Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

направление подготовки: 09.03.01 - «Информатика и вычислительная техника»

**О Т Ч Е Т**

**по лабораторной работе №14 (Интерполяционный поиск)**

**по дисциплине**

**«Основы алгоритмизации и программирования» семестр 2**

Выполнил студент гр. ИВТ-21-1б

Ипатов Дмитрий Сергеевич

Проверил:

Ст. Преподаватель кафедры ИТАС

Яруллин Д.В.

(оценка) (подпись)

г. Пермь-2022

**Постановка задачи:**

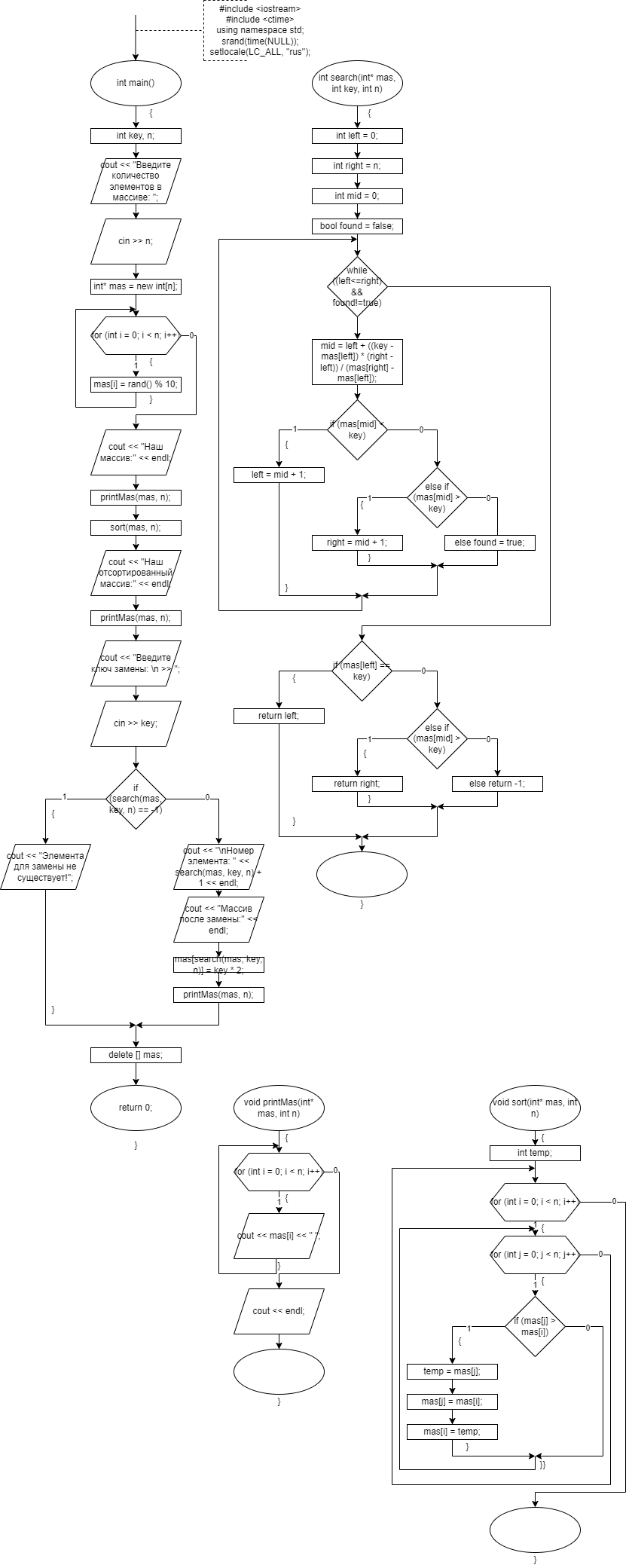
Для интерполяционного поиска: Дан массив целых чисел (или символов, если это дано по заданию), выполнить задание из своего варианта, используя указанный метод поиска. Массив можно заполнить вручную или случайными числами, размерность массива любая (не меньше 5). Массив для интерполяционного поиска предварительно отсортировать.

Найти и заменить значение N на значение K, которое в два раза больше N

Анализ задачи:

1. Создаем массив, выделяем под него память, до этого вводим размер массива с клавиатуры, выводим его;
2. Создаем отдельную функцию (sort), в которую передаем массив и его размер, в ней создаем доп. переменную (третий стакан) и сортируем в ней наш массив и также выводим его;
3. Задаем ключ поиска (элемент, который нужно найти);
4. Создаем функцию поиска, в неё передаем наш отсортированный массив, ключ поиска и количество элементов. Создаем левую и правую границы, переменную для записи индекса элемента;
5. Создаем цикл, который будет работать, пока левый элемент не станет больше или равным правому и не найдется нужный нам элемент. В цикле используем формулу для нахождения индекса элемента;
6. Если левая граница стала равна ключу поиска – возвращаем значение left, если правая – возвращаем значение right, если элемент не найден возвращаем -1.
7. В функции main создаем проверку, в которой выводим надпись на экран был ли найден элемент. Если нет, то выводим – “Элемента для замены не существует!”, если найден, то выводим номер элемента, меняем этот элемент на ключ поиска умноженный на 2 и снова выводим наш массив (Массив после замены);

**Блок схема:**



**Код программы:**

#include <iostream>

#include <ctime>

using namespace std;

int search(int\* mas, int key, int n)

{

int left = 0;

int right = n;

int mid = 0;

bool found = false;

while ((left<=right) && found!=true)

{

mid = left + ((key - mas[left]) \* (right - left)) / (mas[right] - mas[left]);

if (mas[mid] < key) left = mid + 1;

else if (mas[mid] > key) right = mid + 1;

else found = true;

}

if (mas[left] == key) return left;

else if (mas[right] == key) return right;

else return -1;

}

void printMas(int\* mas, int n)

{

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

{

cout << mas[i] << " ";

}

cout << endl;

}

void sort(int\* mas, int n)

{

int temp;

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

{

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

{

if (mas[j] > mas[i])

{

temp = mas[j];

mas[j] = mas[i];

mas[i] = temp;

}

}

}

}

int main()

{

srand(time(NULL));

setlocale(LC\_ALL, "rus");

int key, n;

cout << "Введите количество элементов в массиве: ";

cin >> n;

int\* mas = new int[n];

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

{

mas[i] = rand() % 10;

}

cout << "Наш массив:" << endl;

printMas(mas, n);

sort(mas, n);

cout << "Наш отсортированный массив:" << endl;

printMas(mas, n);

cout << "Введите ключ замены: \n >> ";

cin >> key;

if (search(mas, key, n) == -1)

{

cout << "Элемента для замены не существует!";

}

else

{

cout << "\nНомер элемента: " << search(mas, key, n) + 1 << endl;

cout << "Массив после замены:" << endl;

mas[search(mas, key, n)] = key \* 2;

printMas(mas, n);

}

delete[] mas;

return 0;

}