Беларускі дзяржаўны тэхналагічны ўніверсітэт

Факультэт інфармацыйных тэхналогій

Кафедра праграмнай інжэнерыі

Лабараторная работа №7

Па дысцыпліне «Асновы алгарытмізацыі і праграмавання»

На тэму «Полустатычныя структуры: стэкі»

Выканала:

Студэнтка 1 курса 6 группы

Жучкевіч Кацярына Сяргееўна

Выкладчык: асс. Андронава М.В.

Мінск, 2024

HEADER.h

#pragma once

#include <fstream>

#include <Windows.h>

#include <iostream>

using namespace std;

struct Stack

{

int data;

Stack\* next;

};

void output(Stack\*& stk);

void insert(Stack\*& stk);

void inputFile(Stack\*& stk);

void outputFile(Stack\*& stk);

void clear(Stack\*& stk);

void task(Stack\*& stk);



#include "HEADER.h"

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

Stack\* first = new Stack;//выдзяленне памяці

first = NULL;

int choice;

cout << "0 - выхад\n1 - дадаць элемент у стэк\n2 - выдаліць элемент з стэка\n"

<< "3 - увод элементаў у файл\n4 - вывад з файла\n5 - вывад\n6 - пошук па ўмовах задання\n";

do//цыкл, пакуль не увесці 0

{

cin >> choice;

switch (choice)

{

case 0: exit(1); break;

case 1: insert(first); break;

case 2: clear(first); break;

case 3: inputFile(first); break;

case 4: outputFile(first); break;

case 5: output(first); break;

case 6: task(first); break;

}

} while (choice != 0);

}

void insert(Stack \* &stk) //функцыя для увода элементаў

{

int elem;

int n;

cout << "Колькаць элементаў: "; cin >> n;

cout << "\n";

for (int i = 0; i < n; i++)

{

cin >> elem;

Stack\* newElem = new Stack;//памяць для элементаў

newElem->data = elem;//запісваем элемент ў дату

newElem->next = stk;//пераходзім на наступны элемент

stk = newElem;

}

cout << "Во!\n";

}

void inputFile(Stack \* &stk)//функцыя для ўвода ў файл

{

ofstream fin("file.txt");

if (!fin.is\_open()) {//праглядаем прысутнасць файла

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (stk != NULL)

{

fin << stk->data << "\n";

stk = stk->next;

}

fin.close();

}

cout << "Во!\n";

}

void outputFile(Stack \* &stk)//функцыя для вывада з файла

{

ifstream fout("file.txt");

int buf;

if (!fout.is\_open()) {

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (fout >> buf)

{

Stack\* newElem = new Stack;

newElem->data = buf;

newElem->next = stk;

stk = newElem;

cout << buf << " ";

}

fout.close();//зачыняем файл

}

cout << "Во!\n";

}

void clear(Stack \* &stk)//функцыя для ачысткі

{

while (stk != NULL)//пакуль не канчатак

{

Stack\* del = stk;

if (stk)

{

stk = stk->next;//новая вяршына

}

delete del;

}

cout << "\nЦяпер тут пуста!!\n";

}

void output(Stack \* &stk)//функцыя для вывада стэка

{

Stack\* output = stk;

while (output != NULL)

{

cout << output->data << " ";//вывад

output = output->next;//пераход на наступны

}

cout << "\n";

cout << "Во!\n";

}

void task(Stack \* &stk)//функцыя з асноўным заданнем

{

int count = 0;

Stack\* main = stk;

while (main != NULL)//пакуль не канчатак

{

Stack\* find = main->next;//адрас наступнага элементу

while (find != NULL)//пакуль не канчатак

{

if (main->data == find->data)//і калі роўныя суседзі

{

count++;//лічым

break;

}

find = find->next;//пераходзім на наступны

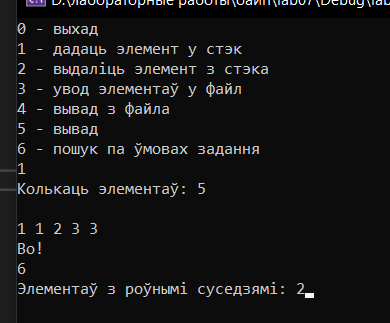
}

main = main->next;//пераходзім на наступны

}

cout << "Элементаў з роўнымі суседзямі: " << count;//вывад

}

****

**Дадатковыя варыянты:**

****

#include "HEADER.h"

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

Stack\* first = new Stack;//выдзяленне памяці

first = NULL;

int choice;

cout << "0 - выхад\n1 - дадаць элемент у стэк\n2 - выдаліць элемент з стэка\n"

<< "3 - увод элементаў у файл\n4 - вывад з файла\n5 - вывад\n6 - пошук па ўмовах задання\n";

do//цыкл, пакуль не увесці 0

{

cin >> choice;

switch (choice)

{

case 0: exit(1); break;

case 1: insert(first); break;

case 2: clear(first); break;

case 3: inputFile(first); break;

case 4: outputFile(first); break;

case 5: output(first); break;

case 6: task(first); break;

}

} while (choice != 0);

}

void insert(Stack \* &stk) //функцыя для увода элементаў

{

int elem;

int n;

cout << "Колькаць элементаў: "; cin >> n;

cout << "\n";

for (int i = 0; i < n; i++)

{

cin >> elem;

Stack\* newElem = new Stack;//памяць для элементаў

newElem->data = elem;//запісваем элемент ў дату

newElem->next = stk;//пераходзім на наступны элемент

stk = newElem;

}

cout << "Во!\n";

}

void inputFile(Stack \* &stk)//функцыя для ўвода ў файл

{

ofstream fin("file.txt");

if (!fin.is\_open()) {//праглядаем прысутнасць файла

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (stk != NULL)

{

fin << stk->data << "\n";

stk = stk->next;

}

fin.close();

}

cout << "Во!\n";

}

void outputFile(Stack \* &stk)//функцыя для вывада з файла

{

ifstream fout("file.txt");

int buf;

if (!fout.is\_open()) {

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (fout >> buf)

{

Stack\* newElem = new Stack;

newElem->data = buf;

newElem->next = stk;

stk = newElem;

cout << buf << " ";

}

fout.close();//зачыняем файл

}

cout << "Во!\n";

}

void clear(Stack \* &stk)//функцыя для ачысткі

{

while (stk != NULL)//пакуль не канчатак

{

Stack\* del = stk;

if (stk)

{

stk = stk->next;//новая вяршына

}

delete del;

}

cout << "\nЦяпер тут пуста!!\n";

}

void output(Stack \* &stk)//функцыя для вывада стэка

{

Stack\* output = stk;

while (output != NULL)

{

cout << output->data << " ";//вывад

output = output->next;//пераход на наступны

}

cout << "\n";

cout << "Во!\n";

}

void task(Stack\*& stk)//функцыя з асноўным заданнем

{

Stack\* main = stk;

int flag = 0;

while (main != NULL)

{

Stack\* find = main->next;

while (find != NULL) {

if (main->data == find->data)//калі супадае

{

flag = 1;

break;

}

find = find->next;

}

if (flag == 1)//выхад з цыкла, калі ёсць такі элемент раней

{

cout << "\nТакі элемент ёсць!!";

break;

}

else {

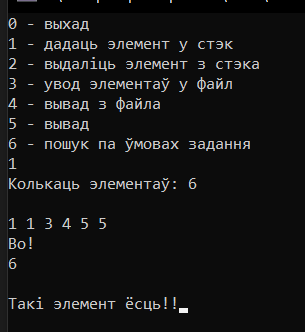
cout << "Такога элемента няма((";

}

main = main->next;//пераход на наступны

}

}

****

****

#include "HEADER.h"

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

Stack\* first = new Stack;//выдзяленне памяці

first = NULL;

int choice;

cout << "0 - выхад\n1 - дадаць элемент у стэк\n2 - выдаліць элемент з стэка\n"

<< "3 - увод элементаў у файл\n4 - вывад з файла\n5 - вывад\n6 - пошук па ўмовах задання\n";

do//цыкл, пакуль не увесці 0

{

cin >> choice;

switch (choice)

{

case 0: exit(1); break;

case 1: insert(first); break;

case 2: clear(first); break;

case 3: inputFile(first); break;

case 4: outputFile(first); break;

case 5: output(first); break;

case 6: task(first); break;

}

} while (choice != 0);

}

void insert(Stack \* &stk) //функцыя для увода элементаў

{

int elem;

int n;

cout << "Колькаць элементаў: "; cin >> n;

cout << "\n";

for (int i = 0; i < n; i++)

{

cin >> elem;

Stack\* newElem = new Stack;//памяць для элементаў

newElem->data = elem;//запісваем элемент ў дату

newElem->next = stk;//пераходзім на наступны элемент

stk = newElem;

}

cout << "Во!\n";

}

void inputFile(Stack \* &stk)//функцыя для ўвода ў файл

{

ofstream fin("file.txt");

if (!fin.is\_open()) {//праглядаем прысутнасць файла

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (stk != NULL)

{

fin << stk->data << "\n";

stk = stk->next;

}

fin.close();

}

cout << "Во!\n";

}

void outputFile(Stack \* &stk)//функцыя для вывада з файла

{

ifstream fout("file.txt");

int buf;

if (!fout.is\_open()) {

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (fout >> buf)

{

Stack\* newElem = new Stack;

newElem->data = buf;

newElem->next = stk;

stk = newElem;

cout << buf << " ";

}

fout.close();//зачыняем файл

}

cout << "Во!\n";

}

void clear(Stack \* &stk)//функцыя для ачысткі

{

while (stk != NULL)//пакуль не канчатак

{

Stack\* del = stk;

if (stk)

{

stk = stk->next;//новая вяршына

}

delete del;

}

cout << "\nЦяпер тут пуста!!\n";

}

void output(Stack \* &stk)//функцыя для вывада стэка

{

Stack\* output = stk;

while (output != NULL)

{

cout << output->data << " ";//вывад

output = output->next;//пераход на наступны

}

cout << "\n";

cout << "Во!\n";

}

void task(Stack\*& stk)//функцыя з асноўным заданнем

{

int count = 0;

Stack\* main = stk;

int flag = 0;

while (main != NULL)

{

Stack\* find = main->next;

while (find != NULL)

{

if (main->data == find->data)

{

count++;//лічым

}

find = find->next;//пераход на наступны

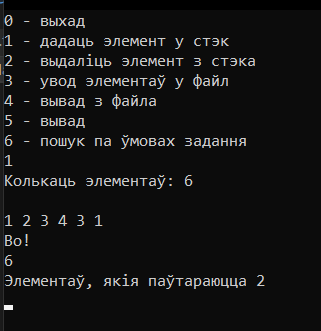
}

main = main->next;

}

cout << "Элементаў, якія паўтараюцца " << count << "\n";

}

****

****

#include "HEADER.h"

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

Stack\* first = new Stack;//выдзяленне памяці

first = NULL;

int choice;

cout << "0 - выхад\n1 - дадаць элемент у стэк\n2 - выдаліць элемент з стэка\n"

<< "3 - увод элементаў у файл\n4 - вывад з файла\n5 - вывад\n6 - пошук па ўмовах задання\n";

do//цыкл, пакуль не увесці 0

{

cin >> choice;

switch (choice)

{

case 0: exit(1); break;

case 1: insert(first); break;

case 2: clear(first); break;

case 3: inputFile(first); break;

case 4: outputFile(first); break;

case 5: output(first); break;

case 6: task(first); break;

}

} while (choice != 0);

}

void insert(Stack \* &stk) //функцыя для увода элементаў

{

int elem;

int n;

cout << "Колькаць элементаў: "; cin >> n;

cout << "\n";

for (int i = 0; i < n; i++)

{

cin >> elem;

Stack\* newElem = new Stack;//памяць для элементаў

newElem->data = elem;//запісваем элемент ў дату

newElem->next = stk;//пераходзім на наступны элемент

stk = newElem;

}

cout << "Во!\n";

}

void inputFile(Stack \* &stk)//функцыя для ўвода ў файл

{

ofstream fin("file.txt");

if (!fin.is\_open()) {//праглядаем прысутнасць файла

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (stk != NULL)

{

fin << stk->data << "\n";

stk = stk->next;

}

fin.close();

}

cout << "Во!\n";

}

void outputFile(Stack \* &stk)//функцыя для вывада з файла

{

ifstream fout("file.txt");

int buf;

if (!fout.is\_open()) {

cout << "Няма такога файла!!!!!!!!!!!";

}

else {

while (fout >> buf)

{

Stack\* newElem = new Stack;

newElem->data = buf;

newElem->next = stk;

stk = newElem;

cout << buf << " ";

}

fout.close();//зачыняем файл

}

cout << "Во!\n";

}

void clear(Stack \* &stk)//функцыя для ачысткі

{

while (stk != NULL)//пакуль не канчатак

{

Stack\* del = stk;

if (stk)

{

stk = stk->next;//новая вяршына

}

delete del;

}

cout << "\nЦяпер тут пуста!!\n";

}

void output(Stack \* &stk)//функцыя для вывада стэка

{

Stack\* output = stk;

while (output != NULL)

{

cout << output->data << " ";//вывад

output = output->next;//пераход на наступны

}

cout << "\n";

cout << "Во!\n";

}

void task(Stack\*& stk)//функцыя з асноўным заданнем

{

int count = 0;

Stack\* main = stk;

int flag = 0;

int max, min;

cout << "Увядзіце max, потым min\n";

cin >> max; cin >> min;

while (main != NULL)

{

if (min < main->data && max > main->data) {

cout << "Такі элемент ёсц\n";

flag = 1;//лічым

break;

}

main = main->next;

}

if (flag == 0) {

cout << "Няма такіх элементаў(\n";

}

}

