#include <stdio.h>

#include <stdlib.h>

struct Nod {

int info;

struct Nod\* succ;

};

struct Nod\* init(const int info) {

struct Nod\* nod\_nou = (struct Nod\*)malloc(sizeof(struct Nod));

nod\_nou->info = info;

nod\_nou->succ = NULL;

return nod\_nou;

}

struct Nod\* inserare\_inceput(struct Nod\* front, const int info) {

if (front == NULL) {

front = init(info);

} else {

struct Nod\* nod\_nou = init(info);

nod\_nou->succ = front;

front = nod\_nou;

}

return front;

}

struct Nod\* inserare\_sfarsit(struct Nod\* front, const int info) {

if (front == NULL) {

front = init(info);

} else {

struct Nod\* nod\_nou = init(info);

struct Nod\* temp = front;

while (temp->succ != NULL) {

temp = temp->succ;

}

temp->succ = nod\_nou;

}

return front;

}

struct Nod\* inserare\_interior(struct Nod\* front, const int info){

if(front == NULL) {

front = init(info);

} else {

struct Nod\* nod\_nou = init(info);

struct Nod\* nod\_urmator=front->succ;

front->succ=nod\_nou;

nod\_nou->succ=nod\_urmator;

}

return front;

}

struct Nod\* stergere\_inceput(struct Nod\* front) {

if (front != NULL) {

front = front->succ;

}

return front;

}

struct Nod\* stergere\_sfarsit(struct Nod\* front) {

if (front != NULL) {

struct Nod\* tmp=front;

struct Nod\* tmp1=front;

while(tmp->succ!=NULL){

tmp=tmp->succ;

}

while(tmp1->succ!=tmp){

tmp1=tmp1->succ;

}

tmp1->succ=NULL;

}

return front;

}

struct Nod\* stergere\_interior(struct Nod\* front){

if(front!=NULL){

struct Nod\* tmp=front->succ;

front->succ=tmp->succ;

}

return front;

}

struct Nod\* distrugere(struct Nod\* front) {

while (front != NULL) {

front = stergere\_inceput(front);

}

return front;

}

int cautare(struct Nod\* front, int cheie, int\* i){

struct Nod\* tmp=front;

while(tmp!=NULL){

if(cheie==tmp->info){

return \*i;

}

else{

tmp=tmp->succ;

(\*i)++;

}

}

}

struct Nod\* actualizare(struct Nod\* front, int pozitie){

if(front!=NULL){

int i=0;

struct Nod\* tmp=front;

while(i!=pozitie){

tmp=tmp->succ;

i++;

}

printf("Actualizeaza valoarea\n");

scanf("%d", &tmp->info);

}

return front;

}

int main() {

struct Nod\* front = NULL;

front = inserare\_inceput(front, 3);

front = inserare\_sfarsit(front, 5);

front = inserare\_interior(front, 6);

printf("%d\n", front->info);

printf("%d\n", front->succ->info);

printf("%d\n", front->succ->succ->info);

//front=stergere\_inceput(front);

// front=stergere\_sfarsit(front);

//front=stergere\_interior(front);

/\* printf("%d\n", front->info);

printf("%d\n", front->succ->info);

printf("%d\n", front->succ->succ->info);\*/

// front = distrugere(front);

int cheie=5;

int i=0;

printf("Pozitia este:\n%d\n", cautare(front, cheie, &i));

printf("Alege pozitia:\n");

int pozitie;

scanf("%d", &pozitie);

front=actualizare(front, pozitie);

printf("Lista actualizata\n");

printf("%d\n", front->info);

printf("%d\n", front->succ->info);

printf("%d\n", front->succ->succ->info);

struct Nod\* tmp=front;

struct Nod\* tmp2;

while(tmp!=NULL){

tmp2=tmp->succ;

while(tmp2!=NULL){

if(tmp->info>tmp2->info){

int temp=tmp->info;

tmp->info=tmp2->info;

tmp2->info=temp;

}

tmp2=tmp2->succ;

}

tmp=tmp->succ;

}

printf("Lista ordonata\n");

printf("%d\n", front->info);

printf("%d\n", front->succ->info);

printf("%d\n", front->succ->succ->info);

printf("Maximul:\n");

tmp=front;

int max=front->info;

while(tmp!=NULL){

if(tmp->info>max){

max=tmp->info;

}

tmp=tmp->succ;

}

printf("%d\n", max);

printf("Minimul:\n");

tmp=front;

int min=front->info;

while(tmp!=NULL){

if(tmp->info<min){

min=tmp->info;

}

tmp=tmp->succ;

}

printf("%d\n", min);

return 0;

}