

ACTIVITY

NAMA : Ajay Alfredo Almani
NPM : 50420093
KELAS : 2IA16
TANGGAL : 23 Oktober 2021
MATERI : Linked List
PERTEMUAN : ke - 2
MATA PRAKTIKUM : Struktur Data

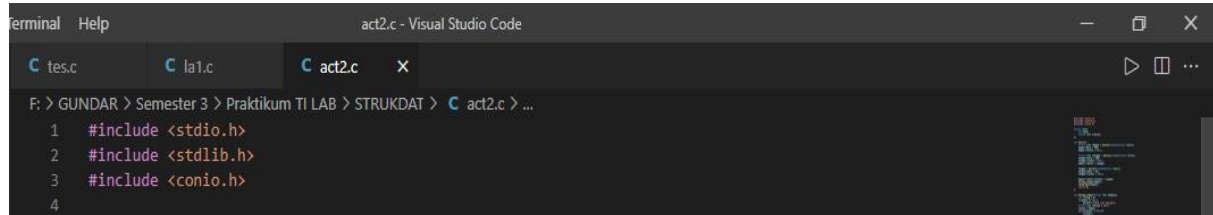


LISTING PROGRAM

```
terminal Help act2.c - Visual Studio Code
C tes.c C la1.c C act2.c X
F: > GUNDAR > Semester 3 > Praktikum TI LAB > STRUKDAT > C act2.c > cetak_data(node *)
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <conio.h>
4
5  struct node{
6      int data;
7      struct node *rantai;
8  };
9
10 int main(){
11     struct node *depan = malloc(sizeof(struct node));
12     depan->data = 504;
13     depan->rantai = NULL;
14
15     struct node *tengah = malloc(sizeof(struct node));
16     tengah->data = 200;
17     tengah->rantai = NULL;
18     depan->rantai = tengah;
19
20     tengah = malloc(sizeof(struct node));
21     tengah->data = 93;
22     tengah->rantai = NULL;
23
24     depan->rantai->rantai = tengah;
25     hitung_simpul(depan);
26     cetak_data(depan);
27     return 0;
28 }
29
30 int hitung_simpul(struct node *depan){
31     int hitung = 0;
32     if(depan == NULL)
33         printf("Linked list kosong");
```

```
terminal Help act2.c - Visual Studio Code
C tes.c C la1.c C act2.c X
F: > GUNDAR > Semester 3 > Praktikum TI LAB > STRUKDAT > C act2.c > ...
26     cetak_data(depan);
27     return 0;
28 }
29
30 int hitung_simpul(struct node *depan){
31     int hitung = 0;
32     if(depan == NULL)
33         printf("Linked list kosong");
34     struct node *tunjuk = NULL;
35     tunjuk = depan;
36     while(tunjuk != NULL){
37         hitung++;
38         tunjuk = tunjuk->rantai;
39     }
40     printf("\nJumlah Node pada linked adalah : %d", hitung);
41
42     getch();
43 }
44
45 int cetak_data(struct node *depan){
46     if(depan == NULL)
47         printf("Linked lis kosong");
48     struct node *tunjuk = NULL;
49     tunjuk = depan;
50     while(tunjuk != NULL){
51         printf("\nData pada linked list adalah %d", tunjuk->data);
52         tunjuk = tunjuk->rantai;
53     }
54
55     getch();
56
57 }
```

LOGIKA SINGKAT PROGRAM



```
terminal Help act2.c - Visual Studio Code
C tes.c C la1.c C act2.c X
F:\GUNDAR > Semester 3 > Praktikum TI LAB > STRUKDAT > C act2.c > ...
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <conio.h>
4
```

Pada program di atas adalah **#include** berfungsi untuk mengimpor fungsi-fungsi yang sudah didefinisikan pada header file. Header file adalah file yang berisi definisi fungsi yang sudah dibuat. Tujuannya agar bisa digunakan pada program C yang lainnya. **#include <stdio.h>** Perintah **#include** digunakan untuk memasukkan sebuah file khusus yang memungkinkan kita mengakses berbagai fitur tambahan dalam bahasa C. **stdlib.h** adalah file header yang berfungsi untuk operasi perbandingan dan operasi konversi. **#include <conio.h>**. Merupakan file header yang berfungsi untuk menampilkan hasil antarmuka kepada pengguna.



```
5 struct node{
6     int data;
7     struct node *rantai;
8 };
9
```

Program di atas berfungsi untuk pembentukan pointer utama. dan juga sebuah struktur dengan nama node lalu selanjutnya **int data**; yang sebagai Variable Interger lalu selanjutnya ada **struct node *rantai**; lalu selanjutnya pointer data.



```
10 int main(){
11     struct node *depan = malloc(sizeof(struct node));
12     depan->data = 504;
13     depan->rantai = NULL;
14
15     struct node *tengah = malloc(sizeof(struct node));
16     tengah->data = 200;
17     tengah->rantai = NULL;
18     depan->rantai = tengah;
19
20     tengah = malloc(sizeof(struct node));
21     tengah->data = 93;
22     tengah->rantai = NULL;
23
24     depan->rantai->rantai = tengah;
25     hitung_simpul(depan);
26     cetak_data(depan);
27     return 0;
28 }
```

Program pada gambar di atas yang pertama yaitu **int main()**, adalah fungsi (function) utama, fungsi yang akan dibaca oleh kompilator pertama kali secara otomatis, tidak akan ada function yang otomatis dibaca oleh kompilator kecuali fungsi utama **int main()**, fungsi utama bagaikan sebuah kepala dari sebuah program yang mengatur arah kompilator. **int** dalam fungsi utama adalah sebuah return type integer. Dalam peraturan C++ fungsi utama memang harus menggunakan return type integer. **struct node *depan = malloc(sizeof(struct node));** yang berfungsi sebagai Struct node depan adalah variable pertama. **depan->data = (504);** yang berisi variable data 504. **struct node *tengah = malloc(sizeof(struct node));** **tengah->data = 200;** yang berisi Struct node tengah adalah pointer / variable yg berisi data 202. Selanjutnya ada **tengah = malloc(sizeof(struct node));** **tengah->data = 93;** yang berisi Struct node tengah adalah pointer / variable yg berisi data 93. **depan->rantai->rantai = tengah;** **hitung_simpul(depan);** yang berisi metode fungsi simpul yg berhubungan simpul depan. Lalu di lanjut dengan **return 0** yang artinya fungsi telah selesai dengan baik. Fungsi mungkin saja mengembalikan nilai lain dalam bentuk integer.

```

29
30 int hitung_simpul(struct node *depan){
31     int hitung = 0;
32     if(depan == NULL)
33         printf("Linked list kosong");
34     struct node *tunjuk = NULL;
35     tunjuk = depan;
36     while(tunjuk != NULL){
37         hitung++;
38         tunjuk = tunjuk->rantai;
39     }
40     printf("\nJumlah Node pada linked adalah : %d", hitung);
41
42     getch();
43 }

```

Program pada gambar di atas yang pertama yaitu int **hitung_simpul(struct node *depan){** yang berisi metode yg berfungsi untuk menghitung simpul yg ada. Lalu melakukan digunakan untuk mengecek kondisi sebuah pernyataan dan menghasilkan output sesuai kondisi dari pernyataan tersebut. Lalu melakukan looping pada program di int main sebelum nya, sebuah output jika kondisi dari pernyataannya masih benar. **getch** sebenarnya merupakan fungsi yang digunakan untuk membaca input dari keyboard. getch() ini merupakan kepanjangan dari Get Character.

```

44
45 int cetak_data(struct node *depan){
46     if(depan == NULL)
47         printf("Linked list kosong");
48     struct node *tunjuk = NULL;
49     tunjuk = depan;
50     while(tunjuk != NULL){
51         printf("\nData pada linked list adalah %d", tunjuk->data);
52         tunjuk = tunjuk->rantai;
53     }
54
55     getch();
56
57 }

```

Program pada gambar di atas yang pertama yaitu int **cetak_data(struct node *depan){** yang berisi metode yg berfungsi menyambung dan mencetak data dan depan. Lalu melakukan digunakan untuk mengecek kondisi sebuah pernyataan dan menghasilkan output sesuai kondisi dari pernyataan tersebut. Lalu melakukan looping pada program di int main sebelum nya, sebuah output jika kondisi dari pernyataannya masih benar. **getch** sebenarnya merupakan fungsi yang digunakan untuk membaca input dari keyboard. getch() ini merupakan kepanjangan dari Get Character.

OUTPUT PROGRAM

The image shows a Visual Studio Code editor with a C program for a linked list. The code is as follows:

```

33     printf("Linked list kosong");
34     struct node *tunjuk = NULL;
35     tunjuk = depan;
36     while(tunjuk != NULL){
37         hitung++;
38         tunjuk = tunjuk->rantai;
39     }
40     printf("\nJumlah Node pada linked adalah : %d", hitung);
41
42     getch();
43 }
44
45 int cetak_data(struct node *depan){
46     if(depan == NULL)
47         return 0;
48     printf("%d\n", depan->data);
49     cetak_data(depan->rantai);
50 }

```

The terminal output shows the execution results:

```

F:\> GUNDAR > Semester 3 > Praktikum TI LAB > STRUKDAT > C act2.c > cetak_data(node *)
Jumlah Node pada linked adalah : 3
Data pada linked list adalah 504
Data pada linked list adalah 200
Data pada linked list adalah 93
PS F:\GUNDAR\Semester 3\Praktikum TI LAB\STRUKDAT>

```

The file explorer on the right shows the project structure:

- Praktikum TI LAB
 - STRUKDAT
 - act2.c
 - cetak_data.c
 - cetak_data.h
 - main.c
 - main.h
 - node.c
 - node.h
 - test.c
 - test.h