

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра систем автоматизированного проектирования**

**ОТЧЕТ**  
**по лабораторной работе №4**  
**по дисциплине «Программирование»**  
**Тема: Файлы для ввода-вывода данных. Обработка последовательности**  
**значений с использованием массивов**

Студентка гр. 3353

Карпенко А.Ю

Преподаватель

Калмычков В.А

Санкт-Петербург

2023

## ОГЛАВЛЕНИЕ:

Исходная формулировка задания	1
Цели работы	1
Математическая постановка задачи	1
Контрольный пример	1
Особенности выполнения на компьютере	1
Реализация ввода/вывода	2
Организация хранения данных	3
Представление алгоритма решения задачи. Блок-схема	3
Текст программы	5
Результаты работы программы	5
Вывод	6

### Исходная формулировка задания

Составить программу, которая будет забирать массив данных из файла. Для этого массива задан произвольный индекс. Удалите ту часть массива от него, которая имеет максимальную сумму. После чего вывести полученный массив в файл.

### Цели работы

- изучение работы с массивами различных типов.
- изучение библиотеки fstream
- закрепление условных конструкций(ветвления)
- закрепления работы с файлами

### Математическая постановка задачи

Дано: массив данных, границы числового диапазона

Найти: номера элементов массива, которые попадают в числовой диапазон

Возьмем массив  $A$  с количеством элементов  $L$ , в котором  $A_n$  -  $n$ -ый элемент массива.  $A$  р-какой-то произвольный индекс. Числовой диапазон  $[x, y]$ .

Если  $x \leq A_n \leq y$ , то элемент подходит  $\rightarrow n$  идет в ответ, иначе игнорируется.

### Контрольный пример

	кол-во элементов массива	границы диапазона		0	1	2	3	4
входной массив	5	29	11	10,9	11,1	20	25,5	28,99
попадает ли в диапазон?				нет	да	да	да	да
итоговый массив				1	2	3	4	

### Особенности выполнения на компьютере

Тип float представляет вещественное число с плавающей точкой в диапазоне от  $-3.4E-38$  до  $3.4E38$  и в памяти занимает 4 байта (32 бита), что накладывает свои ограничения на величины  $x$ ,  $y$ ,  $\min$ ,  $\max$  и на элементы массивов. Тип переменной int, который хранит целочисленные значения в диапазоне от  $-32768$  до  $32767$  и занимает 2 байта (16 бит) накладывает свои ограничения на  $n$ ,  $l$ . В программе рассматриваются два способа задачи массива.

Статический:

$a[\text{количество элементов}]$

Динамический:

float \*a;  
a=new float[(количество элементов)];

Обращение к его элементам тоже выполняется по-разному

Статический:

a[номер элемента]

Динамический:

\*(a+(номер элемента))

В процессе выполнения программы потребуются реализация условных конструкций:

```
if (условие 1) {(выполняется если условие 1 истина);}
else if(условие 2) {выполняется если условие 2 истина;}
else {выполняется если условие 1 И условие 2 ложь;};
```

А также потребуются циклы, будем использовать циклы:

```
while(условие провидения цикла)
{
    (тело цикла)
}
for(начальные условия; условия продолжения; действие после завершения одного тела)
{
    (тело цикла)
}
```

Кроме того, для динамического потребуется освободить взятую для него память:

delete [] a

### Реализация ввода/вывода

Программа выводит приветствие в консоль. Далее идет работа с файлами.

Автор работы: Карпенко Анастасия

Группа: 3353

Дата создания: 19.10.23

Номер задания: 40-г

*Чтение массива из файла* организовано путем взятия каждого элемента оператором извлечения «<>>».

Статический массив: первое значение отвечает за количество элементов в массиве.

Динамический массив: первое значение считывается, но количество элементов подсчитывает программа.

*Помещение результата в файл.*

Вывод - исходный массив: \*выводит по элементам исходный массив в строку\*

Вывод - результат работы: \*в столбец выводит по элементам массив полученный в результате\*

Для ввода в консоль будем использовать:

cin>>(имя переменной)

Для ввода в консоль мы используем библиотеке конструкцию iostream

cout<<«Текст»<<(имя переменной);

Для ввода из файла-библиотеку fstream

file>>(имя переменной);

Для вывода в файл-библиотеку fstream

res<<«Текст»<<(имя переменной);

Так же стоит упомянуть, что для работы с файлами потребуется их открывать, а также открывать потоки вывода. После работы с файлами необходимо их закрыть.

```
ifstream (название потока) ("имя файла");
ofstream (название потока) ("имя файла");
(название потока).close();
```

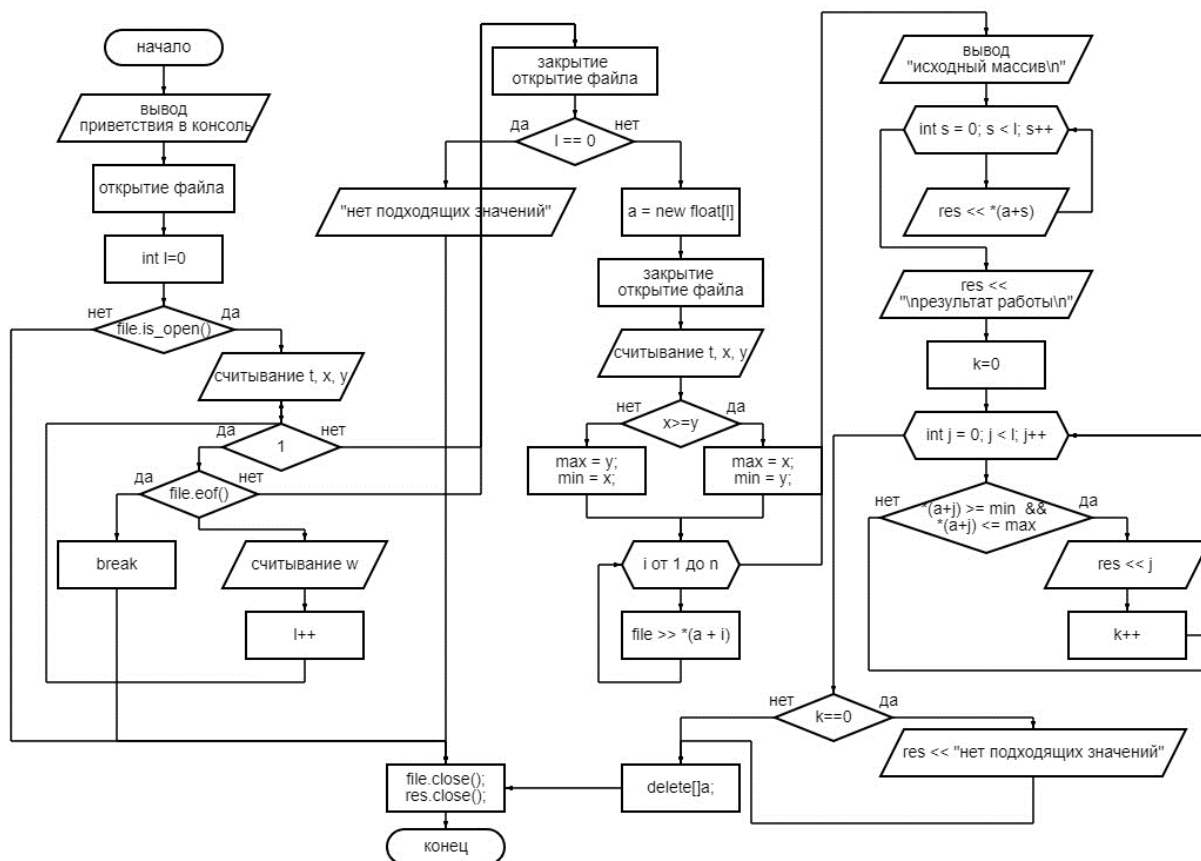
### Организация хранения данных

Название переменной	Тип переменной	Пояснение
n	const int	Длина статического массива
l	int	Длина массива во входном файле
x, y	float	Границы диапазона
min, max	float	Отсортированные границы диапазона
a[n]	float	Статический массив
*a	float	Динамический массив
t, w	float	Мусорные переменные

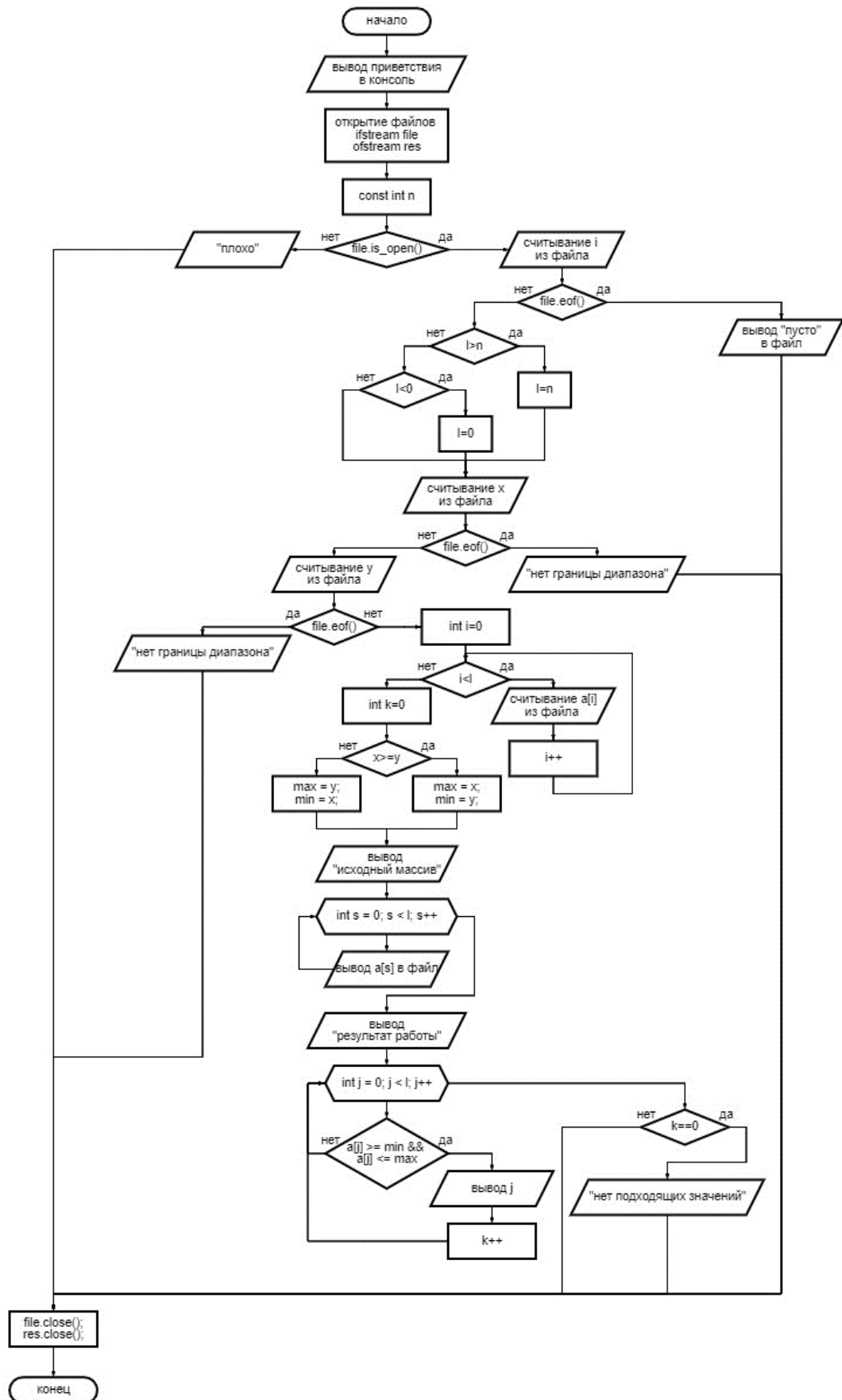
### Представление алгоритма решения задачи. Блок-схема

Программа представляется. Открываются файлы для считывания данных и записи результатов. Проходит проверка на открытие файла, а также на его пустоту. Считывается кол-во элементов массива (в динамической версии это значение игнорируется и подсчитывается программой), границы диапазона, сами элементы массива. Далее каждый элемент массива сравнивается с границами диапазона. Подходящие значения, а точнее их индексы, записываются в выходной файл. В конце программы файлы закрываются. В динамической версии перед закрытием файлов происходит очищение массива.

Блок-схема динамической версии программы.



Блок-схема статической версии программы.



## Текст программы

Статический массив	Динамический массив
<pre> #include &lt;iostream&gt; #include&lt;fstream&gt; using namespace std; int main() {     setlocale(LC_ALL, "RU");     cout &lt;&lt; "Автор работы: Карпенко Анастасия\n";     cout &lt;&lt; "Группа: 3353\n";     cout &lt;&lt; "Дата создания: 19.10.23\n";     cout &lt;&lt; "Номер задания: 40-г\n";      ifstream file("C:\\Users\\Анастасия\\source\\repos\\TEST\\TEST\\1test.txt"); ofstream res("C:\\Users\\Анастасия\\source\\repos\\TEST\\TEST\\1ans.txt"); const int n = 5; int l; float x, y, min, max; float a[n]; if (file.is_open()) {     file &gt;&gt; l;     if (file.eof()) {         res &lt;&lt; "нуто" &lt;&lt; "\n";     }     else {         if (l &gt; n) { l = n; };         if (l &lt; 0) { l = 0; };         file &gt;&gt; x;         if (file.eof()) {             res &lt;&lt; "нет границы диапазона" &lt;&lt; "\n";         }         else {             file &gt;&gt; y;             if (file.eof()) {                 res &lt;&lt; "нет границы диапазона" &lt;&lt; "\n";             }             else {                 int i = 0;                 while (i &lt; l) {                     file &gt;&gt; a[i];                     i++;                 }                 int k=0;                 if (x &gt;= y) {                     max = x;                     min = y;                 }                 else {                     max = y;                     min = x;                 }                 res &lt;&lt; "исходный массив\n";                 for (int s = 0; s &lt; l; s++) {                     res &lt;&lt; a[s] &lt;&lt; " ";                 }                 res &lt;&lt; "результат работы\n";                 for (int j = 0; j &lt; l; j++) {                     if (a[j] &gt;= min &amp;&amp; a[j] &lt;= max) {                         res &lt;&lt; j &lt;&lt; '\n';                         k++;                     }                 }                 if (k==0) {                     res &lt;&lt; "нет подходящих значений" &lt;&lt; '\n';                 }             }         }     } } else {     res &lt;&lt; "плохо" &lt;&lt; '\n'; } file.close(); res.close(); } </pre>	<pre> #include &lt;iostream&gt; #include&lt;fstream&gt; using namespace std; int main() {     setlocale(LC_ALL, "RU");     cout &lt;&lt; "Автор работы: Карпенко Анастасия\n";     cout &lt;&lt; "Группа: 3353\n";     cout &lt;&lt; "Дата создания: 19.10.23\n";     cout &lt;&lt; "Номер задания: 40-г\n";      ifstream file("C:\\Users\\Анастасия\\source\\repos\\TEST\\TEST\\1test.txt"); ofstream res("C:\\Users\\Анастасия\\source\\repos\\TEST\\TEST\\1ans.txt"); int l=0; float *a; float t, w, x, y; float min, max; if (file.is_open()) {     file &gt;&gt; t;     file &gt;&gt; x;     file &gt;&gt; y;     while (1){         if (file.eof()) { break; };         file &gt;&gt; w;         l++;     }     file.close();     ifstream file("C:\\Users\\Анастасия\\source\\repos\\TEST\\TEST\\1test.txt"); if (l == 0) {     res &lt;&lt; "нет подходящих значений"; } else {     a = new float[l];     file.close();     ifstream file("C:\\Users\\Анастасия\\source\\repos\\TEST\\TEST\\1test.txt");     file &gt;&gt; t;     file &gt;&gt; x;     file &gt;&gt; y;     if (x&gt;=y){         max = x;         min = y;     }     else {         max = y;         min = x;     }     for (int i = 0; i &lt; l; i++) {         file &gt;&gt; *(a + i) ;     }     res &lt;&lt; "исходный массив\n";     for (int s = 0; s &lt; l; s++) {         res &lt;&lt; *(a+s) &lt;&lt; " ";     }     res &lt;&lt; "\nрезультат работы\n";     int k = 0;     for (int j = 0; j &lt; l; j++) {         if (*(a+j) &gt;= min &amp;&amp; *(a+j) &lt;= max) {             res &lt;&lt; j &lt;&lt; '\n';             k++;         }     }     if (k == 0) {         res &lt;&lt; "нет подходящих значений" &lt;&lt; '\n';     }     delete[]a; } file.close(); res.close(); } </pre>

## Результаты работы программы

Входной файл	Выходной файл. Статика	Выходной файл. Динамика
5	исходный массив	исходный массив
29	10.9 11.1 20 25.5 28.99	10.9 11.1 20 25.5 28.99
11	результат работы	результат работы
10.9	1	1
11.1	2	2
20	3	3
25.5	4	4
28.99		

50 29 11 10.9 11.1 20 25.5 28.99	исходный массив 10.9 11.1 20 25.5 28.99 результат работы 1 2 3 4	исходный массив 10.9 11.1 20 25.5 28.99 результат работы 1 2 3 4
0 29 11 10.9 11.1 20 25.5 28.99	исходный массив результат работы нет подходящих значений	исходный массив 10.9 11.1 20 25.5 28.99 результат работы 1 2 3 4

### Вывод

На основе работы изучены массивы и их типы, особенности работы с ними. А также библиотека `fstream` для записи с чтения файлов. А также закреплены навыки работы с циклами и условным конструкциями.