**TUGAS PRAKTIKUM ALGORITMA & STRUKTUR DATA**

**Jilid 6**



**Oleh :**

**Nama : Rosi Arif Mulyadi**

**NRP : 3121522021**

**Prodi : D3 Teknik Informatika PENS PSDKU Sumenep**

**Kelas : 1 ITA D3 Sumenep**

**Dosen :**

**LUSIANA AGUSTIEN M.Kom**

**POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**

**MODUL 3.1**

**LINKED LIST (Reverse Linked List)**

**B. Kegiatan Praktikum**

1. Implementasikan dan tentukan output percobaan yang ada dalam modul praktikum ini dan lakukan analisa pada tiap fungsi yang dibuat.

Jawab :

Listing Program :

#include <stdio.h>

#include <stdlib.h>

struct node

{

int val;

struct node \*next;

};

void print\_list(struct node \*head)

{

printf("H->");

while(head)

{

printf("%d->", head->val);

head = head->next;

}

printf("|||\n");

}

void insert\_front(struct node \*\*head, int value)

{

struct node \* new\_node = NULL;

new\_node = (struct node \*)malloc(sizeof(struct node));

if(new\_node == NULL)

{

printf("Failed to insert element. Out of memory");

}

new\_node->val = value;

new\_node->next = \*head;

\*head = new\_node;

}

void reverse\_linked\_list(struct node \*\*head)

{

struct node \* new\_head = NULL;

struct node \* tmp = NULL;

while(\*head)

{

tmp = \*head;

\*head = (\*head)->next;

tmp->next = new\_head;

new\_head = tmp;

}

\*head = new\_head;

}

void main()

{

int count = 0, i, val;

struct node \* head = NULL;

printf("Enter number of elements : ");

scanf("%d", &count);

for(i=0; i<count; i++)

{

printf("Enter %d th element : ", i);

scanf("%d", &val);

insert\_front(&head, val);

}

printf("Initial List : ");

print\_list(head);

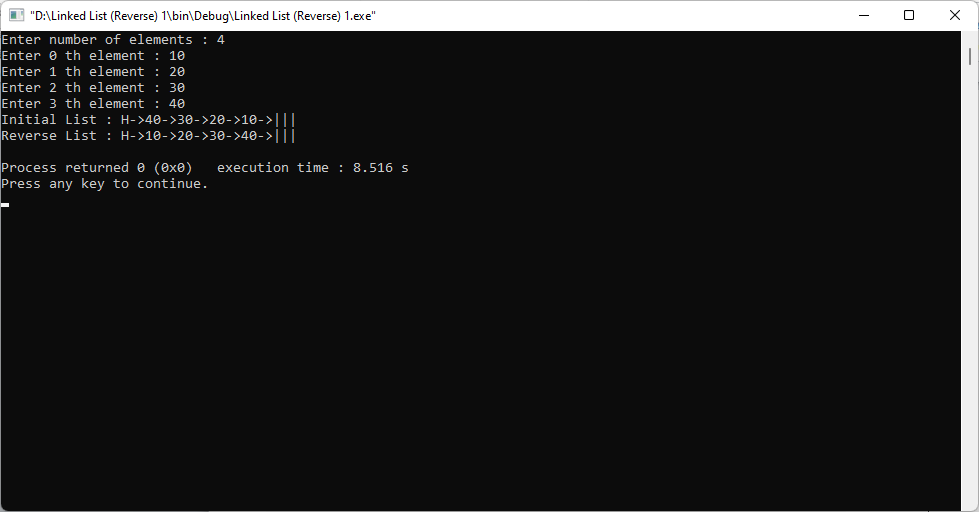
reverse\_linked\_list(&head);

printf("Reverse List : ");

print\_list(head);

}

Output :



Analisa :

* Reverse digunakan untuk mengubah dan mengurutkan value pada percobaan.
* Dapat mengurutkan value sesuai dengan element yang dimasukkan / diinputkan.

**C. Tugas Praktikum**

1. Implementasikan algoritma reverse pada program percobaan yang ada pada modul 2.

Jawab :

Algoritma :

1. Keluarkan simpul pertama dari daftar, satu pointer baru akan menunjuk ke node pertama dan head akan berpindah ke node baru yang di inoutkan pada list baru

2. Ulangi langkah sebelumnya sampai semua node pindah ke list baru.

3. Arahkan head ke node pertama dari list baru tersebut.

Listing Program (Reverse) :

#include <stdio.h>

#include <stdlib.h>

struct node

{

char nama[50];

int val;

struct node \*next;

};

void print\_list(struct node \*head)

{

printf("H->");

while(head)

{

printf("%s %i ->",head->nama, head->val);

head = head->next;

}

printf("|||\n");

}

void insert\_front(struct node \*\*head, char nama[],int value)

{

struct node \* new\_node = NULL;

new\_node = (struct node \*)malloc(sizeof(struct node));

if(new\_node == NULL)

{

printf("Failed to insert element. Out of memory");

}

strcpy(new\_node->nama,nama);

new\_node->val = value;

new\_node->next = \*head;

\*head = new\_node;

}

void reverse\_linked\_list(struct node \*\*head, char nama[],int value)

{

struct node \* new\_head = NULL;

struct node \* tmp = NULL;

while(\*head)

{

tmp = \*head;

\*head = (\*head)->next;

tmp->next = new\_head;

new\_head = tmp;

}

\*head = new\_head;

}

void main()

{

int count = 0, i, val;

char nama[10];

struct node \* head = NULL;

printf("Enter number of elements : ");

scanf("%d", &count);

for(i=0; i<count; i++)

{

printf("Enter %d th element : ", i);

scanf("%s", &nama);

fflush(stdin);

printf("Enter %d th element : ", i);

scanf("%d", &val);

fflush(stdin);

insert\_front(&head, nama, val);

}

printf("Initial List : ");

print\_list(head);

reverse\_linked\_list(&head, nama, val);

printf("Reverse List : ");

print\_list(head);

}

Output :

