**ЛАБОРАТОРНАЯ РАБОТА № 6**

**«ИССЛЕДОВАНИЕ ШАБЛОНОВ ФУНКЦИЙ»**

**Цель работы**: Исследование назначения и способа описания шаблонов функций, применение их при написании объектно-ориентированных программ.

**Вариант задания**

Описать функцию-шаблон, заданную по варианту. Проиллюстрировать ее корректную работу на различных по типу наборах данных (не менее трех: int, char и др.).

Вариант 11

Написать функцию-шаблон последовательного поиска в массиве по ключу.

Функция возвращает индекс первого найденного элемента в массиве, равного ключу.

**2. Код программы на языке C++**

#include <iostream>

template <typename t, typename t\_key>

int search(t t1, int size, t\_key key)

{

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

if (t1[i] == key) return i;

}

return 0;

}

using namespace std;

int main()

{

system("chcp 1251");

int m[] = { 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,28,30 };

short s[] = { 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,28,30 };

char c[] = { 'f','a', 's', 'd', 'e', 'n', '3', ';', 'l', '\'' };

char vibor = 'x'; int key = -1; short keys = -1;

while (key != 0) {

cout << "Введите значение для поиска в массиве типа int, 0 для перехода на поиск в short --> "; cin >> key;

int temp = search<int[], int>(m, 30, key);

cout << (temp ? "Значение найдено " : "Значение не найдено ") <<"(индекс элемента ->" << temp<< ")" << endl;

}

while (keys != 0) {

cout << "Введите значение для поиска в массиве типа short, 0 для перехода на поиск в char --> "; cin >> keys;

int temp = search<short[], short>(s, 30, keys);

cout << (temp ? "Значение найдено " : "Значение не найдено ") << "(индекс элемента ->" << temp << ")" << endl;

}

while (vibor != '0') {

cout << "Введите значение для поиска в массиве типа char, 0 для выхода--> "; cin >> vibor;

int temp = search<char[], char>(c, 30, key);

cout << (temp ? "Значение найдено " : "Значение не найдено ") << "(индекс элемента ->" << temp << ")" << endl;

}

}

**3. Тестирование и отладка**

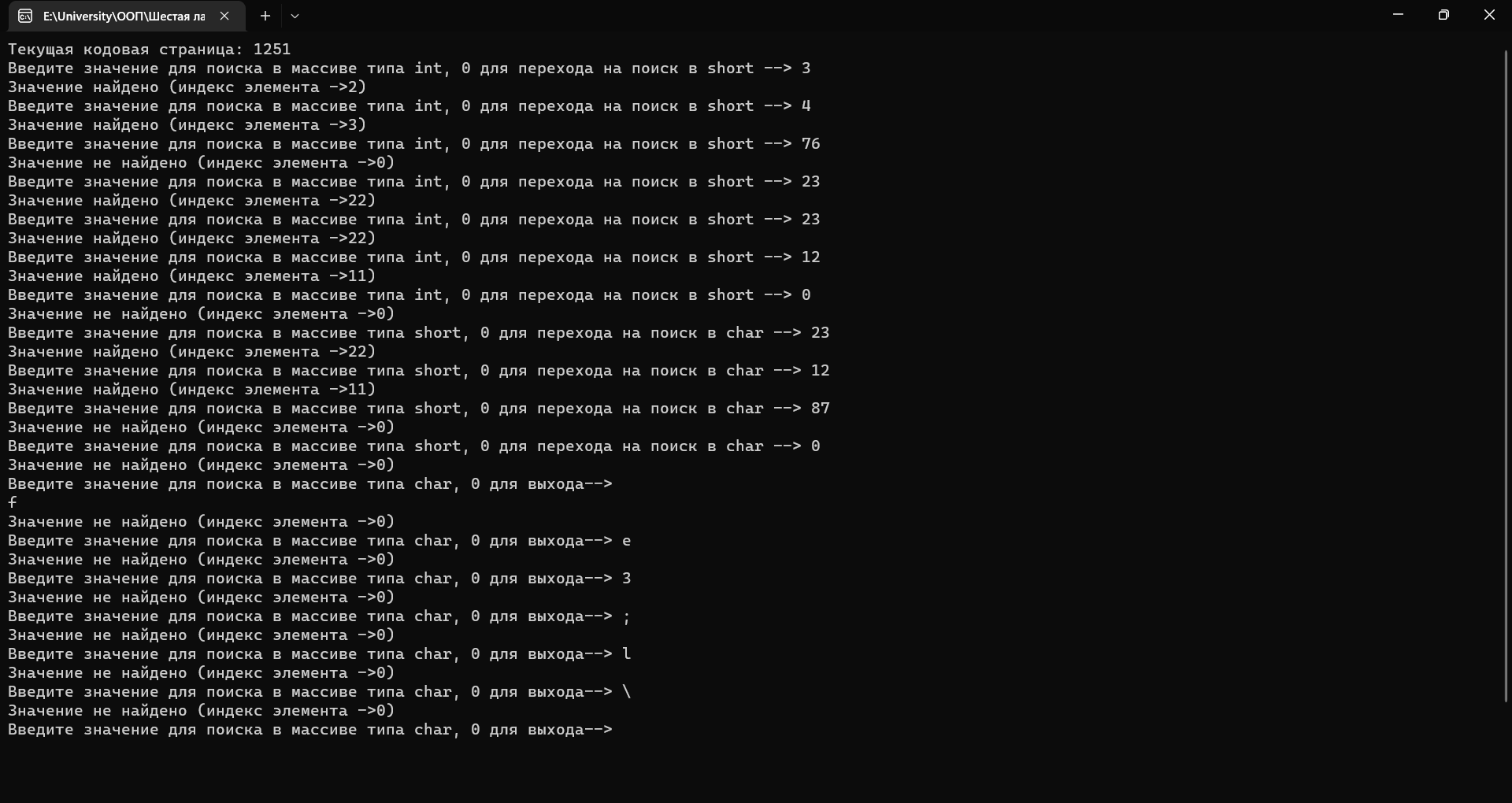
Для тестирования данной программы, как хранящиеся в массивах значения, были заранее проинициализированы значения. Далее пользователю предлагается ввести элемент, и программа попробует его найти, используя ссответствующий темплейт.

Рисунок 1 – Результат выполнения программы.

В результате тестирования, видно, что один и тот же код поиска работает с разными типами данных.

**Вывод**

При выполнении данной лабораторной работы были получены навыки работы с шаблонами. Положительными качествами использования шаблонов являются реализация полиморфизма и использование более безопасной системы типов. Недостатками являются: повышенный объём итогового кода ввиду хранения полного кода для каждой реализации; не для всех типов реализация функции оптимальна.