|  |  |  |
| --- | --- | --- |
|  | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования «Новосибирский государственный технический университет» | | |
|  | | |
| Кафедра параллельных вычислительных технологий | | |
| Практическое задание № 6 | | |
| по дисциплине «Методы построения и анализа алгоритмов» | | |
| **Основы работы в STL** | | |
|  | | |
|  |  | АРИФУЛОВ ТИГРАН |
| Группа ПМ-01 | ДЫЧКО АРСЕНИЙ |
|  | ЖИЖЧЕНКО ЛЕОНИД |
|  | САМСОНОВ СЕМЕН |
|  | ЯКОВЛЕВА ЕЛЕНА |
|  |  |
| Преподаватель | домников пётр александрович |
|  |  |
| Новосибирск, 2021 | | |

1. Задания:
   1. С экрана вводится последовательность целых чисел, оканчивающаяся нулем. Вывести эту последовательность в обратном порядке. Использовать стек из стандартной библиотеки С++.
   2. С экрана вводится последовательность целых чисел, оканчивающаяся нулем. Отсортировать эту последовательность по возрастанию и вывести на экран. Удалить повторения (соответственно, тестовый пример должен содержать повторения и не быть возрастающей последовательностью). Использовать функцию сортировки из стандартной библиотеки С++.
   3. Создать массив динамических массивов на С++. Заполнить его следующим образом. С клавиатуры вводятся последовательность пар чисел: номер массива и число, добавляемое в этот массив. Признаком конца ввода служит пара, состоящая из двух нулей. Отсортировать каждый динамических массив по возрастанию. Вывести содержимое массива динамических массивов в файл.
2. Текст программы

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <stack>

#include <algorithm>

#include <vector>

#define all(x) (x).begin(), (x).end()

#define pb push\_back

#define Fin(x) freopen((x), "r", stdin)

#define Fout(x) freopen((x), "w", stdout)

using namespace std;

template <typename T> inline void printvec(vector < T > vec)

{

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

{

cout << vec [ i ] << ' ';

}

}

void task1()

{

cout << "Задание 1:\nВведите последовательность целых чисел, оканчивающуюся нулём:\n";

int x;

cin >> x;

stack <int> st;

while (x)

{

st.push(x);

cin >> x;

}

cout << "Последовательность в обратном порядке:\n";

while (!st.empty())

{

cout << st.top() << " ";

st.pop();

}

cout << "\n\n";

}

void task2()

{

cout << "Задание 2:\nВведите последовательность целых чисел, содержащую повторения и оканчивающуюся нулём:\n";

int x;

cin >> x;

vector <int> v;

while (x)

{

v.push\_back(x);

cin >> x;

}

sort(all(v));

auto last = unique(all(v));

v.erase(last, v.end());

cout << "Последовательность без повторяющихся элементов и упорядоченная по возрастанию:\n";

printvec(v);

cout << "\n\n";

}

void task3()

{

cout << "Задание 3:\nВведите последовательность пар чисел: номер массива и число, добавляемое в этот массив. Признаком конца ввода служит пара, состоящая из двух нулей:\n";

int x, y;

cin >> x >> y;

vector <vector <int> > g;

while (x || y)

{

if (g.size() <= x)

{

g.resize(x + 1);

}

g[x].push\_back(y);

cin >> x >> y;

}

for (auto& i : g)

{

sort(all(i));

}

cout << "Отсортированные динамические массивы по возрастанию выведены в out.txt";

Fout("out.txt");

for (const auto& k : g)

{

static size\_t i = 0;

cout << "[" << i << "] : ";

printvec(k);

cout << endl;

++i;

}

}

int main()

{

setlocale(LC\_ALL, "rus");

task1();

task2();

task3();

return 0;

}

1. Результаты работы программы:
   1. Вывод в консоль:

Задание 1:

Введите последовательность целых чисел, оканчивающуюся нулём:

1 2 3 1 2 1 1 14 7 6 8 0

Последовательность в обратном порядке:

8 6 7 14 1 1 2 1 3 2 1

Задание 2:

Введите последовательность целых чисел, содержащую повторения и оканчивающуюся нулём:

6 6 5 7 7 1 2 2 13 736 8 6 21 0

Последовательность без повторяющихся элементов и упорядоченная по возрастанию:

1 2 5 6 7 8 13 21 736

Задание 3:

Введите последовательность пар чисел: номер массива и число, добавляемое в этот массив. Признаком конца ввода служит пара, состоящая из двух нулей:

0 1

0 5

0 9

0 2

0 3

1 4

1 5

7 4

8 1

1 1

1 23

1 25

1 -19

0 0

Отсортированные динамические массивы по возрастанию выведены в out.txt

* 1. Вывод в файл out.txt

g[0]: 1 2 3 5 9

g[1]: -19 1 4 5 23 25

g[7]: 4

g[8]: 1