

Nama : La Ode Muhammad Gazali
NIM : 222212696
Kelas : 2KS2

MODUL 4 STRKTUR DATA

1. Buatlah fungsi untuk menghitung jumlah node dalam sebuah linked list! (looping sama seperti pada saat menampilkan nilai dari linked list).

Program

```
#include <stdio.h>
#include <stdlib.h>

//structure untuk node linked list
struct node {
    int data;
    struct node *next;
};
typedef struct node *ptrnode;

//Fungsi untuk membuat node baru
ptrnode createnode(int nilai) {
    ptrnode p;
    p = (ptrnode)malloc(sizeof(struct node));
    p->data = nilai;
    p->next = NULL;
    return (p);
}

//Fungsi untuk menambahkan nilai pada node baru
ptrnode insert_head(ptrnode head, int nilai) {
    ptrnode newnode = createnode(nilai);
    newnode->next = head;
    head = newnode;
    return (head);
}

Fungsi untuk menghitung jumlah node
void count(ptrnode head) {
    int count = 0;
    struct node *p;
    p = head;
    while (p != NULL) {
        p = p->next;
        count++;
    }
}
```

```

    }
    printf("\nJumlah node adalah : %d\n", count);
}

int main() {
    int data1,n;
    ptrnode head = NULL;

    printf("Masukan jumlah data : ");scanf("%d",&n);

    //looping untuk menginput data pada node secara dinamis
    for (int i = 0; i < n; i++) {
        printf("Masukkan data ke-%d = ", i + 1);
        scanf("%d", &data1);
        head = insert_head(head, data1);
    }

    //memanggil fungsi untuk menghitung jumlah node
    count(head);

    return 0;
}

```

Output:

```

Masukan jumlah data : 4
Masukkan data ke-1 = 12
Masukkan data ke-2 = 43
Masukkan data ke-3 = 22
Masukkan data ke-4 = 45

Jumlah node adalah : 4

```

2. Buatlah fungsi untuk membalik nilai dari head ke tail!

Contoh: 5->4->3->2->1 menjadi 1->2->3->4->5

Hint:

- hanya nilai saja, memory address (pointer node) tetap sama
- buat temporary pointer node sebagai bantuan: prev, current, next dan loop dari head ke tail

Program:

```

#include <stdio.h>
#include <stdlib.h>
//structure untuk node linked list

```

```

struct node {
    int data;
    struct node *next;
};
typedef struct node *ptrnode;

//Fungsi untuk membuat node baru

ptrnode createnode(int nilai) {
    ptrnode p;
    p = (ptrnode)malloc(sizeof(struct node));
    p->data = nilai;
    p->next = NULL;
    return (p);
}

//Fungsi untuk menambahkan nilai pada node baru

ptrnode insert_head(ptrnode head, int nilai) {
    ptrnode newnode = createnode(nilai);
    newnode->next = head;
    head = newnode;
    return (head);
}

//Fungsi untuk menampilkan data
void tampilnilai(ptrnode head){
    int i=1;
    ptrnode n = head;
    while (n != NULL){
        printf("node ke %d : %d\n",i, n->data);
        n = n->next;
        i++;
    };
    printf("\n");
}

//Fungsi untuk merubah tampilah data menjadi terbalik
ptrnode tampilnilaiterbalik(ptrnode head){
    struct node *current, *prev, *next;
    current = head;
    prev=NULL;
    while (current != NULL){
        next = current ->next;
        current->next =prev;
        prev = current;
        current = next;
    }
}

```

```

    }
    head = prev;
}

int main() {
    int data1,n;
    ptrnode head = NULL;

    printf("Masukan jumlah data : ");scanf("%d",&n);

    for (int i = 0; i < n; i++) {
        printf("Masukkan data ke-%d = ", i + 1);
        scanf("%d", &data1);
        head = insert_head(head, data1);
    }
    printf("\nNilai sebelum di reverse\n");
    tampilnilai(head);

    printf("Nilai setelah di reverse\n");
    head=tampilnilaiterbalik(head);
    tampilnilai(head);

    return 0;
}

```

Output:

```

Masukan jumlah data : 5
Masukkan data ke-1 = 1
Masukkan data ke-2 = 2
Masukkan data ke-3 = 3
Masukkan data ke-4 = 4
Masukkan data ke-5 = 5

Nilai sebelum di reverse
node ke 1 : 5
node ke 2 : 4
node ke 3 : 3
node ke 4 : 2
node ke 5 : 1

Nilai setelah di reverse
node ke 1 : 1
node ke 2 : 2
node ke 3 : 3
node ke 4 : 4
node ke 5 : 5

```

3. Buat program untuk menyimpan data students berisi int nim, char nama[50] secara dinamis!

Program:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Struktur untuk merepresentasikan data mahasiswa

struct mahasiswa {
    int nim;
    char nama[50];
    struct mahasiswa* next;
};

typedef struct mahasiswa mhs;

// Fungsi untuk membuat node baru dengan data mahasiswa

mhs* createmhs(int nim, const char* nama) {
    mhs* mahasiswaBaru = (mhs*)malloc(sizeof(mhs));

    mahasiswaBaru->nim = nim;
    strncpy(mahasiswaBaru->nama, nama, sizeof(mahasiswaBaru->nama));
    mahasiswaBaru->next = NULL;

    return mahasiswaBaru;
}

// Fungsi untuk menambahkan node mahasiswa ke linked list

mhs* addmahasiswa(mhs* head, int nim, const char* nama) {
    mhs* mahasiswaBaru = createmhs(nim, nama);
    mahasiswaBaru->next = head;
    head=mahasiswaBaru;
    return (head);
}

// Fungsi untuk menampilkan daftar mahasiswa dalam linked list

void tampilkanmahasiswa(mhs* head) {
    mhs* n = head;

    printf("Daftar Mahasiswa:\n");
    while (n != NULL) {
```

```

        printf("NIM: %d, Nama: %s\n", n->nim, n->nama);
        n = n->next;
    }
    printf("\n");
}

int main() {
    mhs* head = NULL;
    int n, nim;
    char nama[50];
    printf("Masukan jumlah data mahasiswa : "); scanf("%d", &n);

    for (int i = 0; i < n; i++) {
        printf("\nMahasiswa ke-%d\n", i+1);
        printf("Nama   = "); scanf("%s", &nama);
        printf("NIM    = "); scanf("%d", &nim);
        head = addmahasiswa(head, nim, nama);
    }
    //memanggil fungsi untuk menampilkan data mahasiswa
    tampilmahasiswa(head);

    return 0;
}

```

Output:

```

Masukan jumlah data mahasiswa : 3

Mahasiswa ke-1
Nama   = Gaza
NIM    = 11

Mahasiswa ke-2
Nama   = Tiara
NIM    = 12

Mahasiswa ke-3
Nama   = Komang
NIM    = 13
Daftar Mahasiswa:
NIM: 13, Nama: Komang
NIM: 12, Nama: Tiara
NIM: 11, Nama: Gaza

```