari data, hapus data, edit data, dan exit. User akan diminta untuk menginputkan pilihan yang diinginkan.

- b. Selanjutnya jika user menginputkan pilihan input data maka akan muncul tampilan output seperti di bawah ini.

```
List Kosong
-----
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan: 1
Masukkan NIM: 510
Input nama   : Reza
Input umur   : 20
Input tinggi : 160
Input berat  : 80
```

User akan diminta untuk menginputkan NIM, nama, umur, tinggi, dan berat badan untuk selanjutnya program akan melakukan perhitungan BMI.

Jika user meng-klik enter, maka selanjutnya program akan menampilkan output hasil perhitungan dan keterangan BMI dari data yang telah di-inputkan

```
DATA BERAT BADAN MAHASISWA MENURUT RUMUS BODY MASS INDEX
-----
No.  NIM NAMA      UMUR  TB  BB  BMI  KETERANGAN
-----
1    510 Reza      20    160 80  80    OBESITAS
-----
KETERANGAN:
-BMI < 18 = BERAT BADAN KURANG
-BMI 18-24= BERAT BADAN IDEAL
-BMI > 30 = OBESITAS
-----
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan: _
```

Dari output di atas dapat dilihat hasil perhitungan BMI dari data yang telah di-inputkan serta keterangan BMI untuk berat badan kurang, ideal, dan obesitas. Terdapat juga menu pilihan di bagian bawah dimana user dapat melakukan input data lagi, mencari data, menghapus data, melakukan edit data, dan exit.

- c. Selanjutnya untuk pilihan edit data, user juga akan diminta untuk menginputkan NIM dari data yang ingin diedit.

```
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan: 4
Input NIM: 510_
```

```
EDIT DATA BERIKUT INI
NIM      : 510
NAMA     : Reza
UMUR     : 20
TINGGI   : 160
BERAT    : 80

PROSES EDIT DATA
NIM      : 511
NAMA     : Reza
UMUR     : 20
TINGGI   : 165
BERAT    : 85
DATA BERHASIL DIEDIT.
TEKAN ENTER UNTUK MELANJUTKAN_
```

Pada menu edit data akan ditampilkan data dari NIM yang dipilih untuk dilakukan pengeditan sehingga memudahkan user untuk melakukan pengeditan dan menghindari terjadinya kesalahan. Pada proses edit data user diminta untuk menginputkan ulang semua data dan akan muncul keterangan data berhasil diedit.

- d. Langkah selanjutnya jika user menginputkan pilihan cari data maka user akan diminta untuk menginputkan NIM dari data mahasiswa yang sebelumnya telah diinputkan.

```
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan: 2
Masukkan NIM: 510_
```

Jika user telah menginputkan NIM maka akan muncul tampilan output seperti berikut.

```
DATA BERHASIL DITEMUKAN
NAMA     :Reza
NIM      :510
UMUR     :20
TINGGI   :160
BERAT    :80
TEKAN ENTER UNTUK MELANJUTKAN PROSES
```

- e. Jika user mengklik enter maka output akan kembali pada tampilan sebelumnya.

```
DATA BERAT BADAN MAHASISWA MENURUT RUMUS BODY MASS INDEX
-----
No.  NIM  NAMA      UMUR  TB  BB  BMI  KETERANGAN
-----
1    510  Reza          20   160 80 80  OBESITAS
-----
KETERANGAN:
-BMI < 18 = BERAT BADAN KURANG
-BMI 18-24= BERAT BADAN IDEAL
-BMI > 30 = OBESITAS
-----
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan:
```

Selanjutnya akan dilakukan penghapusan data

```
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan: 3
Input NIM: 510
```

User akan diminta kembali untuk menginputkan NIM dari data yang telah di-inputkan sebagai penunjuk agar program bisa melakukan penghapusan data. Jika data telah berhasil dihapus akan ditampilkan output seperti di bawah ini.

```
DATA BERHASIL DIHAPUS.
TEKAN ENTER UNTUK MELANJUTKAN PROSES
```

Dan tampilan program juga akan kembali seperti tampilan awal sebelum ada data yang diinputkan.

```
List Kosong
-----
Menu Pilihan
1. Input Data
2. Cari Data
3. Hapus Data
4. Edit Data
5. Exit
Pilihan:
```

Namun, jika user salah menginputkan NIM dari data yang telah dihapus atau tidak terdapat NIM dari data yang telah di-inputkan, maka program tidak berhasil melakukan penghapusan data sehingga akan muncul tampilan seperti di bawah ini.

```
DATA GAGAL DIHAPUS.
TEKAN ENTER UNTUK MELANJUTKAN PROSES
```

Sorting Statis

- a. Pada program sorting statis akan muncul tampilan awal program seperti berikut.

```
===== PILIHAN MENU DATA STATIS =====  
1. Input Data  
2. Tampilkan  
3. Hapus  
4. Keluar  
>>
```

Pada tampilan awal terdapat pilihan menu yang terdiri dari input data, tampilkan data, hapus data, serta keluar.

- b. Jika user ingin menambahkan data, maka dapat memilih menu input data.

```
Nama Mahasiswa : Reza  
NIM : 510  
Umur : 19  
Tinggi Badan(cm) : 20  
Berat Badan(kg) : 80
```

Selanjutnya user akan diminta untuk menginputkan nama, NIM, umur, tinggi badan, dan berat badan yang ingin dilakukan perhitungan BMI nya.

Jika sudah, maka akan muncul tampilan output seperti di bawah ini.

```
Nama Mahasiswa : Reza  
NIM : 510  
Umur : 20  
Tinggi Badan(cm) : 160  
Berat Badan(kg) : 80  
  
OBESITAS  
  
===== PILIHAN MENU DATA STATIS =====  
1. Input Data  
2. Tampilkan  
3. Hapus  
4. Keluar  
>>
```

Pada tampilan output akan ditampilkan data yang telah diinputkan sebelumnya dan keterangan hasil perhitungan BMI. Pada bagian bawah juga terdapat pilihan menu untuk memudahkan user jika ingin melakukan penambahan data, menampilkan data, menghapus data, dan keluar dari program.

Jika ingin menambahkan data lagi, user dapat memilih menu input data dan program akan meminta kembali agar user menginputkan data yang sama.

- c. Jika user ingin menampilkan data-data yang sebelumnya telah ditambahkan, user dapat memilih menu tampilkan data dan akan muncul tampilan output seperti di bawah ini.

```

Data ke- 1
Nama           : Kiku
NIM            : 250
Umur           : 20
Tinggi Badan (cm) : 155
Berat Badan (kg) : 75
OBESITAS

Data ke- 2
Nama           : Reza
NIM            : 510
Umur           : 20
Tinggi Badan (cm) : 160
Berat Badan (kg) : 80
OBESITAS

===== PILIHAN MENU DATA STATIS =====

1. Input Data
2. Tampilkan
3. Hapus
4. Keluar
>>

```

- d. Selanjutnya jika user ingin menghapus data-data yang telah diinputkan, user dapat memilih menu hapus data. Pada menu ini user akan diminta untuk menginputkan data keberapa yang ingin dihapus.

```

.....
| Pilih Data Yang Ingin Dihapus |
.....

PILIH NOMOR DATA YANG INGIN DIHAPUS : 2

===== PILIHAN MENU DATA STATIS =====

1. Input Data
2. Tampilkan
3. Hapus
4. Keluar
>> 2

```

```

Data ke- 1
Nama           : Kiku
NIM            : 250
Umur           : 20
Tinggi Badan (cm) : 155
Berat Badan (kg) : 75
OBESITAS

===== PILIHAN MENU DATA STATIS =====

1. Input Data
2. Tampilkan
3. Hapus
4. Keluar
>>

```

Setelah itu program akan menampilkan pilihan menu seperti di atas yang menandakan bahwa data berhasil dihapus dan program dapat terus dijalankan. Namun, jika user salah menginputkan data beberapa yang akan dihapus maka akan muncul tampilan seperti di bawah ini.

```
.....
| Pilih Data Yang Ingin Dihapus |
.....

PILIH NOMOR DATA YANG INGIN DIHAPUS : 4

DATA YANG INGIN DIHAPUS TIDAK ADA !!!

===== PILIHAN MENU DATA STATIS =====

1. Input Data
2. Tampilkan
3. Hapus
4. Keluar
>>
```

- e. Pada menu terakhir yaitu menu keluar, jika user menginputkan pilihan menu empat maka program akan langsung berhenti dengan menampilkan tampilan seperti di bawah ini.

```
-----
Process exited after 309.9 seconds with return value 1
Press any key to continue . . . _
```

End Tagged Dense Code (ETDC)

Data Dinamis

NIM : 510

Nama : Reza

Umur : 20

Tinggi Badan : 160

Berat Badan : 80

OBESITAS

NIM : 250

Nama : Kiku

Umur : 20

Tinggi Badan : 155

Berat Badan : 75

OBESITAS

String: "510 Reza 20 160 80 OBESITAS 250 Kiku 20 155 75 OBESITAS"

Tabel Data

Tabel data dibuat sebelum dilakukan kompresi dengan urutan charset berdasarkan frekuensi terbesar.

Char	Freq	ASCII Code	Bit	Bit x Freq
space	9	00100000	8	72
0	6	00000000	8	48
5	5	00000101	8	40
S	4	01010011	8	32
1	3	00000001	8	24
2	3	00000010	8	24
O	2	01001111	8	16
B	2	01000010	8	16
E	2	01000101	8	16
I	2	01001001	8	16
T	2	01010100	8	16
A	2	01000001	8	16
R	1	01010010	8	8
e	1	01100101	8	8
z	1	01111010	8	8
a	1	01100001	8	8
6	1	00000110	8	8

8	1	00001000	8	8
K	1	01001011	8	8
i	1	01101001	8	8
k	1	0110101	8	8
u	1	01110101	8	8
7	1	00000111	8	8
Total				424

B yang dipakai pada perhitungan ini adalah 6, jadi setiap karakter saat pengubahan ke dalam End-Tagged Dense Code akan diubah menjadi codeword yang jumlah bit minimal 3 dan selanjutnya akan memiliki jumlah bit kelipatan 6.

Tabel data setelah dikompresi

i	Char	Freq	Codeword	Bit	Bit x Freq
0	space	9	100000	6	54
1	0	6	100001	6	36
2	5	5	100010	6	30
3	S	4	100011	6	24
4	1	3	100100	6	18
5	2	3	100101	6	12
6	O	2	100110	6	12
7	B	2	100111	6	12
8	E	2	101000	6	12
9	I	2	101001	6	12
10	T	2	101010	6	12
11	A	2	101011	6	12
12	R	1	101100	6	6
13	e	1	100101	6	6
14	z	1	101110	6	6
15	a	1	101111	6	6
16	6	1	110000	6	6
17	8	1	110001	6	6
18	K	1	110010	6	6

19	i	1	110011	6	6
20	k	1	110100	6	6
21	u	1	110101	6	6
22	7	1	110110	6	6
Total					312

Total bit hasil kompresi adalah 312 bit. Selanjutnya total bit dibagi dengan 8 menyisakan 0 sehingga didapat:

String bit: 312 bit

: 39 byte + 0 bit

Padding : -

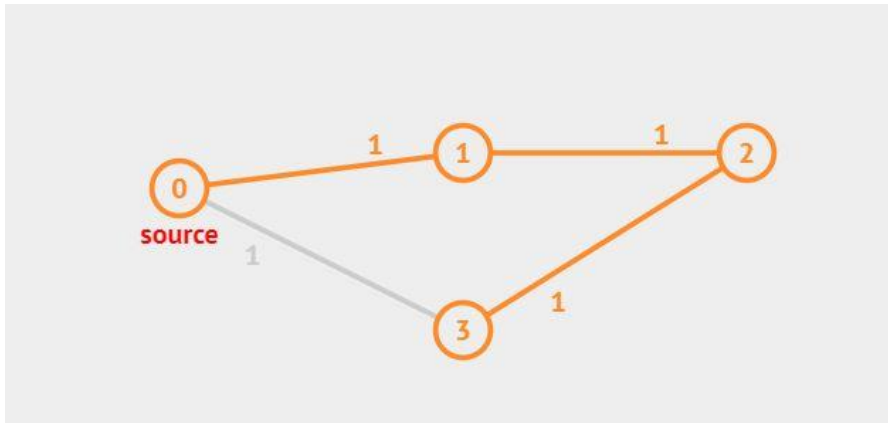
Flag : 00000000

String : “510 Reza 20 160 80 OBESITAS 250 Kiku 20 155 75 OBESITAS”

```
100010100100100001100000101100100101101110101111100000100101100001100000
100100110000100001100000110001100001100000100110100111101000100011101001
101010101011100011100000100101100101100001100000110010110011110100110101
100000100101100001100000100100100010100010100000110110100010100000100110
10011110100010001110100110101010101110001100000000
```

Minimum Spanning Tree (MST)

Algoritma Sorting Dinamis



Keterangan:

0: Input data

1: Edit data

2: Cari data

3: Hapus Data