

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

**ОТЧЕТ**  
**по лабораторной работе №5**  
**по дисциплине «Программирование»**  
**Тема: Использование файлов для ввода-вывода данных.**  
**Программирование функций пользователя**

Студент гр. 2302

\_\_\_\_\_

Коняев А.Е.

Преподаватель

\_\_\_\_\_

Калмычков В.А.

Санкт-Петербург

2022

## Оглавление

Исходная формулировка задачи.....	3
Анализ задачи и устранение неясностей.....	3
Математическая постановка задачи .....	3
Контрольный пример .....	3
Особенности решения задачи на компьютере .....	4
Организация интерфейса пользователя в файле .....	4
Организация данных в программе .....	5
Описание функций .....	6
Организация ввода/вывода .....	6
Представление алгоритма решения задачи .....	7
Текст программы.....	9
Результаты работы программы .....	11
Вывод о проделанной работе.....	11

## Исходная формулировка задачи

Найти наибольший элемент заштрихованной области таблицы A размера  $N \times N$ .

ж)     *m*

X	X	X	X	X	X	X
			X			
			X			
			X			
X	X	X	X	X	X	X

*k*

## Анализ задачи и устранение неясностей

Программа учитывает использование пустых строк и подряд идущих пробелов. Если  $n$  или  $m$  больше, чем соответствующее количество строк или столбцов матрицы, то оно округляется до реального количества столбцов или строк матрицы.

## Математическая постановка задачи

Необходимо найти максимальный элемент из  $k$ -ой строки,  $m$ -ого столбца и последней строки путём сравнения текущего значения с предыдущим, содержащимся в переменной.

## Контрольный пример

```
7 5
0 3
9 5      23 13 2
5 3 5 32
6      2 3 4
1 2      3 9

131 31 32 3
1 23      12 31
```

## Особенности решения задачи на компьютере

При решении задачи использовались переменные типа `int`, `unsigned` и `const unsigned`, `bool`. Отсюда,  $a, b, size, i, j, count\_ws, n, m, k \in [0; 4\,294\,967\,295]$ ,  $num, A[i][j], max \in [-2\,147\,483\,647; 2\,147\,483\,647]$ ,  $flag$  и  $ws\_flag$  принимает значения 0 или 1.

## Организация интерфейса пользователя в файле

in.txt:

$\underbrace{d\ d}_2$   
 $\underbrace{d\ d}_2$   
d d d d d  
d d d d  
d d d d d d  
d d  
...

InpArr.txt:

$\underbrace{d\ d}_2$   
 $\underbrace{d\ d}_2$   
d d d  
d d d  
d d d  
...

out.txt:

Задача: Найти наибольший элемент заштрихованной области таблицы размера  $N*N$

Автор: Коняев Александр Евгеньевич; Группа: 2302; Версия 4.2

Дата начала: 09.11.2022; Дата окончания: 16.11.2022

Res: d

### Организация данных в программе

Тип	Название	Назначение
int[]	A[]	Матрица с числами из файла in.txt
int	num	Временная переменная для считывания лишних значений из файла
	max	Искомое значение максимального элемента
unsigned	i	Итерируемая переменная цикла
	j	
	a	Реальное количество строк
	b	Реальное количество столбцов
	size	Подсчёт непустых строк
	count_ws	Подсчёт пробелов
	m	Параметр столбца для задания
	k	Параметр строки для задания
const unsigned	n	Максимальное количество строк исходной матрицы
	m	Максимальное количество столбцов исходной матрицы
bool	flag	Флаг для проверки на пустую строку
	ws_flag	Флаг для проверки на подряд идущие пробелы

char	tmp	Временная переменная для поиска перехода на следующую строку
fstream	f	Файл in.txt, InpArr.txt или out.txt
	g	Файл in.txt для функции, где открыто сразу 2 файла

### Описание функций

Имя	Назначение	Параметры			
		Входные	Выходные	Модифицируемые	Транзитные
SizeArr	Нахождение реального размера матрицы			a, b	
InpArr	Чтение данных из файла	a, b	f	A[n][m], m, k	
Process	Поиск максимального значения	A[n][m], a, b, m, k		max	
OutRes	Вывод результата в файл	max	f		
main	Основная функция				

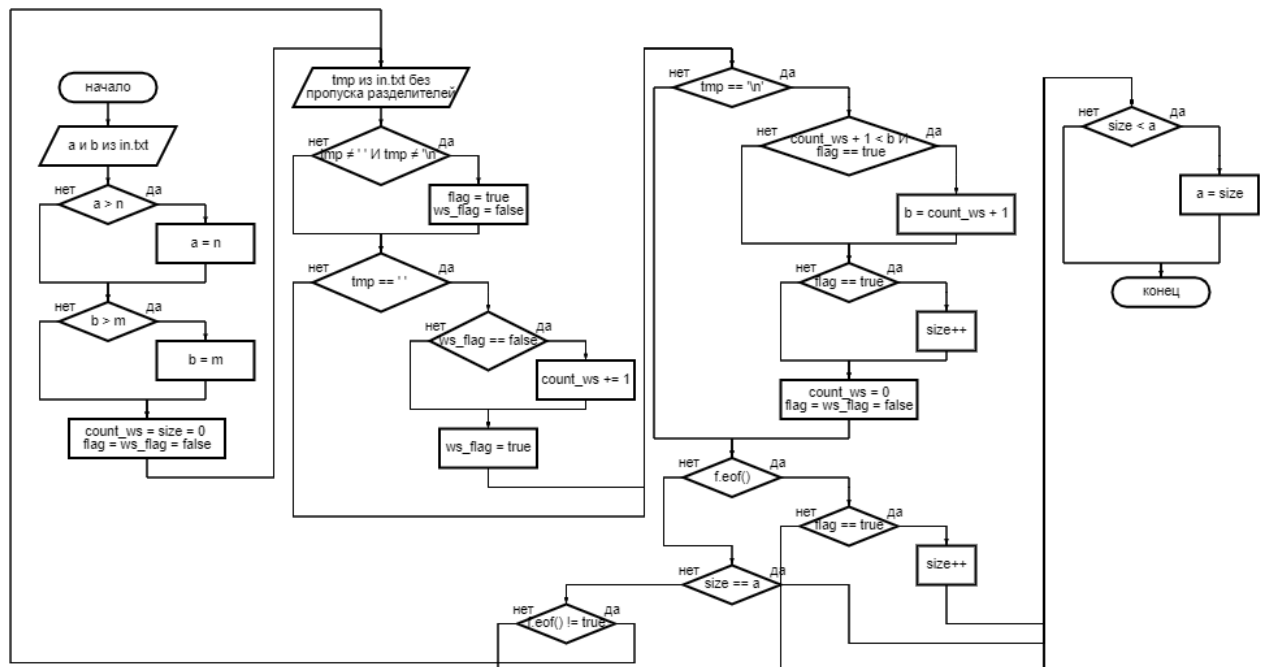
### Организация ввода/вывода

	iomanip	fstream
Ввод		f >>
Вывод		f <<
Вывод из буфера в поток	flush	

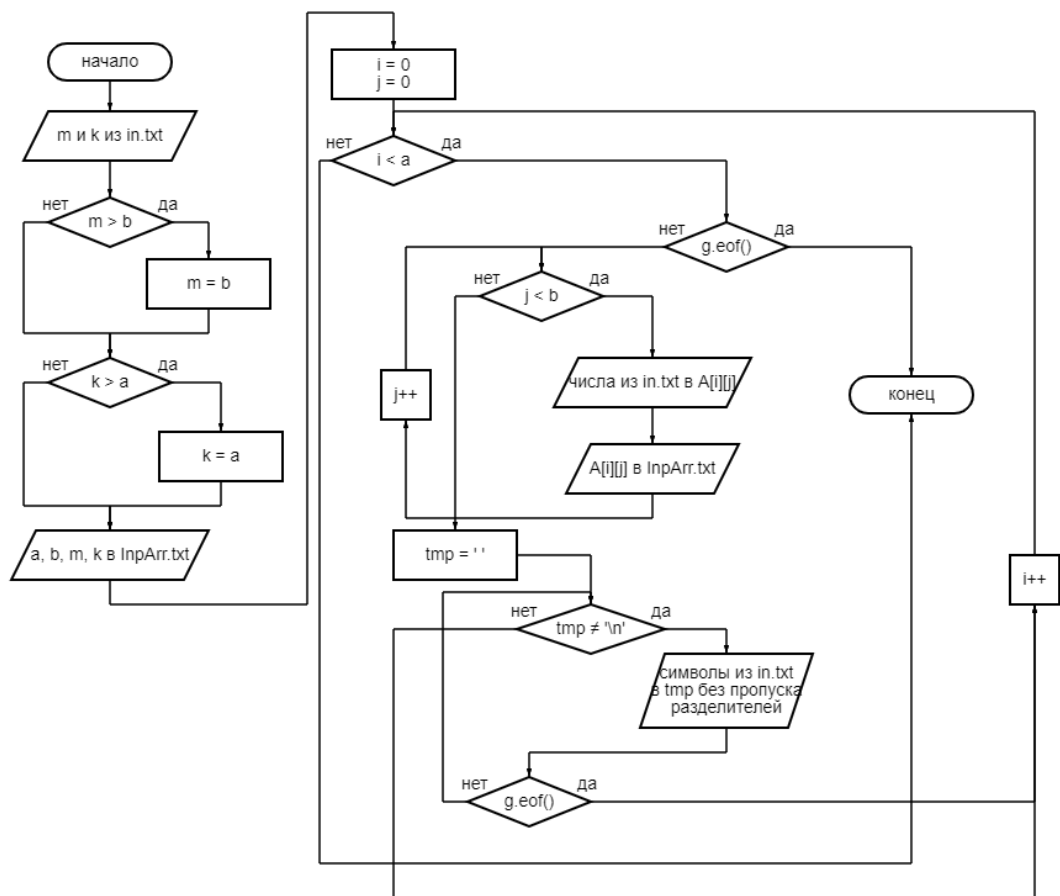
Работа с разделителем	skipws noskipws	
Настройка вывода	setw() left	
Работа с файлом		fstream f f.open() f.close()

## Представление алгоритма решения задачи

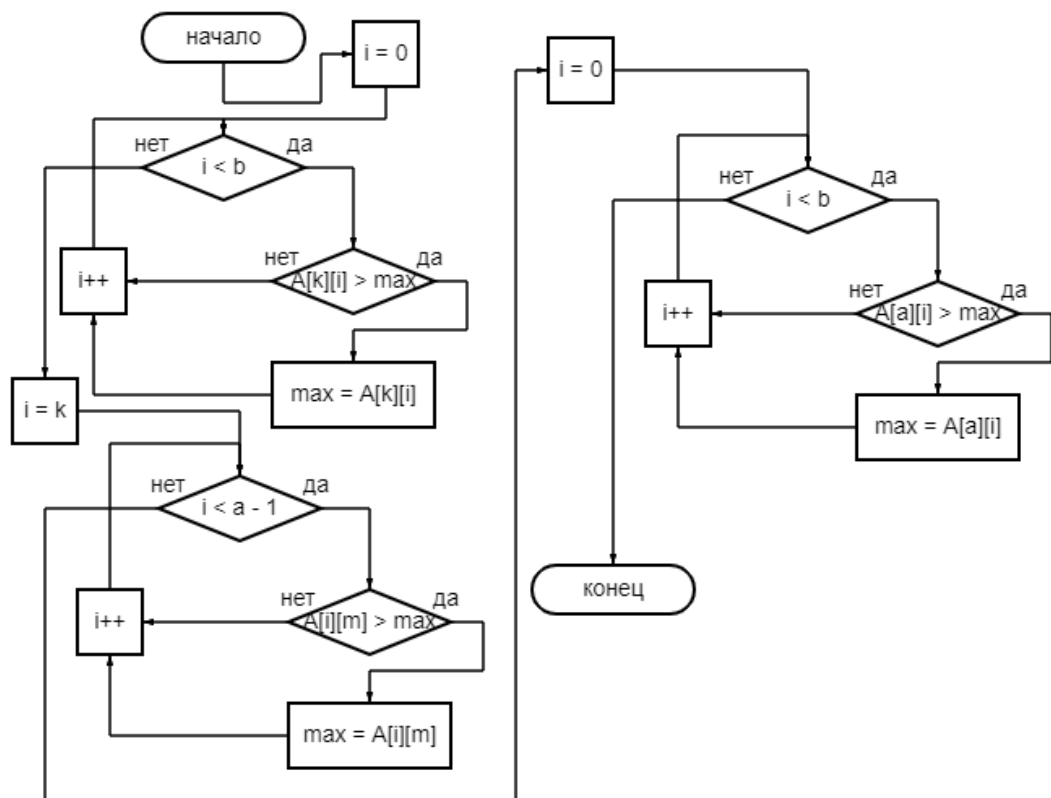
SizeArr:



InpArr:



Process:

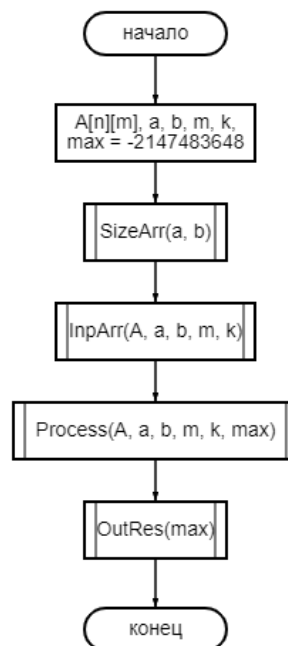




OutRes:



main:



## Текст программы

```
#include <iomanip>
#include <fstream>

const unsigned n = 200, m = 100;

void SizeArr(unsigned& a, unsigned& b) {
    std::fstream f; f.open("in.txt", std::ios::in);
    if (f.is_open()) {
        f >> a >> b;
        unsigned size = 0;
        char tmp; int num;
        if (a > n) a = n;
        if (b > m) b = m;
        f >> num >> num;
        unsigned count_ws = 0; bool flag = false, ws_flag = false;
        do {
            f >> std::noskipws >> tmp;
            if (tmp != ' ' && tmp != '\n') {
                flag = true;
            }
        } while (f.get() != '\n');
```

```

        ws_flag = false;
    }
    if (tmp == ' ') {
        if (ws_flag == false) count_ws += 1;
        ws_flag = true;
    }
    if (tmp == '\n') {
        if (count_ws + 1 < b && flag == true) b = count_ws + 1;
        if (flag == true) size++;
        count_ws = 0; flag = false; ws_flag = false;
    }
    if (f.eof()) {
        if (flag == true) size++;
        break;
    }
    if (size == a) break;
} while (f.eof() != true);
if (size < a) a = size;
f.close();
}

}

void InpArr(int A[n][m], unsigned a, unsigned b, unsigned& m, unsigned& k) {
    std::fstream g; g.open("in.txt", std::ios::in);
    std::fstream f; f.open("InpArr.txt", std::ios::out);
    if (g.is_open() && f.is_open()) {
        int num; char tmp;
        g >> num >> num;
        g >> m >> k;
        if (m > b) m = b;
        if (k > a) k = a;
        f << a << ' ' << b << '\n' << m << ' ' << k << '\n' << std::flush;
        for (unsigned i = 0; i < a; i++) {
            if (g.eof()) break;
            for (unsigned j = 0; j < b; j++) {
                g >> std::skipws >> A[i][j];
                f << std::left << std::setw(4) << A[i][j] << ' ';
            }
            tmp = ' ';
            while (tmp != '\n') {
                g >> std::noskipws >> tmp;
                if (g.eof()) break;
            }
            f << '\n';
        }
        g.close();
        f.close();
    }
}

void Process(int A[n][m], unsigned a, unsigned b, unsigned m, unsigned k, int& max)
{
    for (unsigned i = 0; i < b; i++){
        if (A[k][i] > max) max = A[k][i];
    }
    for (unsigned i = k; i < a - 1; i++) {
        if (A[i][m] > max) max = A[i][m];
    }
    for (unsigned i = 0; i < b; i++) {
        if (A[a][i] > max) max = A[a][i];
    }
}

void OutRes(int max) {
    std::fstream f; f.open("out.txt", std::ios::out);

```

```

        if (f.is_open()) {
            f << "Задача: Найти наибольший элемент заштрихованной области таблицы
размера N*N\n"
            << "Автор: Коняев Александр Евгеньевич; Группа: 2302; Версия
4.2\n"
            << "Дата начала: 09.11.2022; Дата окончания: 16.11.2022\n\n";
            f << "Res: " << max;
            f.close();
        }
    }

int main()
{
    int A[n][m]; unsigned a, b, m, k; int max = -2147483648;
    SizeArr(a, b);
    InpArr(A, a, b, m, k);
    Process(A, a, b, m, k, max);
    OutRes(max);
}

```

## Результаты работы программы

InpArr.txt:

6	4		
0	3		
9		5	23
5		3	5
6		2	3
1		2	3
131	31	32	3
1	23	12	31

out.txt:

Задача: Найти наибольший элемент заштрихованной области таблицы размера N\*N  
 Автор: Коняев Александр Евгеньевич; Группа: 2302; Версия 4.2  
 Дата начала: 09.11.2022; Дата окончания: 16.11.2022  
 Res: 131

## Вывод о проделанной работе

В ходе выполнения лабораторной работы я научился работать с пользовательскими функциями и двумерными массивами в C++.