NAMA : M AL FAIZ PUTRA JALASENANDRA

NIM : 21091397072

PRAKTIKUM 6

1. #include <stdio.h>

#include <stdlib.h>

struct link{

int data;

struct link \*next;

};

struct infoku{

struct link \*awal=0;

struct link \*akhir=0;

}info;

link \*hapus\_data(infoku \*info){

link \*selanjutnya=0;

if(info->awal!=0){

selanjutnya=info->awal->next;

free(info->awal);

info->awal=selanjutnya;

}

if(selanjutnya!=0){

info->akhir=0;

}

return selanjutnya;

}

link \*tambah\_data(infoku \*info){

link \*penunjuk=0;

penunjuk=(link\*)malloc(sizeof(link));

penunjuk->next=0;

if(info->awal==0){

info->awal=penunjuk;

}else info->akhir->next=penunjuk;

info->akhir=penunjuk;

return penunjuk;

}

int main(){

int i=0;

int n=10;

struct link \*penunjuk;

info.awal=0;

info.akhir=0;

printf("Masukkan jumlah data!\n");

scanf("%d", &n);//masukkan jumlah data yang diinginkan

//buat linked list

while(i<n){

penunjuk=tambah\_data(&info);

penunjuk->data=i\*2;

i++;

}

//tampilkan isi struct dalam linked list

printf("\nAntrian : ");

penunjuk=info.awal;

while(penunjuk!=0){

printf("%d ", penunjuk->data);

penunjuk=penunjuk->next;

}

hapus\_data(&info);

//tampilkan isi struct dalam linked list setelah data pertama dihapus

printf("\n\nSetelah data pertama dalam antrian dihapus?\n");

printf("Antrian : ");

penunjuk=info.awal;

while(penunjuk!=0){

printf("%d ", penunjuk->data);

penunjuk=penunjuk->next;

}

printf("\n");

//bebaskan semua memori yang sudah digunakan

penunjuk=info.awal;

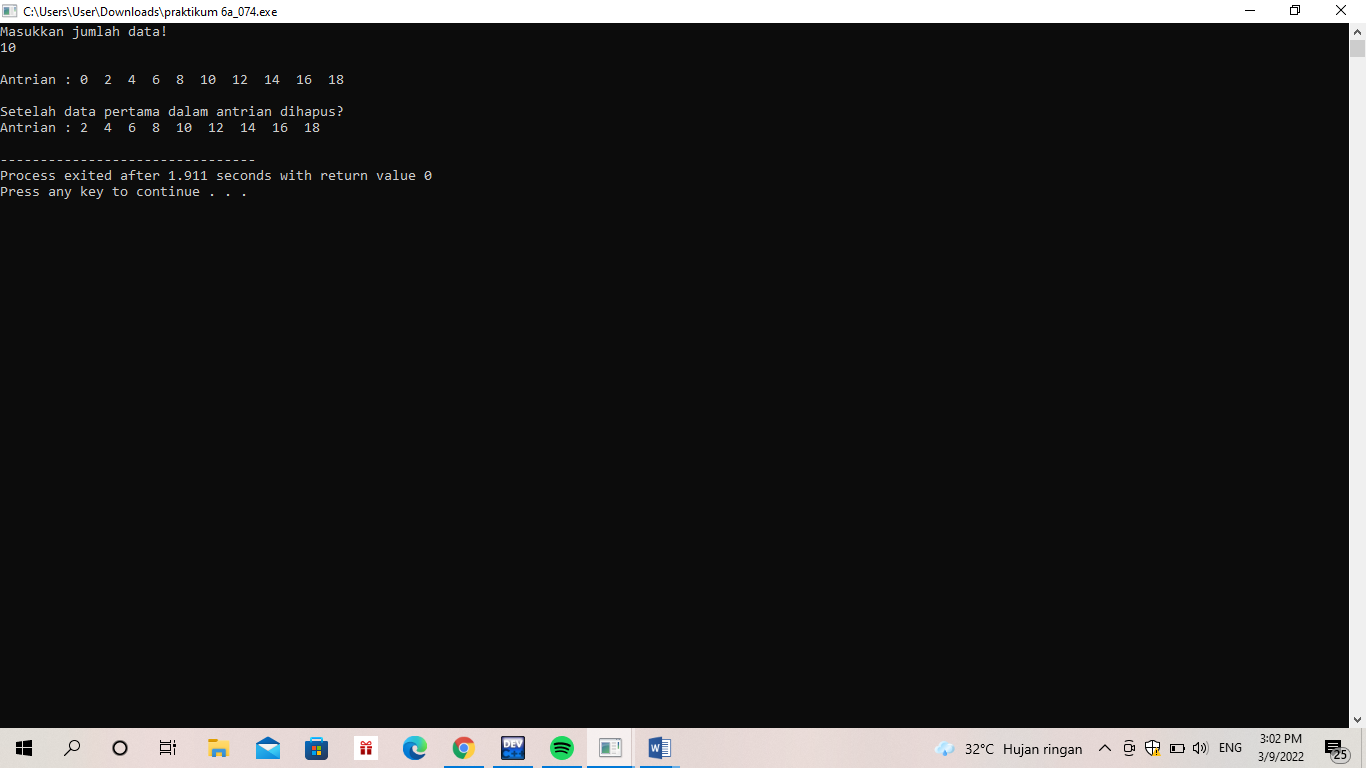
while(penunjuk!=0){

penunjuk=hapus\_data(&info);

}

return 0;

}



2. #include <stdio.h>

#include <stdlib.h>

struct link{

int data;

struct link \*prev;

};

struct infoku{

struct link \*atas=0;

struct link \*bawah=0;

}info;

link \*hapus\_data(infoku \*info){

link \*sebelumnya=0;

if(info->atas!=0){

sebelumnya=info->atas->prev;

free(info->atas);

info->atas=sebelumnya;

}

if(sebelumnya!=0){

info->bawah=0;

}

return sebelumnya;

}

link \*tambah\_data(infoku \*info){

link \*penunjuk=0;

penunjuk=(link\*)malloc(sizeof(link));

if(info->bawah==0){

penunjuk->prev=0;

info->bawah=penunjuk;

}else penunjuk->prev=info->atas;

info->atas=penunjuk;

return penunjuk;

}

int main(){

int i=0;

int n=10;

struct link \*penunjuk;

info.atas=0;

info.bawah=0;

printf("Masukkan jumlah data!\n");

scanf("%d", &n);//masukkan jumlah data yang diinginkan

//buat linked list

while(i<n){

penunjuk=tambah\_data(&info);

penunjuk->data=i\*2;

i++;

}

//tampilkan isi struct dalam linked list

i=0;

penunjuk=info.atas;

while(penunjuk!=0){

i++;

printf("Tumpukan ke-%d : %d\n", i, penunjuk->data);

penunjuk=penunjuk->prev;

}

hapus\_data(&info);

//tampilkan isi struct dalam linked list setelah data pertama dihapus

printf("\nSetelah data pertama dalam tumpukan dihapus?\n");

i=0;

penunjuk=info.atas;

while(penunjuk!=0){

i++;

printf("Tumpukan ke-%d : %d\n", i, penunjuk->data);

penunjuk=penunjuk->prev;

}

//bebaskan semua memori yang sudah digunakan

penunjuk=info.atas;

while(penunjuk!=0){

penunjuk=hapus\_data(&info);

}

return 0;

}

