**מחסנית תשובה לתרגיל 1 - stack**

#include<stdio.h>

const int N = 10;

struct stack {

int arr[N];

int size;

};

void stackInit(struct stack& koko) {

koko.size = 0;

}

int empty(struct stack& koko) {

if (koko.size == 0)

return 1;

else

return 0;

}

void push(struct stack& koko, int x) {

if (koko.size == N) {

printf("stack full!");

}

if (koko.size < N && koko.size > -1) {

koko.arr[koko.size] = x;

koko.size++;

}

}

int stacktop(struct stack& koko) {

if (empty(koko) == 0)

return koko.arr[koko.size - 1];

}

int pop(struct stack &koko) {

if (empty(koko) == 0) {

int x = koko.arr[koko.size - 1];

koko.size = koko.size - 1;

return x;

}

}

void print(struct stack& koko) {

for (int i = 0; i < koko.size; i++) {

printf("%d,", koko.arr[i]);

}

}

void Delete(struct stack& koko, int x) {

struct stack s2;

stackInit(s2);

int counter = 0;

if (empty(koko) == 0)

{

while (stacktop(koko) != x) {

push(s2, pop(koko));

counter++;

}

pop(koko);

while (counter > 0) {

push(koko, pop(s2));

counter--;

}

}

else

printf("the stack is empty!");

}

int search(struct stack &koko, int x) {

struct stack s2;

stackInit(s2);

int counter = 0;

if(empty(koko) == 0) {

while(stacktop(koko) != x) {

push(s2, pop(koko));

counter++;

}

}

int j = counter;

while (counter > 0) {

push(koko, pop(s2));

counter--;

}

return j + 1;

}

void main() {

struct stack s;

stackInit(s);

push(s, 1);

push(s, -9);

push(s, 2);

push(s, 5);

push(s, 7);

printf("the first number we pulled out is: %d\n",pop(s));

printf("the second number we pulled out is: %d\n",pop(s));

push(s, 8);

push(s, 6);

push(s, 4);

print(s);

}