

**LAPORAN PRAKTIKUM  
ALGORITMA & STRUKTUR DATA  
MODUL 3**



**SINGLE LINK LIST**

**Oleh:**

**Indra Suryadilaga      NIM. 2410817310014**

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS TEKNIK  
UNIVERSITAS LAMBUNG MANGKURAT  
APRIL 2025**

**LEMBAR PENGESAHAN**  
**LAPORAN PRAKTIKUM ALGORITMA & STRUKTUR DATA**  
**MODUL 2**

Laporan Praktikum Algoritma & Struktur Data Modul 3: Single Link List ini disusun sebagai syarat lulus mata kuliah Praktikum Algoritma & Struktur Data. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Indra Suryadilaga  
NIM : 2410817310014

Menyetujui,  
Asisten Praktikum

Mengetahui,  
Dosen Penanggung Jawab Praktikum

Zulfa Auliya Akbar  
NIM. 2210817210010

Andreyan Rizky Baskara, S.Kom., M.Kom  
NIP. 199307032019031011

## DAFTAR ISI

LEMBAR PENGESAHAN .....	2
DAFTAR ISI.....	3
DAFTAR GAMBAR .....	5
DAFTAR TABEL.....	<b>Error! Bookmark not defined.</b>
SOAL 1 .....	6
A. Screenshoot .....	6
SOAL 2 .....	7
A. Screenshoot .....	7
SOAL 3 .....	8
A. Screenshoot .....	8
SOAL 4 .....	9
A. Screenshoot .....	9
SOAL 5 .....	10
A. Screenshoot .....	10
SOAL 6 .....	11
A. Screenshoot .....	11
SOAL 7 .....	12
A. Screenshoot .....	12
SOAL 8 .....	13
A. Screanshoot .....	13
B. Pembahasan.....	13
SOAL 9 .....	14

A. Pembahasan.....	14
SOAL 10 .....	15
B. Pembahasan.....	15
GITHUB.....	16

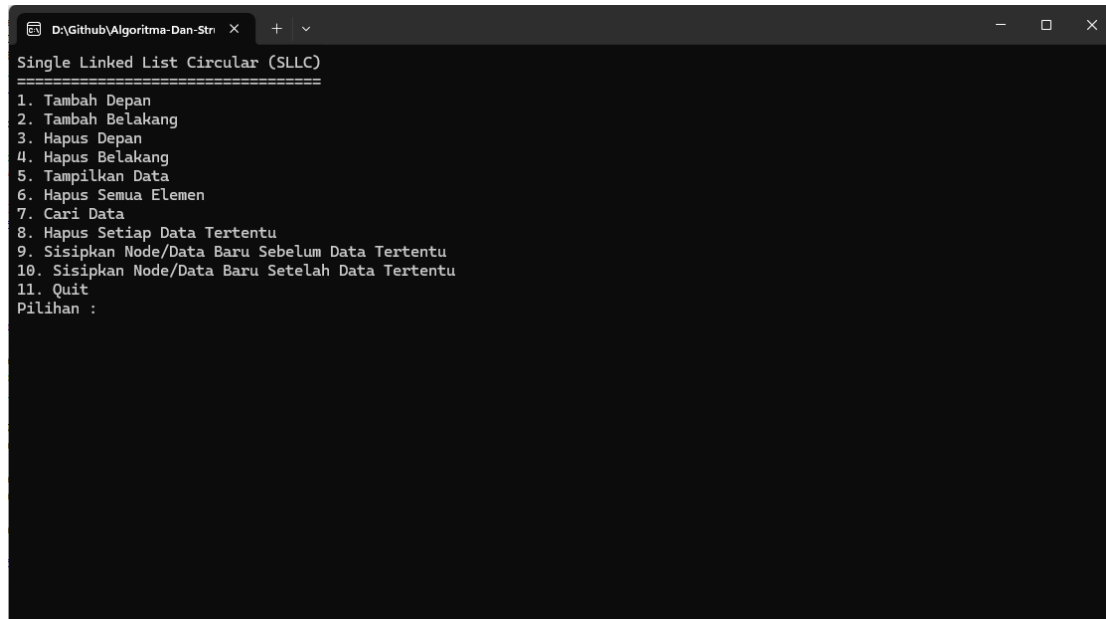
## **DAFTAR GAMBAR**

Gambar 1. Screanshoot Hasil Perintah Soal 1 .....	6
Gambar 2. Screanshoot Hasil Perintah Soal 2 .....	7
Gambar 3. Screanshoot Hasil Perintah Soal 3 .....	8
Gambar 4. Screanshoot Hasil Perintah Soal 4 .....	9
Gambar 5. Screanshoot Hasil Perintah Soal 5 .....	10
Gambar 6. Screanshoot Hasil Perintah Soal 6 .....	11
Gambar 7. Screanshoot Hasil Perintah Soal 7 .....	12
Gambar 8. Screanshoot Hasil Perintah Soal 8 .....	13

## SOAL 1

Cobalah program berikut, running, simpan program, dan screenshoot hasil running !

### A. Screenshoot



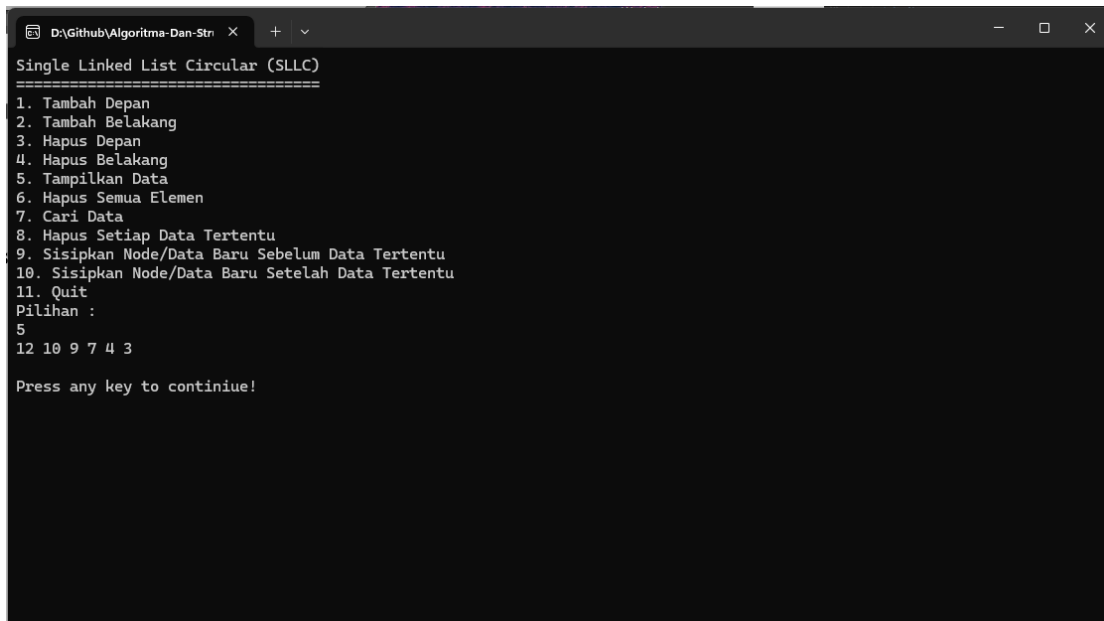
```
D:\Github\Algoritma-Dan-Stri >
Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan :
```

*Gambar 1. Screenshoot Hasil Perintah Soal 1*

## SOAL 2

Lakukan tambah data depan 3, 4, 7, 9, 10, 12 dan kemudian lakukan tampilkan data lalu screenshot hasilnya !

### A. Screenshot



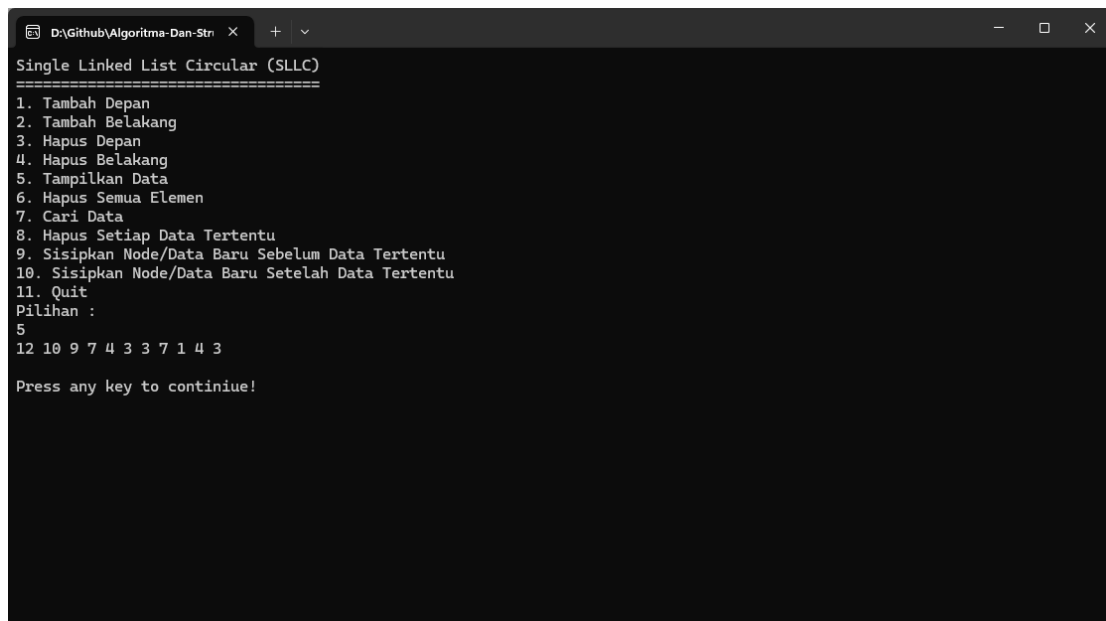
```
D:\Github\Algoritma-Dan-Str > Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan :
5
12 10 9 7 4 3
Press any key to continue!
```

Gambar 2. Screenshot Hasil Perintah Soal 2

### SOAL 3

Lakukan tambah data belakang 3, 7, 1, 4, 3 dan kemudian lakukan tampilkan data lalu screenshot hasilnya !

#### A. Screenshot



```
D:\Github\Algoritma-Dan-Str > Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan :
5
12 10 9 7 4 3 3 7 1 4 3
Press any key to continue!
```

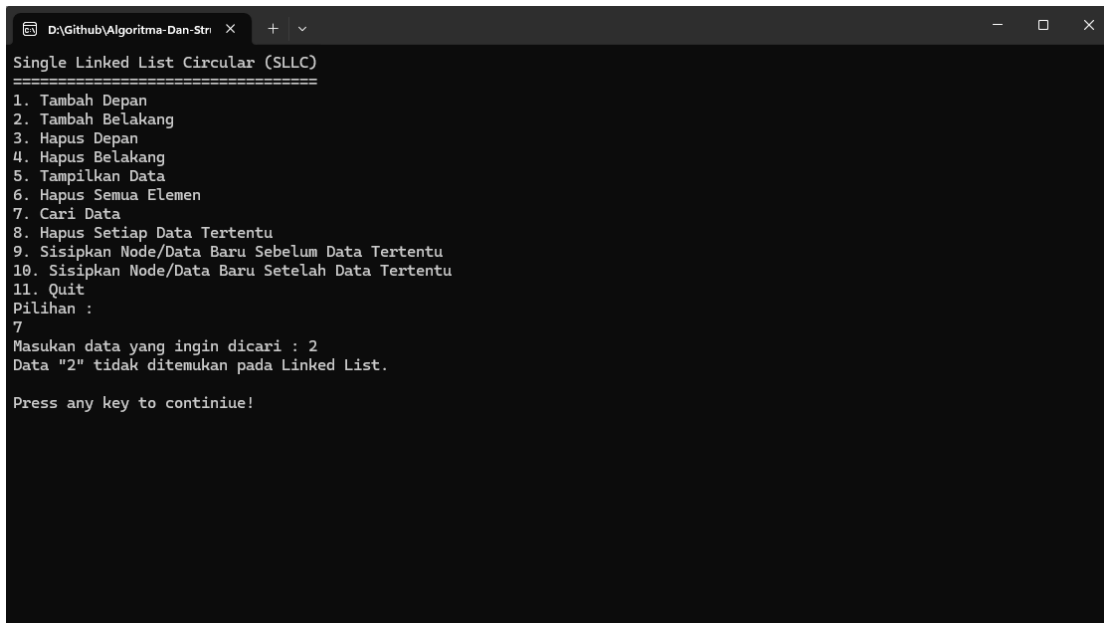
Gambar 3. Screenshot Hasil Perintah Soal 3



## SOAL 4

Apa yang terjadi jika mencari angka 2 pada Single Linked List Circular (SLLC) pada data yang telah ditambahkan/dimasukkan sebelumnya dan screenshot hasilnya

### A. Screenshot



```
D:\Github\Algoritma-Dan-Str > Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan :
7
Masukan data yang ingin dicari : 2
Data "2" tidak ditemukan pada Linked List.

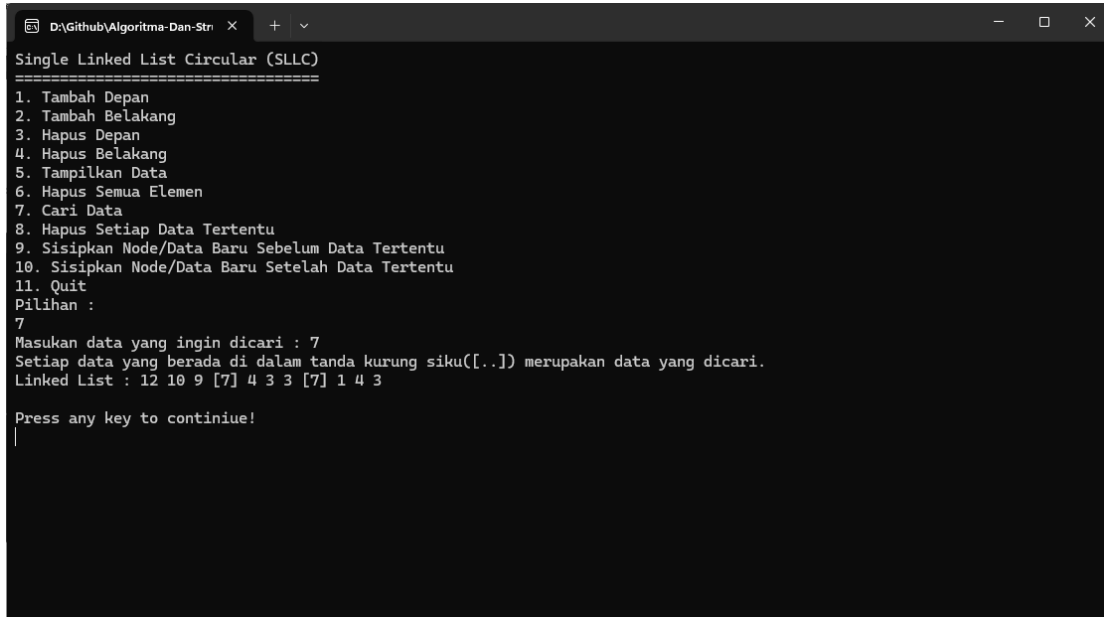
Press any key to continue!
```

Gambar 4. Screenshot Hasil Perintah Soal 4

## SOAL 5

Coba cari angka 7 dan screenshoot hasilnya !

### A. Screenshoot



```
D:\Github\Algoritma-Dan-Stri > Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan : 7
Masukan data yang ingin dicari : 7
Setiap data yang berada di dalam tanda kurung siku([..]) merupakan data yang dicari.
Linked List : 12 10 9 [7] 4 3 3 [7] 1 4 3

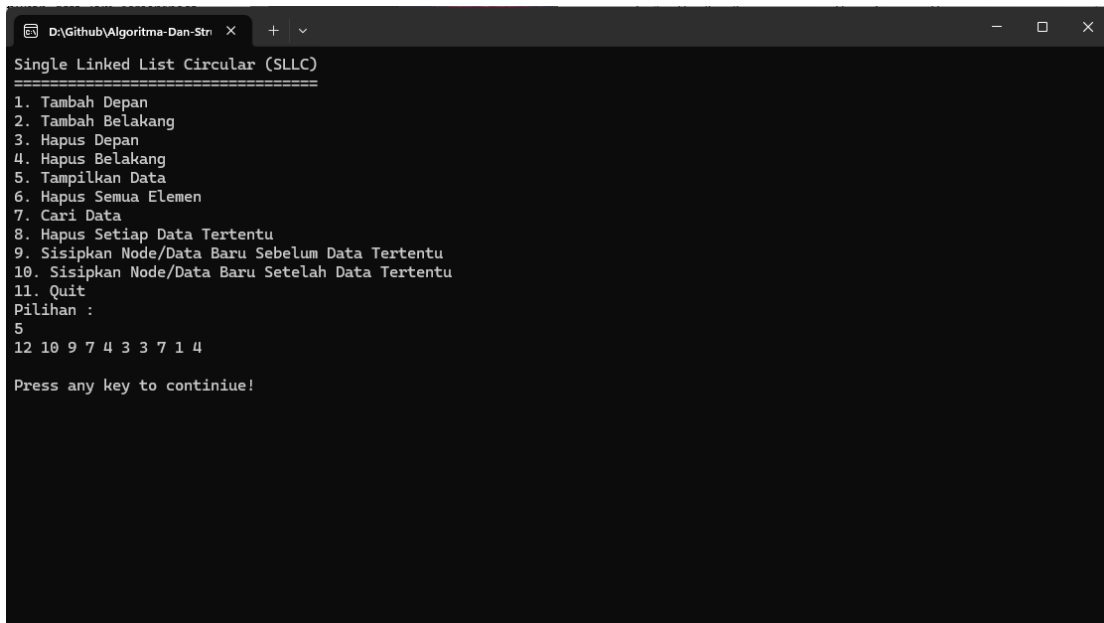
Press any key to continue!
```

Gambar 5. Screenshoot Hasil Perintah Soal 5

## SOAL 6

Lakukan hapus belakang dan kemudian lakukan tampilkan data lalu screenshot hasilnya !

### A. Screenshot



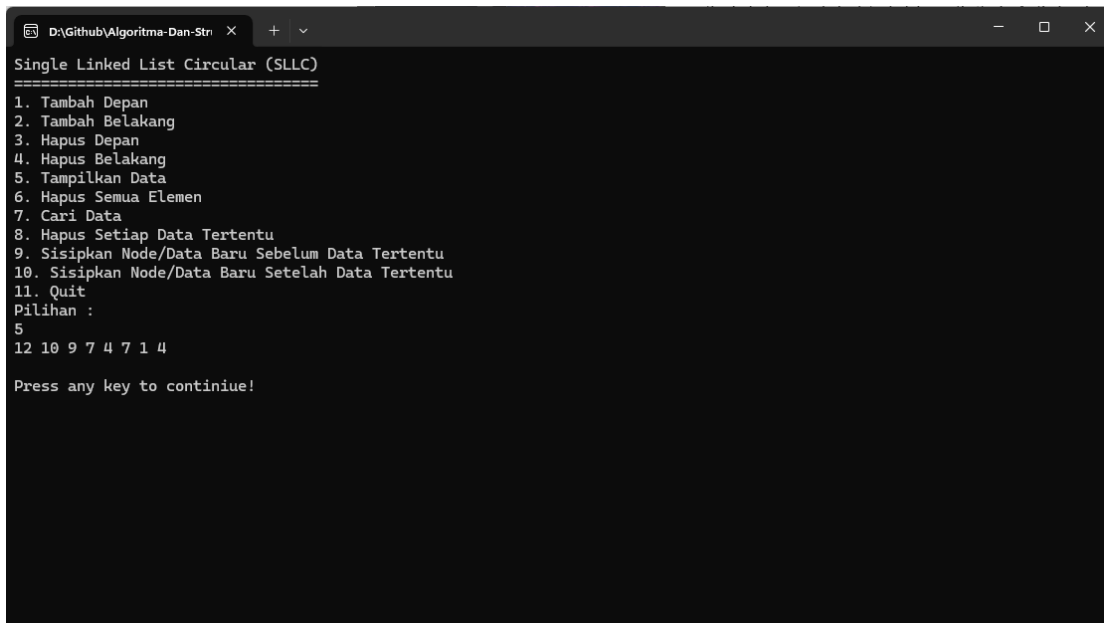
```
D:\Github\Algoritma-Dan-Str > Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan :
5
12 10 9 7 4 3 3 7 1 4
Press any key to continue!
```

Gambar 6. Screenshot Hasil Perintah Soal 6

## SOAL 7

Lakukan hapus setiap angka 3 dan kemudian lakukan tampilkan data lalu screenshot hasilnya !

### A. Screenshot



```
D:\Github\Algoritma-Dan-Stri >
Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan :
5
12 10 9 7 4 7 1 4

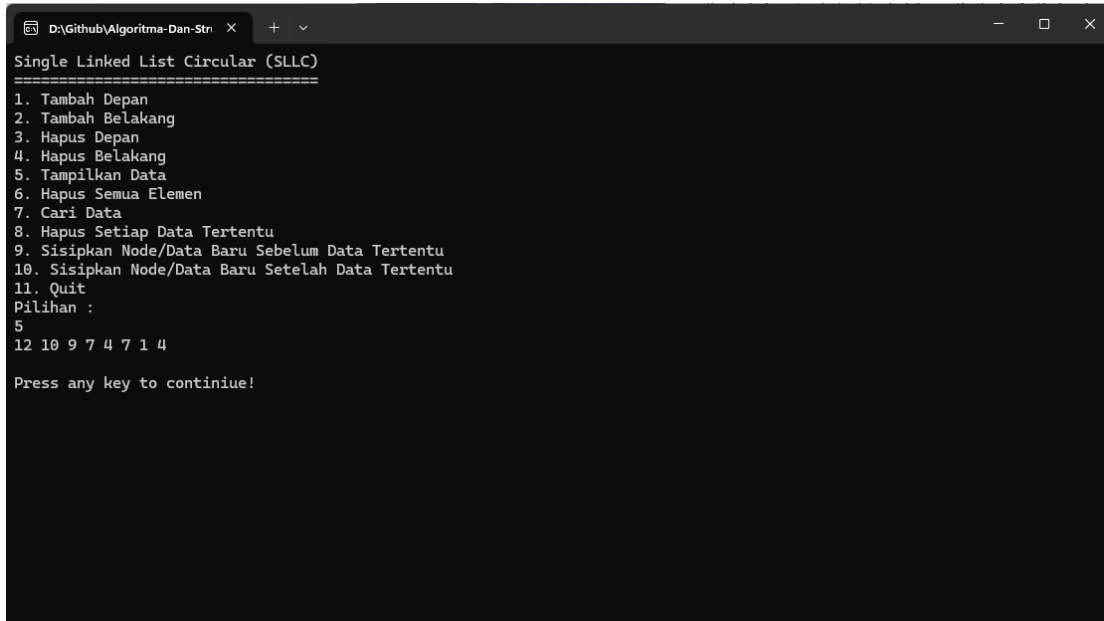
Press any key to continue!
```

Gambar 7. Screenshot Hasil Perintah Soal 7

## SOAL 8

Tampilkan data lalu jelaskan yang mana head dan yang mana tail.

### A. Screanshoot



```
D:\Github\Algoritma-Dan-Stri > Single Linked List Circular (SLLC)
=====
1. Tambah Depan
2. Tambah Belakang
3. Hapus Depan
4. Hapus Belakang
5. Tampilkan Data
6. Hapus Semua Elemen
7. Cari Data
8. Hapus Setiap Data Tertentu
9. Sisipkan Node/Data Baru Sebelum Data Tertentu
10. Sisipkan Node/Data Baru Setelah Data Tertentu
11. Quit
Pilihan : 5
12 10 9 7 4 7 1 4
Press any key to continue!
```

Gambar 8. Screanshoot Hasil Perintah Soal 8

### B. Pembahasan

Berdasarkan dari gambar 8 urutan data yang ditampilkan secara berurutan adalah “12, 10, 9, 7, 4, 7, 1, 4”. Maka data yang menjadi head adalah data yang berada paling kiri yaitu 12, karena head adalah node yang menunjuk ke data paling pertama yang diakses dalam list. Sedangkan data yang menjadi tail adalah data yang berada paling kanan yaitu 4, karena tail adalah node terakhir dalam list, yang pointer next-nya menunjuk kembali ke head.

## SOAL 9

Jika baris ke 103 dan 104 dihapus maka apa yang akan terjadi pada saat memasukkan data, dan jelaskan mengapa?

### A. Pembahasan

Di awal kita masih bisa menjalankan programnya, tetapi kita mencoba untuk menambahkan data menggunakan opsi menu satu dan dua maka program mungkin akan berhenti seketika. Ini disebabkan baris yang dihapus merupakan isi dari fungsi isEmpty(), yang mana fungsi ini pada normalnya akan mengembalikan value berupa 1 atau 0. Namun ketika pada kondisi di mana isi dari fungsi dari isEmpty() dihapus, nilai yang akan dikembalikan adalah garbage value atau nilai tidak pasti bisa saja 1 ataupun -11. Padahal nilai dari fungsi isEmpty() digunakan untuk logika pada fungsi penambahan data baru maupun fungsi lainnya. Program bisa salah mengira bahwa list sudah berisi, dan memproses else membuat struktur pointer jadi salah dan berpotensi crash.

## SOAL 10

Jelaskan apa itu variabel head dan tail pada slc!

### **B. Pembahasan**

Variabel head adalah pointer yang menunjuk ke simpul (node) pertama dalam struktur Single Linked List Circular (SLLC. Ketika pengguna ingin menampilkan seluruh isi list, maka penelusuran dimulai dari node yang ditunjuk oleh head. Karena sifat circular dari list ini, proses penelusuran akan berakhir kembali ke node head setelah semua node dilalui. Oleh karena itu, head memegang peran krusial sebagai acuan awal dan penanda batas dalam proses traversal list.

Variabel tail adalah pointer yang menunjuk ke simpul terakhir dalam SLLC. Tidak seperti linked list biasa yang node terakhirnya menunjuk ke NULL, pada SLLC node terakhir justru menunjuk kembali ke node pertama (yaitu head). Dengan demikian, tail->next akan selalu menunjuk ke head, yang menjadikan struktur data ini berbentuk melingkar atau circular. tail juga sering digunakan saat menambahkan node di bagian belakang list, karena node baru akan ditempatkan setelah node yang ditunjuk oleh tail, lalu tail diperbarui agar menunjuk ke node yang baru ditambahkan.

## **GITHUB**

<https://github.com/IndraSuryadilaga/Algoritma-Dan-Struktur-Data>