Белорусский государственный университет информатики и радиоэлектроники

Факультет компьютерных технологий

ОТЧЕТ

по лабораторной работе №3

на тему: **Динамическая структура СТЕК**

|  |  |
| --- | --- |
| Выполнил  студент  группы №980161  Алейчик И.Д. | Проверил  доцент  Казакевич Ю.Д. |

Минск 2020

**Лаборатоорная работа №3**

**Вариант 1**

**Тема:**

Динамическая структура СТЕК

**Цель работы:**

Изучить алгоритмы работы с динамическими структурами данных в виде стека.

**Краткие теоретические сведения:**

Стек(Stack)–структура типа LIFO(Last In, First Out) –последним вошел,первымвыйдет. Элементы в стек можно добавлять или извлекать только через его вершину. Программно стек реализуется в виде однонаправленного списка с одной точкой входа–вершиной стека. Максимальное число элементов стека не ограничивается, т. е.по мере до-бавления в стек нового элемента память под него должна запрашиваться, апри удалении –освобождаться. Таким образом, стек –динамическая структура дан-ных, состоящая из переменного числа элементов.

При работе со стеком обычно выполняются следующие операции:–формирование стека(добавление элемента в стек);–обработка элементов стека (просмотр, поиск, удаление);–освобождение памяти, занятой стеком

**Задание:**

Во всех заданиях создать список из положительных и отрицательных случайных целых чисел.

1. Разделить созданный списокна два: в первом –положительные числа, вовтором–отрицательные.

**Ход выполнения работы:**

struct Stack

{

int info;

Stack\* next;

};

Stack\* InToStack(Stack\* p, int in) {

Stack\* t = new Stack;

t->info = in;

t->next = p;

return t;

}

void View(Stack\* p, string text) {

Stack\* t = p;

if (t != NULL) {

cout << text << "| Adress to next element " << "\n";

while (t != NULL) {

cout << t->info << " | " << t->next << "\n";

t = t->next;

}

}

else {

cout << "List is empty \n";

}

}

void deleteAllStack(Stack\*\* p) {

Stack\* buf;

while (\*p != NULL) {

buf = \*p;

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

delete buf;

}

}

void singleDirStack() {

Stack\* beginSt1 = NULL;

Stack\* beginSt2 = NULL;

Stack\* beginSt3 = NULL;

srand(time(0));

for (int i = 0; i < 10; i++)

{

beginSt1 = InToStack(beginSt1, (-(rand() % 100) + (rand() % 100)));

}

while (beginSt1 != NULL) {

if (beginSt1->info < 0) {

beginSt2 = InToStack(beginSt2, beginSt1->info);

}

if (beginSt1->info > 0) {

beginSt3 = InToStack(beginSt3, beginSt1->info);

}

beginSt1 = beginSt1->next;

}

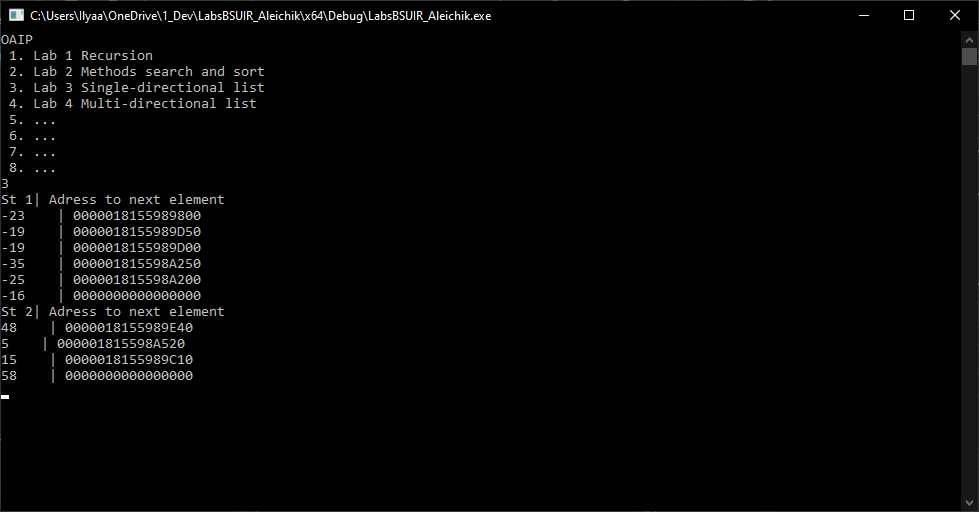
View(beginSt2, "St 1");

View(beginSt3, "St 2");

deleteAllStack(&beginSt1);

\_getch();}

**Результат работы программы:**



**Вывод:**

Были изучены алгоритмы работы с динамическими структурами данных в виде стека.