МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития

Кафедра информационных систем и технологий

# Отчет к лабораторной работе №1

Тема: **«Динамические структуры: стеки, очереди и списки»**Дисциплина: **«Информационные технологии и программирование»**

|  |  |  |
| --- | --- | --- |
|  |  | **Выполнил:** |
|  |  | Студент группы ПИЖ-б-о-23-1 Направление подготовки: 09.03.04 «Программная инженерия»  Сабодаш Алексей Игоревич |

Ставрополь 2023

**Вариант 8.**

**Задание 1.1**

Дано число N (> 0) и набор из N чисел. Создать стек, содержащий исходные числа (последнее число будет вершиной стека), и вывести указательна его вершину.

#include <iostream>

#include <stack>

using namespace std;

int main()

{

setlocale(LC\_ALL, "RU");

stack<int> st;

int n;

cout << "N = ";

cin >> n;

cout << "Ввести " << n << " чисело(а): ";

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

{

int a;

cin >> a;

st.push(a);

}

cout << "Вершина у стека - " << st.top();

return 0;

}

**Задание 1.2**

Создать очередь из вещественных чисел. Определить количество отрицательных чисел очереди. Организовать просмотр данных очереди.

#include <iostream>

#include <queue>

using namespace std;

int main()

{

setlocale(LC\_ALL, "RU");

queue<double> que;

int n, neg = 0;

cout << "N = ";

cin >> n;

cout << "Ввести " << n << " чисело(а): ";

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

{

double a;

cin >> a;

que.push(a);

if (a < 0)

{

neg++;

}

}

cout << "Кол-во отрицательных элементов: " << neg << endl;

cout << "Данные очереди:" << endl;

while (!que.empty())

{

cout << que.front() << " ";

que.pop();

}

return 0;

}

**Задание 2.1**

Создать стек строковых значений, для реализации используя односвязные списки. Реализовать операции добавления (push) и удаления (pop) элемента из стека. Добавьте в стек строки «abc», «fx», «glc», «hi», «gogo» и распечатайте содержимое стека. Удалите один элемент из стека, затем добавьте строку «the end» и распечатайте содержимое стека еще раз. Найдите количество строк в стеке, состоящих из 2 символов.

#include <iostream>

#include <string>

#include <vector>

using namespace std;

vector<string> v;

string s;

void push(string s)

{

v.push\_back(s);

}

void pop()

{

v.pop\_back();

}

void print()

{

int n = v.size();

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

{

cout << v[i] << " ";

}

cout << endl;

}

int count()

{

int n = v.size();

int count = 0;

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

{

if (v[i].length() == 2)

{

count++;

}

}

return count;

}

int main()

{

setlocale(LC\_ALL, "RU");

push("abc");

push("fx");

push("glc");

push("hi");

push("gogo");

print();

pop();

push("the end");

print();

cout << "\nКол-во строк из 2 символов: " << count() << endl;

return 0;

}

**Задание 2.2**

Создать очередь строковых значений, для реализации используя односвязные списки. Реализовать операции добавления (enqueue) и удаления(dequeue) элемента из очереди. Добавьте в очередь строки «one», «two», «three», «four», «five», «six», и распечатайте содержимое очереди. Удалите 2 элемента из очереди, затем добавьте в очередь строку «seven»и распечатайте очередь еще раз. Найдите количество строк начинающихся с букв «f» или «t».

#include <iostream>

#include <deque>

#include <string>

using namespace std;

deque<string> deq;

void enqueue(string s)

{

deq.push\_back(s);

}

void dequeue()

{

deq.pop\_front();

}

void print()

{

int n = deq.size();

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

{

cout << deq[i] << " ";

}

cout << endl;

}

int count()

{

int count = 0;

while (!deq.empty())

{

string current = deq.front();

deq.pop\_front();

if (current[0] == 'f' or current[0] == 't')

{

count++;

}

}

return count;

}

int main()

{

setlocale(LC\_ALL, "RU");

enqueue("one");

enqueue("two");

enqueue("three");

enqueue("four");

enqueue("five");

enqueue("six");

print();

dequeue();

dequeue();

enqueue("seven");

print();

cout << "\nКоличество строк начинающихся с букв «f» или «t» - " << count();

return 0;

}

**Задание 3**

Дана последовательность скобок вида «(», «)», «[», «]», «{», «}». Правильными скобочными последовательностями называются пустая последовательность, а также «(P)», «[P]», «{P}», где P – некоторая правильная последовательность. Например «{}()[]» и «{[][()()]}()» – правильные скобочные последовательности, а «(]», «[()» и «({)}» – неправильные. Определите является ли заданная строка правильным скобочным выражением. Подсказка: обработайте по очереди все символы входной строки, помещая открывающие скобки в стек, а для закрывающих скобок извлекайте открывающую скобку из стека и проверяйте, соответствуют ли они друг другу.

#include <iostream>

#include <string>

#include <stack>

using namespace std;

int main()

{

int n;

string s;

cin >> s;

n = s.size();

stack<char> st;

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

{

if (s[i] == '(' or s[i] == '[' or s[i] == '{')

{

st.push(s[i]);

}

else

{

if (st.size() == 0)

{

cout << "No";

return 0;

}

if (s[i] == ')' and st.top() == '(')

{

st.pop();

}

else if (s[i] == ']' and st.top() == '[')

{

st.pop();

}

else if (s[i] == '}' and st.top() == '{')

{

st.pop();

}

else

{

cout << "No";

return 0;

}

}

}

if (st.size() == 0)

{

cout << "Yes";

}

else

{

cout << "No";

}

return 0;

}

**Задание 4.1**

Создать линейный однонаправленный список из вещественных чисел. Определить количество элементов списка со значениями больше 7.

#include <iostream>

#include <forward\_list>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

forward\_list<float> l;

int n, count = 0;

cout << "N = ";

cin >> n;

cout << "Введите вещественные числа: ";

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

{

float num;

cin >> num;

l.push\_front(num);

if (num > 7)

{

count++;

}

}

cout << "Количество элементов списка со значениями больше 7: " << count;

return 0;

}

**Задание 4.2**

Создать линейный однонаправленный список из целых чисел. Вставить в список число 15 после каждого элемента, по модулю меньшего 2.

#include <iostream>

#include <forward\_list>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

forward\_list<int> l;

int n;

cout << "N = ";

cin >> n;

cout << "Введите целые числа: ";

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

{

int num;

cin >> num;

l.push\_front(num);

if (abs(num) < 2)

{

l.push\_front(15);

}

}

l.reverse();

cout << "Ваш новый список: ";

for (int& elem : l)

{

cout << elem << " ";

}

return 0;

}

**Задание 5.1**

Создать линейный однонаправленный список из вещественных чисел. Вставить в список число 1.5 после каждого элемента с отрицательным значением.

#include <iostream>

#include <forward\_list>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

forward\_list<float> l;

int n;

cout << "N = ";

cin >> n;

cout << "Введите целые числа: ";

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

{

float num;

cin >> num;

l.push\_front(num);

if (num < 0)

{

l.push\_front(1.5);

}

}

l.reverse();

cout << "Ваш новый список: ";

for (float& elem : l)

{

cout << elem << " ";

}

return 0;

}

**Задание 5.2**

Создать линейный двунаправленный список из символов. Удалить из списка элемент после каждого символа &.

#include <iostream>

#include <list>

#include <string>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

list<char> l;

string s;

cout << "Введите символы: ";

cin >> s;

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

{

l.push\_back(s[i]);

}

auto it = l.begin();

while (it != l.end())

{

if (\*it == '&')

{

if (next(it) != l.end())

{

it = l.erase(next(it));

}

else

{

break;

}

}

else

{

it++;

}

}

for (char& elem : l)

{

cout << elem << " ";

}

return 0;

}

**Задание 6**

Создать циклический двунаправленный список из целых чисел. Удалить из списка последний элемент со значением меньшим 15.

#include <iostream>

#include <list>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

list<int> l;

int n;

cout << "N = ";

cin >> n;

cout << "Введите ваши числа: ";

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

{

int num;

cin >> num;

l.push\_back(num);

}

l.reverse();

for (auto it = l.begin(); it != l.end(); it++)

{

if (\*it < 15)

{

l.erase(it);

break;

}

}

l.reverse();

cout << "Новый список: ";

for (int& elem : l)

{

cout << elem << " ";

}

return 0;

}   
  
**https://github.com/strit595/Progrm**