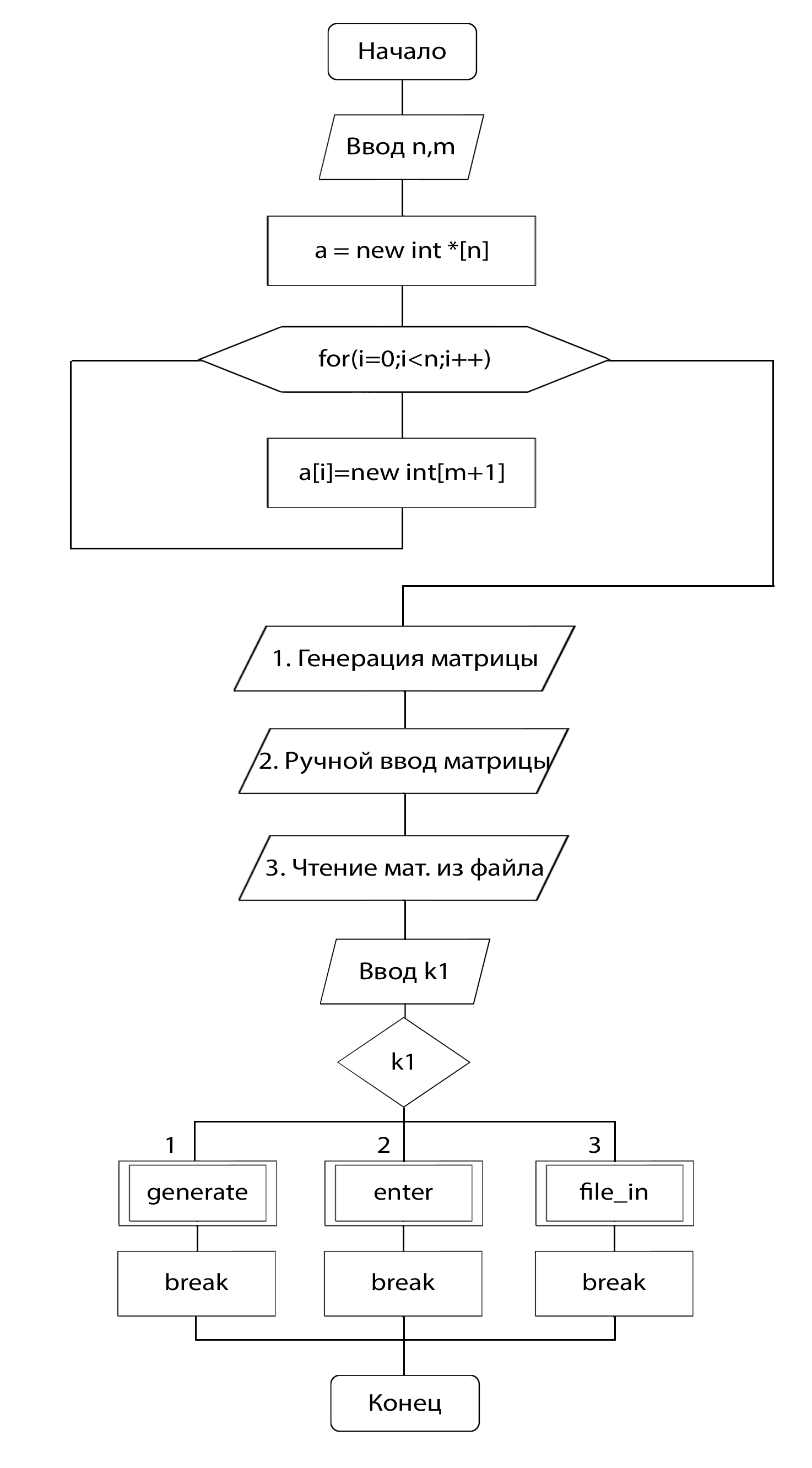
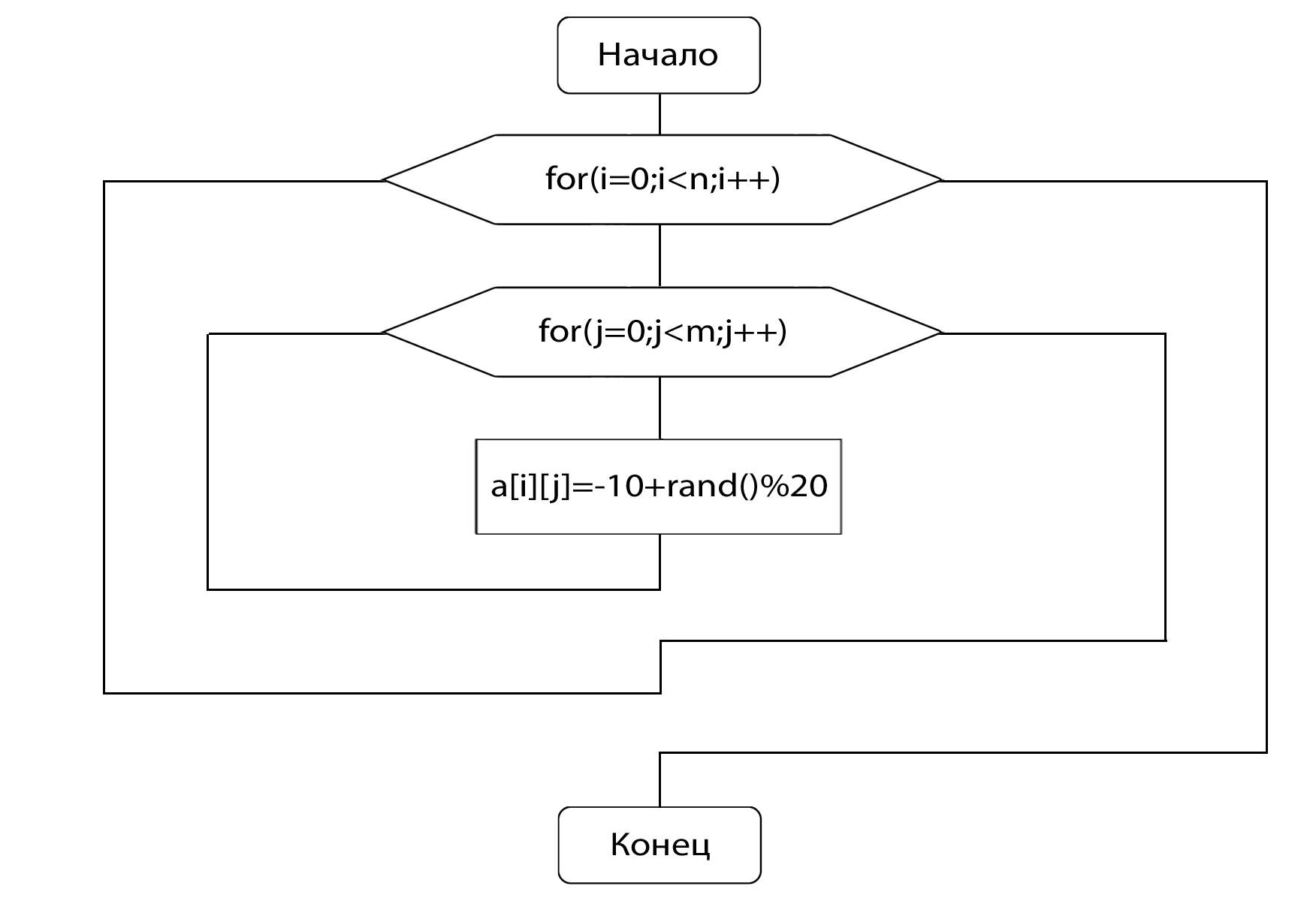
Задание №1

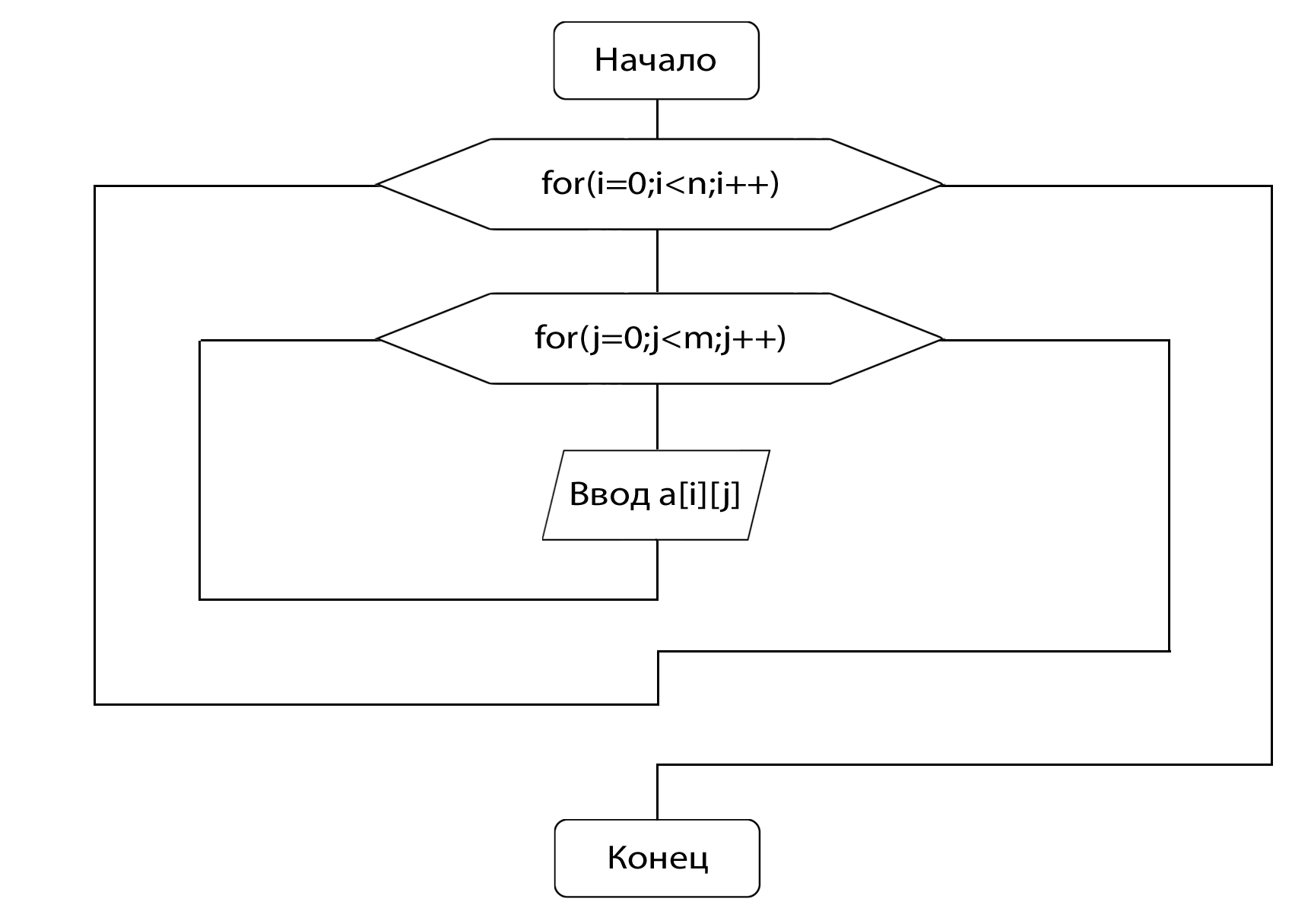
Условие:

Изменить матрицу, заменив каждый отрицательный элемент, лежащий выше главной диагонали, его абсолютной величиной. Найти также сумму элементов главной диагонали.

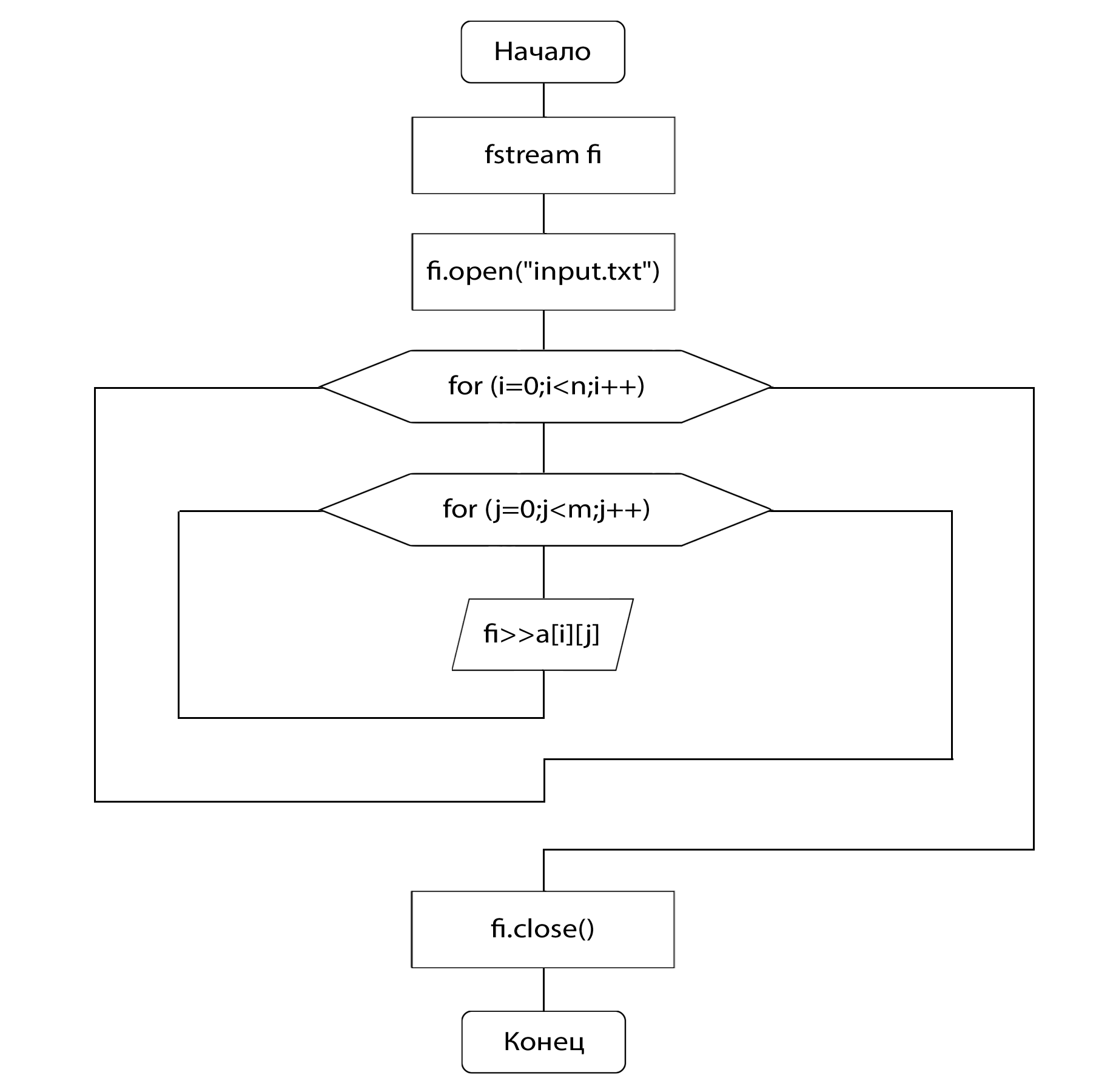
Алгоритм процедуры initarray:



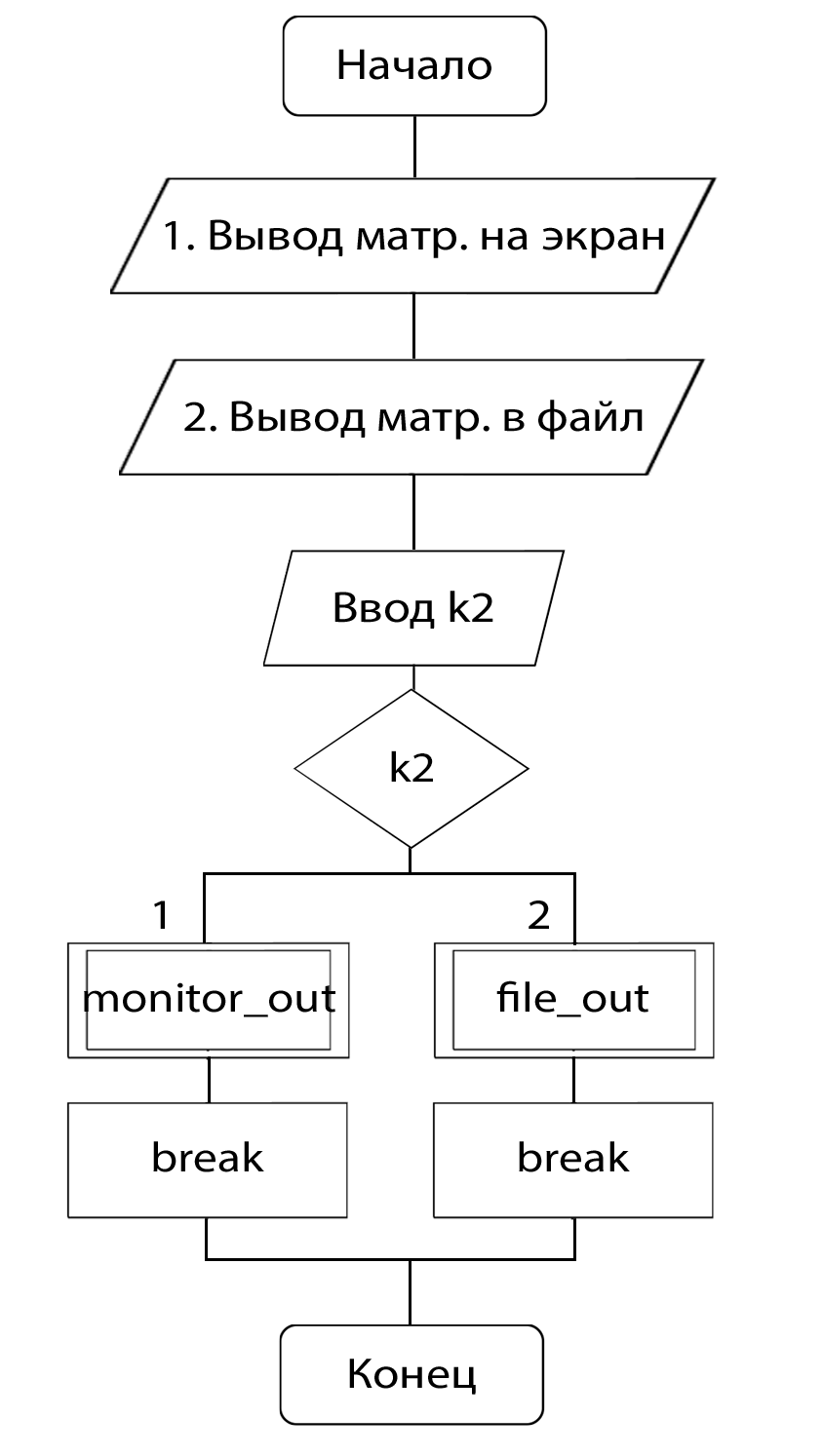
Алгоритм процедуры generate: Алгоритм процедуры enter:



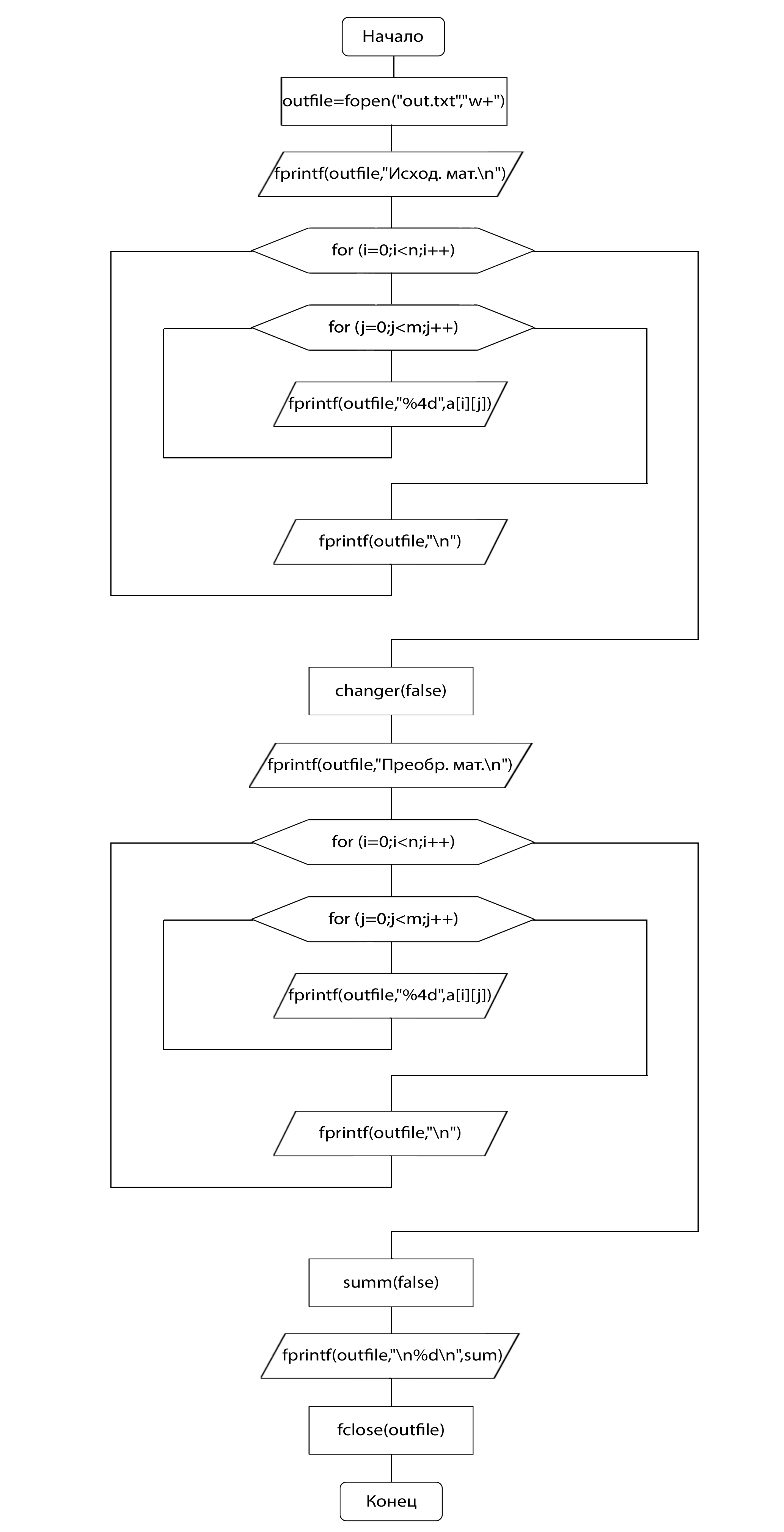
Алгоритм процедуры file\_in:



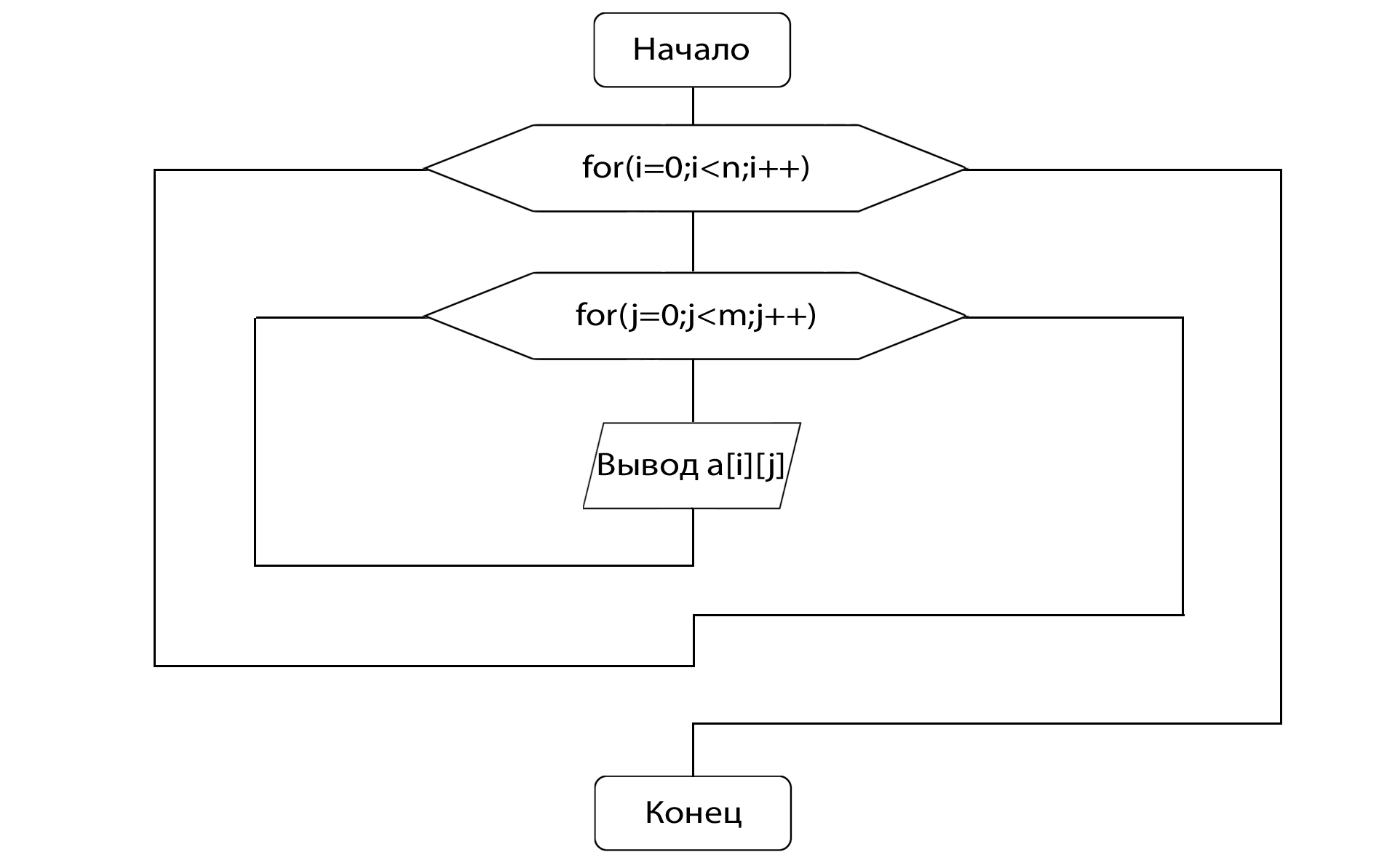
Алгоритм процедуры outarray:



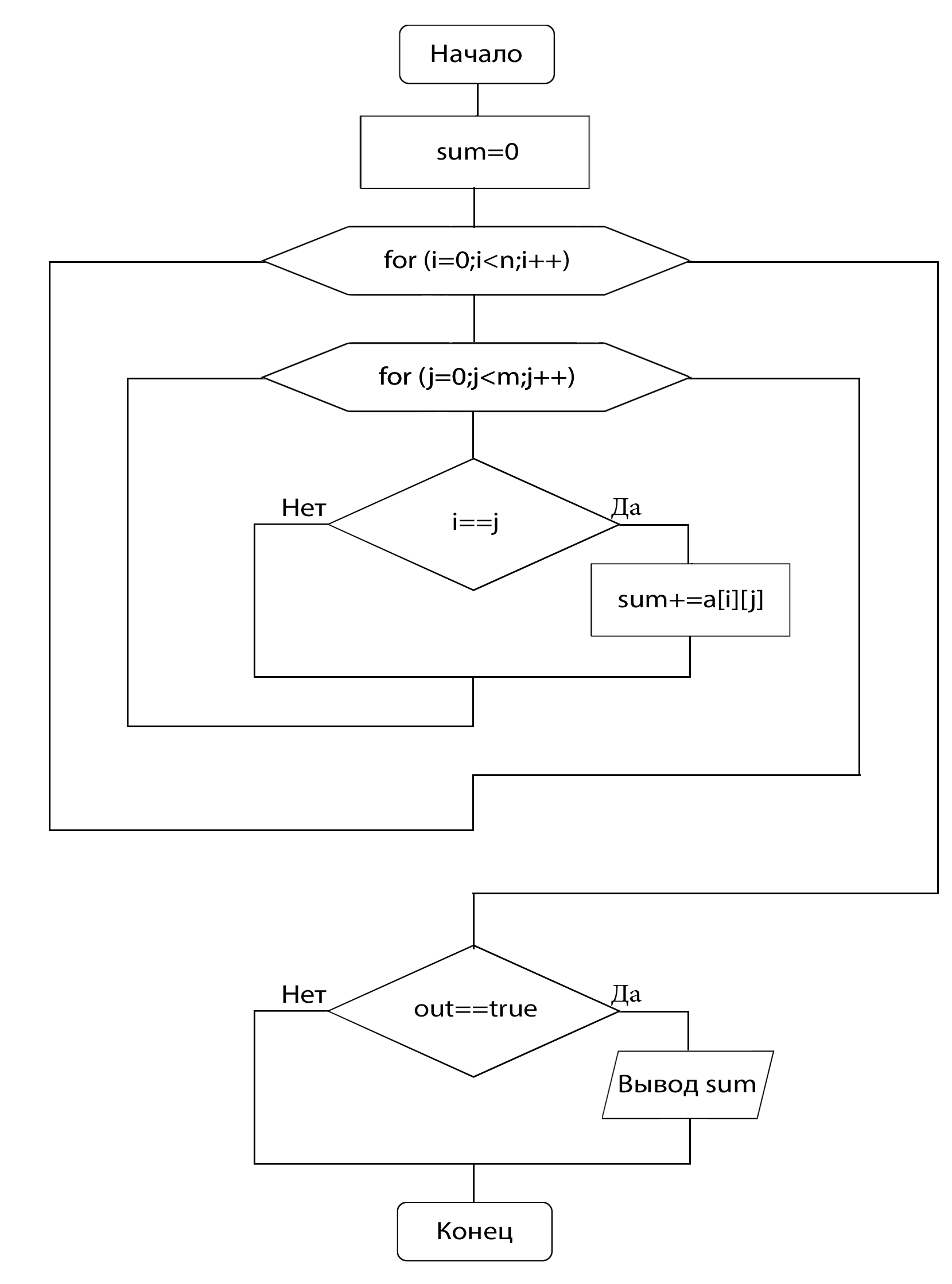
Алгоритм процедуры file\_out:

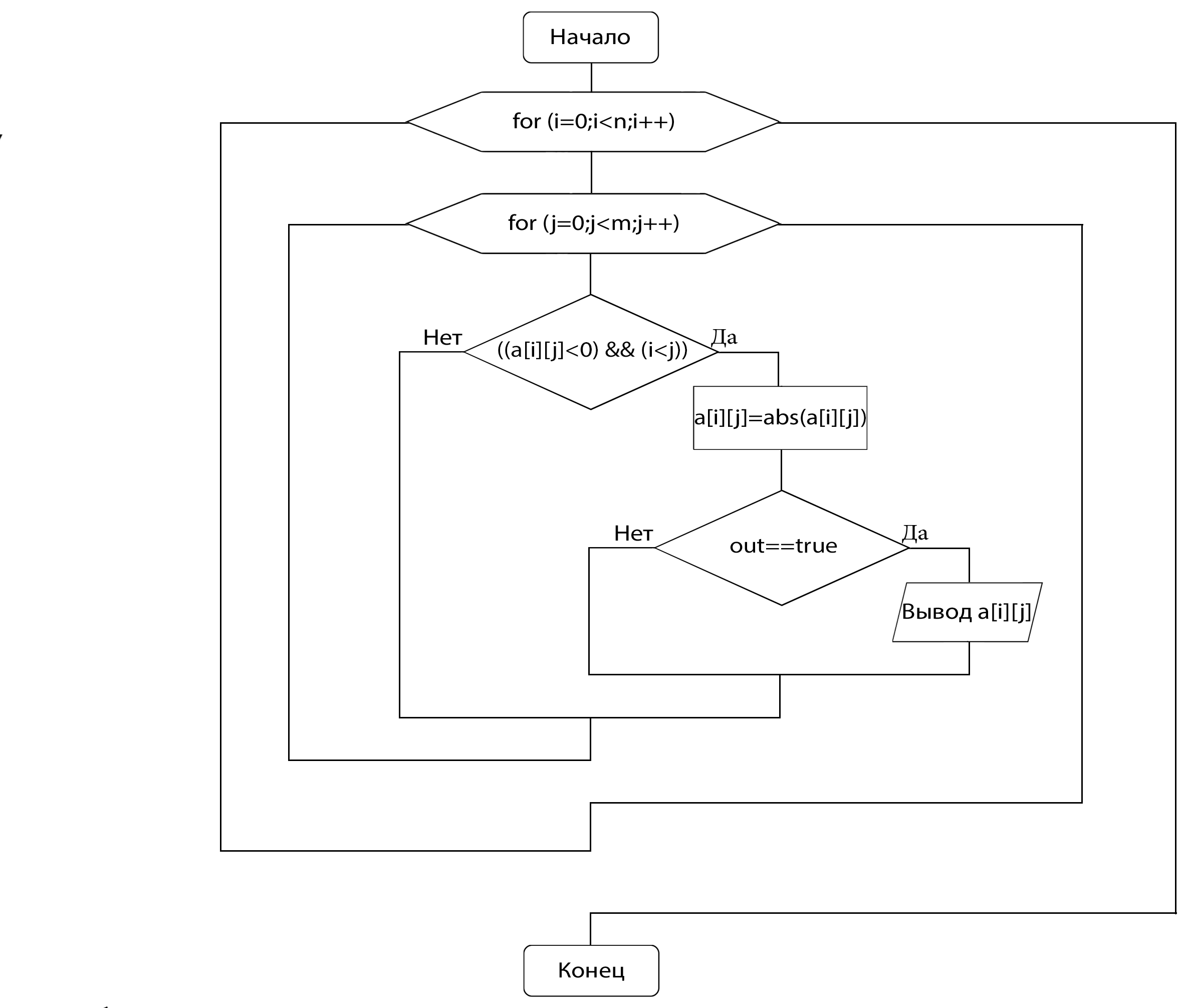


Алгоритм процедуры monitor\_out:

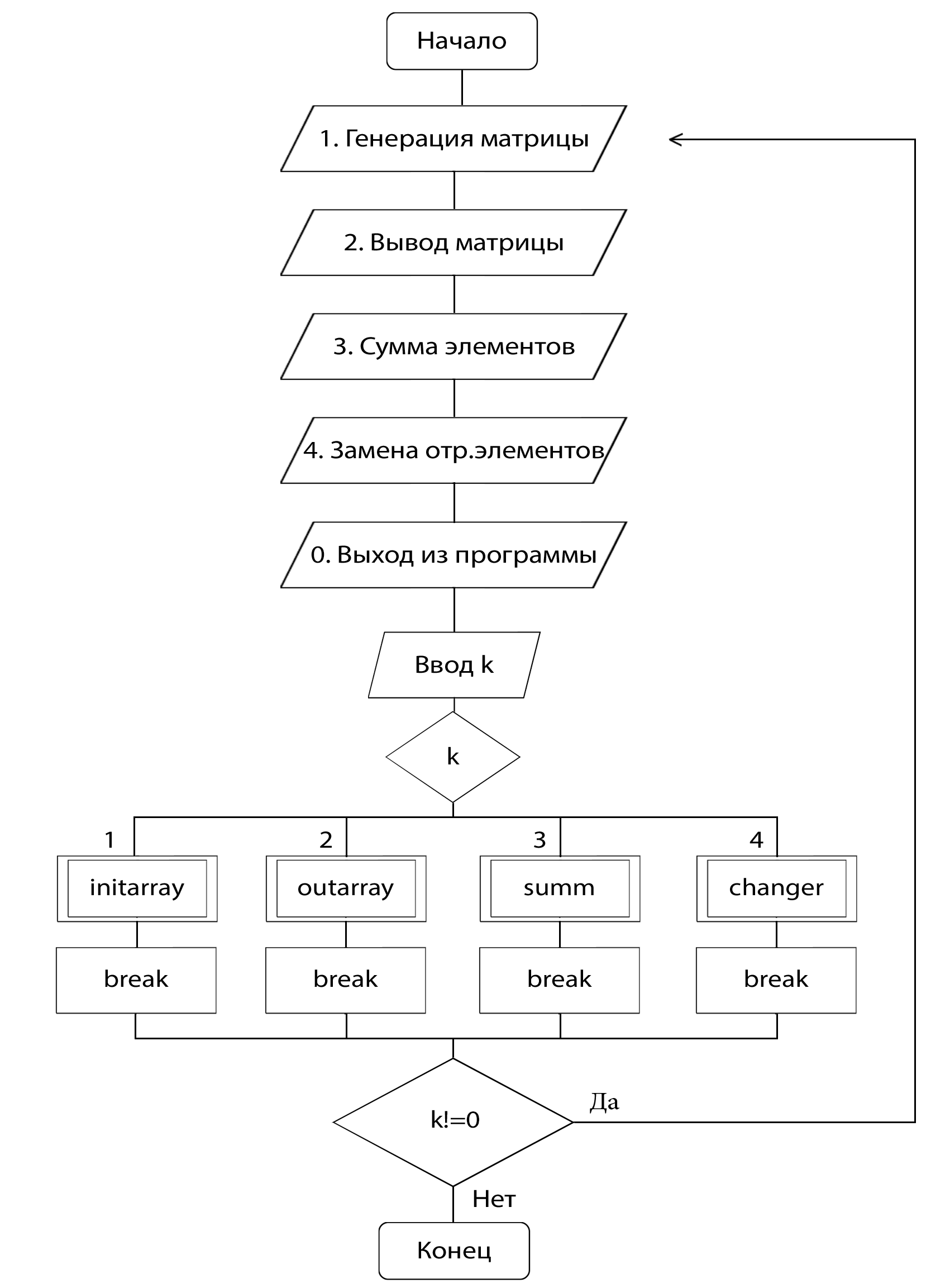


Алгоритм процедуры summ:



Алгоритм процедуры changer:

Алгоритм основной программы:

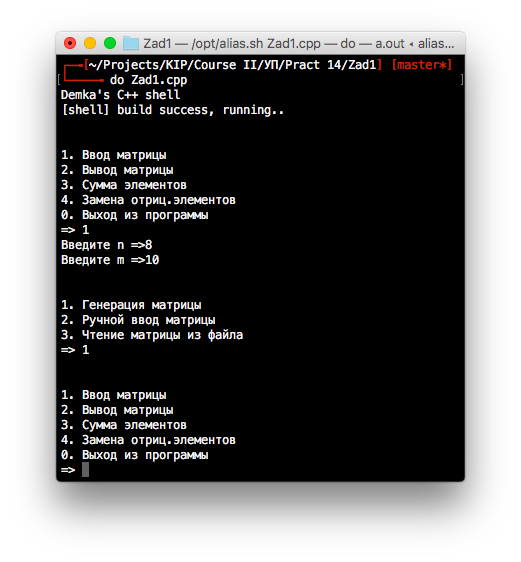


Исходный код программы:

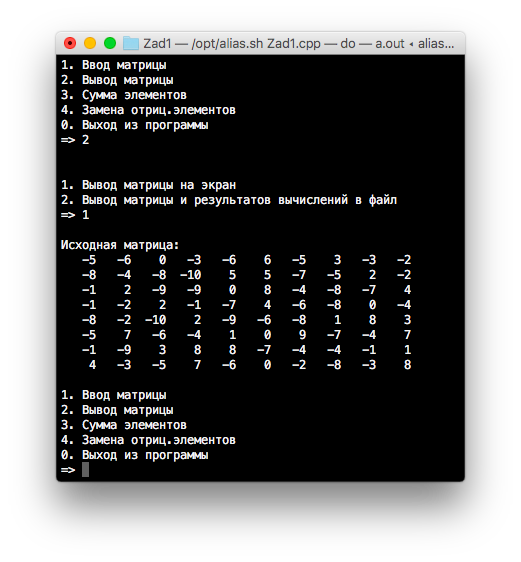
#include **<iostream>**#include **<iomanip>**#include **<fstream>  
  
using namespace** std;  
  
**int** i, j, n, m, sum, k, k1, k2, \*\*a;  
FILE \*outfile;  
  
*//Генерация матрицы***void** generate(){  
  
 srand((**unsigned**)time(0));  
 **for**(i=0;i<n;i++)  
 **for**(j=0;j<m;j++)  
 a[i][j]=-10+rand()%20;  
  
}  
  
*//Ручной ввод матрицы***void** enter(){  
  
 **for**(i=0;i<n;i++)  
 **for**(j=0;j<m;j++){  
 cout<<**"Введите элемент ["**<<i<<**"]["**<<j<<**"]: "**;  
 cin>>a[i][j];  
 }  
  
}  
  
*//Ввод матрицы из файла***void** file\_in(){  
  
 fstream fi;  
 fi.open(**"input.txt"**);  
 **for** (i=0;i<n;i++)  
 **for** (j=0;j<m;j++)  
 fi>>a[i][j];  
 fi.close();  
  
}  
  
*//Меню выбора ввода***void** initarray(){  
  
 cout<<**"Введите n =>"**; cin>>n;  
 cout<<**"Введите m =>"**; cin>>m;  
  
 a = **new int** \*[n];  
 **for**(i=0;i<n;i++)  
 a[i]=**new int**[m+1];  
  
 cout<<**"\n\n1. Генерация матрицы\n2. Ручной ввод матрицы\n3. Чтение матрицы из файла\n=> "**;  
 cin>>k1;  
 **switch** (k1)  
 {  
 **case** 1:  
 generate();  
 **break**;  
  
 **case** 2:  
 enter();  
 **break**;  
  
 **case** 3:  
 file\_in();  
 **break**;  
 }  
  
}  
  
*//Процедура вывода на экран***void** monitor\_out(){  
  
 cout<<**"\nИсходная матрица:"**;  
 **for**(i=0;i<n;i++){  
 cout<<**"\n"**;  
 **for**(j=0;j<m;j++)  
 cout<<setw(5)<<a[i][j];  
 }  
  
}  
  
*//Замена элементов***void** changer(**bool** out){  
  
 **for** (i=0;i<n;i++)  
 **for** (j=0;j<m;j++)  
   
 **if** ((a[i][j]<0) && (i<j)){  
 a[i][j]=abs(a[i][j]);  
 **if** (out==**true**)  
 cout<<**"Замена ["**<<i<<**"]["**<<j<<**"] "**<<a[i][j]<<**" на "**<<abs(a[i][j])<<**"\n"**;  
 }  
  
}  
  
*//Нахождение суммы элементов на главной диагонали***void** summ(**bool** out){  
  
 sum=0;  
 **for** (i=0;i<n;i++)  
 **for** (j=0;j<m;j++)  
 **if** (i==j)  
 sum+=a[i][j];  
 **if** (out==**true**)  
 cout<<**"Сумма элементов на главной диагонали:\n"**<<sum;  
  
}  
  
*//Процедура вывода в файл***void** file\_out(){  
  
 outfile=fopen(**"out.txt"**,**"w+"**);  
 fprintf(outfile,**"Исходная матрица:\n"**);  
 **for** (i=0;i<n;i++){  
 **for** (j=0;j<m;j++)  
 fprintf(outfile,**"%4d"**,a[i][j]);  
 fprintf(outfile,**"\n"**);  
 }  
   
 changer(**false**);  
 fprintf(outfile,**"\nПреобразованная матрица:\n"**);  
 **for** (i=0;i<n;i++){  
 **for** (j=0;j<m;j++)  
 fprintf(outfile,**"%4d"**,a[i][j]);  
 fprintf(outfile,**"\n"**);  
 }  
  
 summ(**false**);  
 fprintf(outfile,**"\nСумма элементов на главной диагонали: %d\n"**,sum);  
  
 fclose(outfile);  
  
}  
  
*//Меню выбора вывода***void** outarray(){  
  
 cout<<**"\n\n1. Вывод матрицы на экран\n2. Вывод матрицы и результатов вычислений в файл\n=> "**;  
 cin>>k2;  
 **switch** (k2)  
 {  
 **case** 1:  
 monitor\_out();  
 **break**;  
  
 **case** 2:  
 file\_out();  
 **break**;  
 }  
  
}  
  
**int** main(){  
  
 **do** {  
 cout<<**"\n\n1. Ввод матрицы\n2. Вывод матрицы\n3. Сумма элементов\n4. Замена отриц.элементов\n0. Выход из программы\n=> "**;  
 cin>>k;  
 **switch** (k)  
 {  
 **case** 1:  
 initarray();  
 **break**;  
  
 **case** 2:  
 outarray();  
 **break**;  
  
 **case** 3:  
 summ(**true**);  
 **break**;  
  
 **case** 4:  
 changer(**true**);  
 **break**;  
  
 }  
  
 }  
 **while** (k!=0);  
  
cout<<**"\n"**;  
**return** 0;  
}

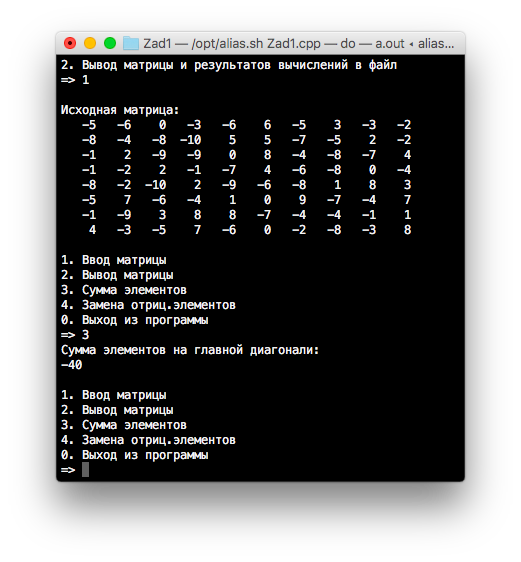
Скриншоты программы:

а) Ввод исходной матрицы через генератор случайных чисел

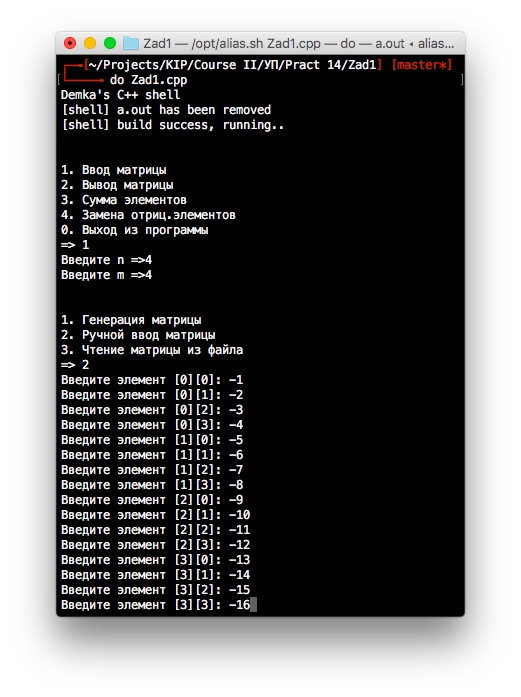


б) Вывод матрицы на экран

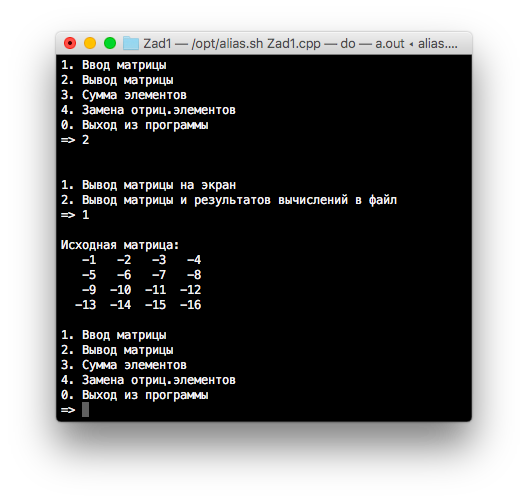


в) Вычисление суммы элементов на главной диагонали:

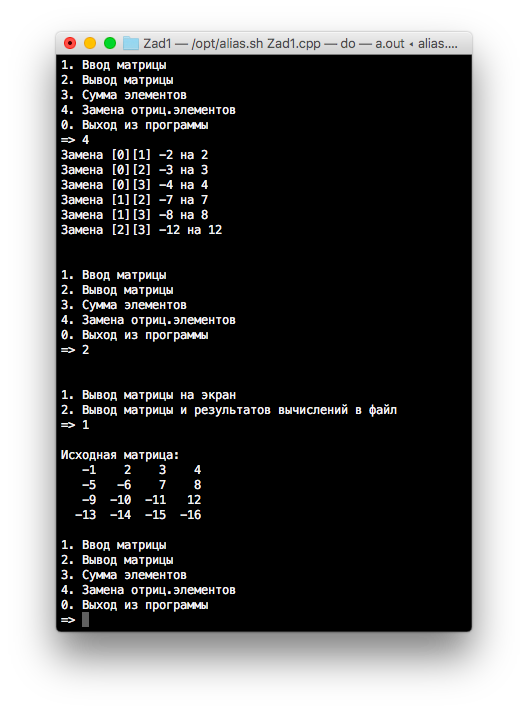
г) Ручной ввод матрицы



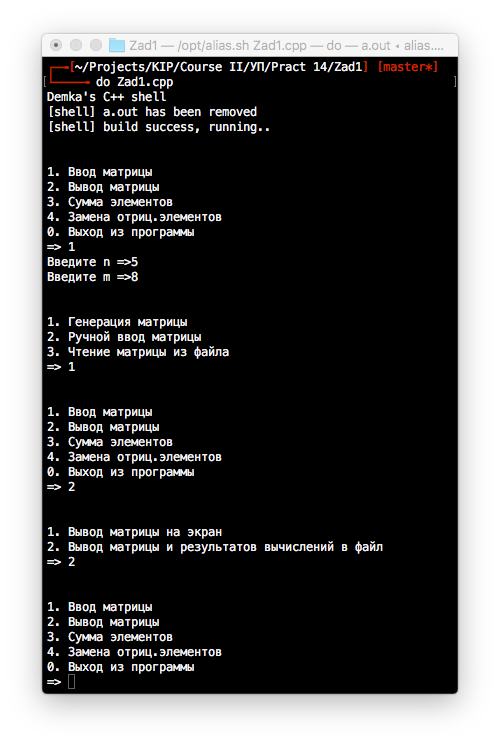
д) Вывод матрицы, введенной вручную



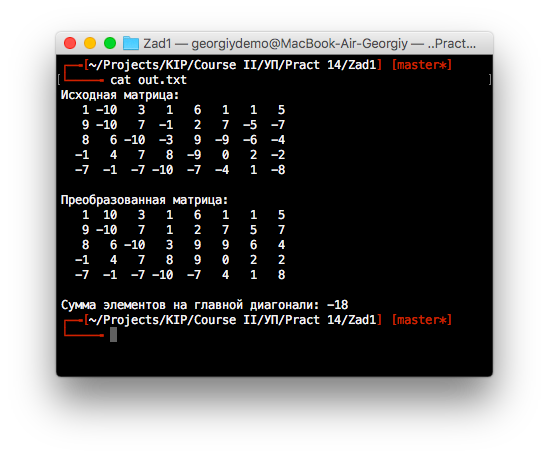
е) Замена отриц. элементов, лежащих выше главной диагонали его абсолютной величиной и вывод полученной матрицы



ё) Вывод результатов всех операций в файл



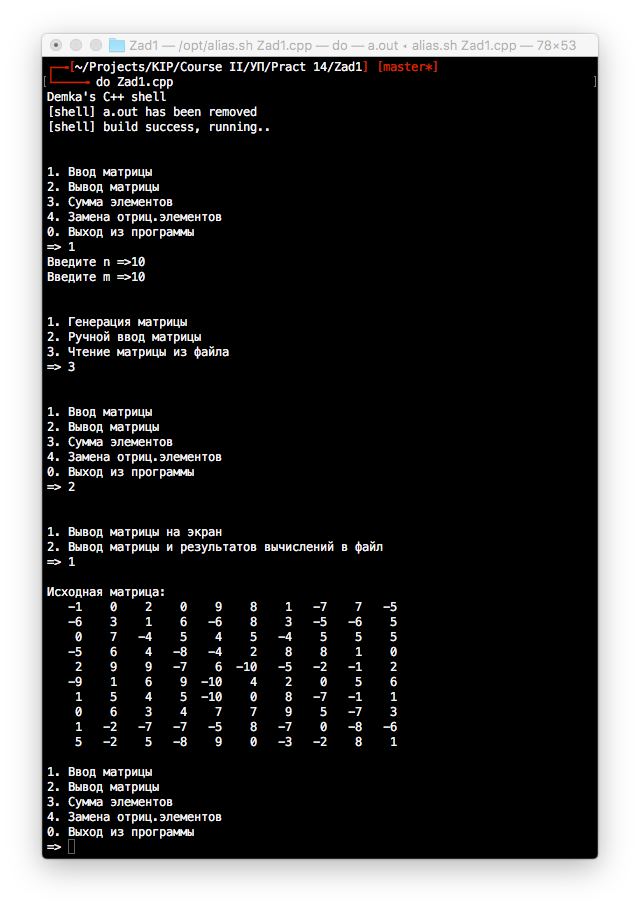
ж) Исходный файл на выходе



з) Исходный файл с матрицей



и) Чтение матрицы из файла

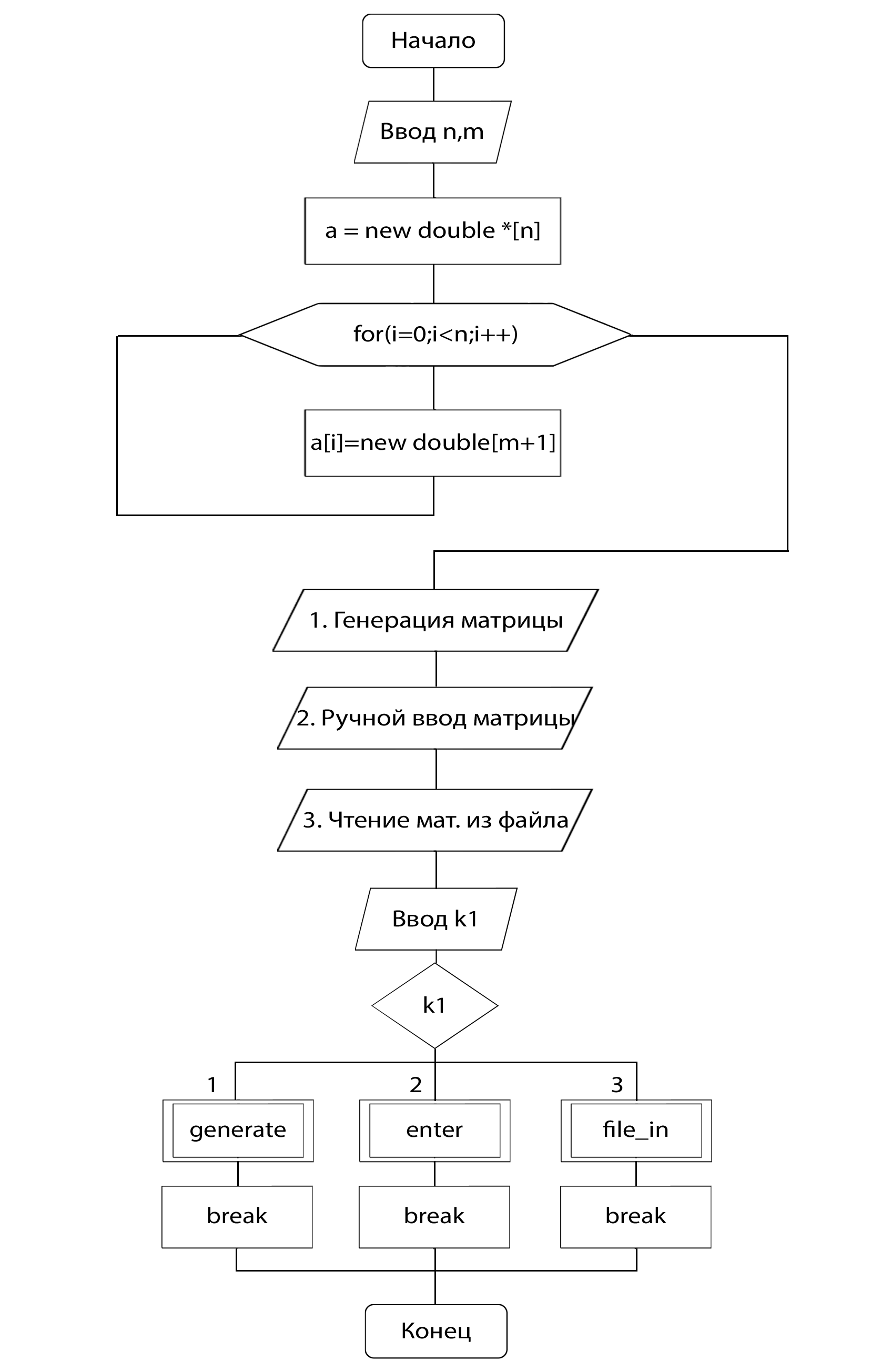


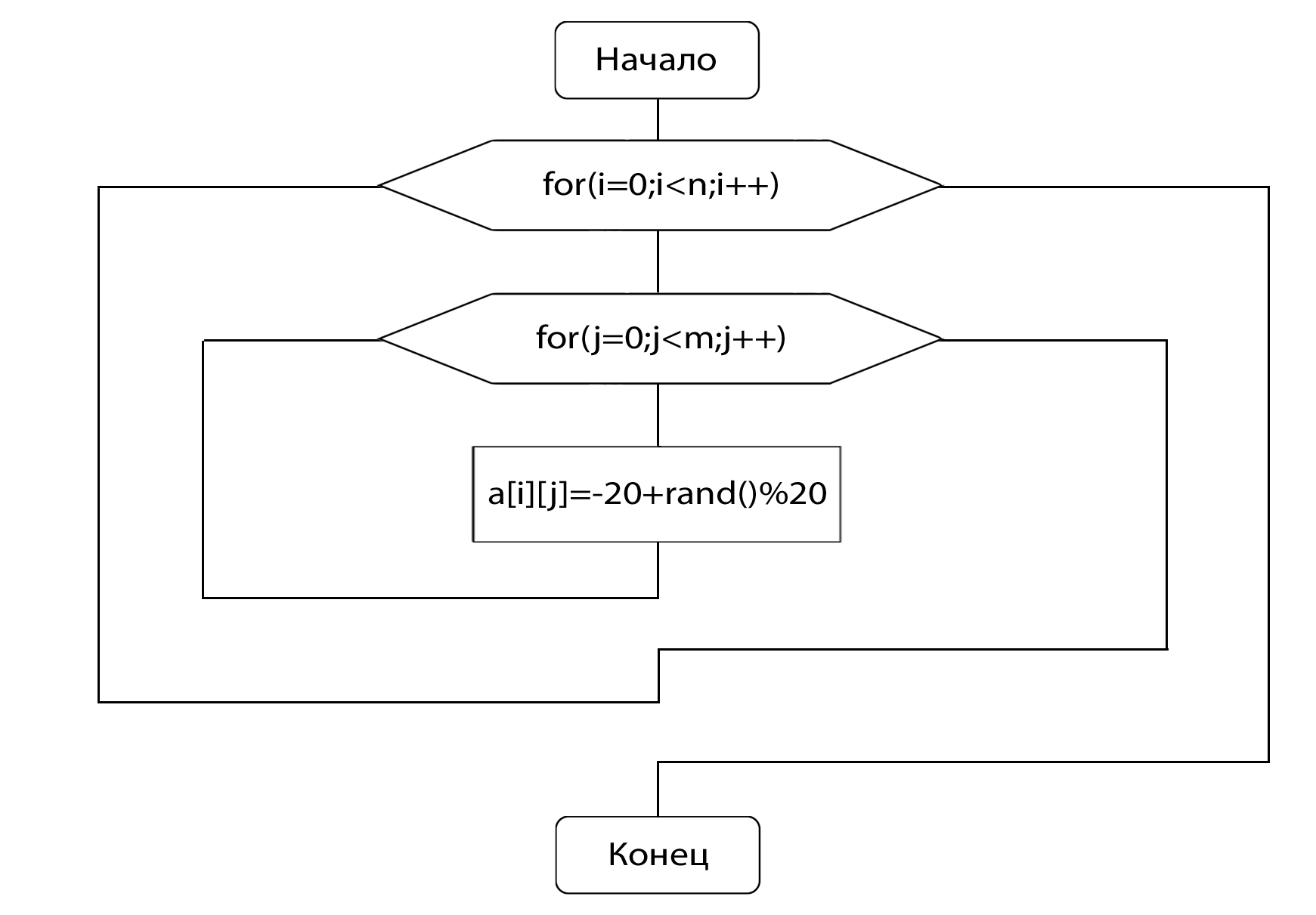
Задание №2

Условие:

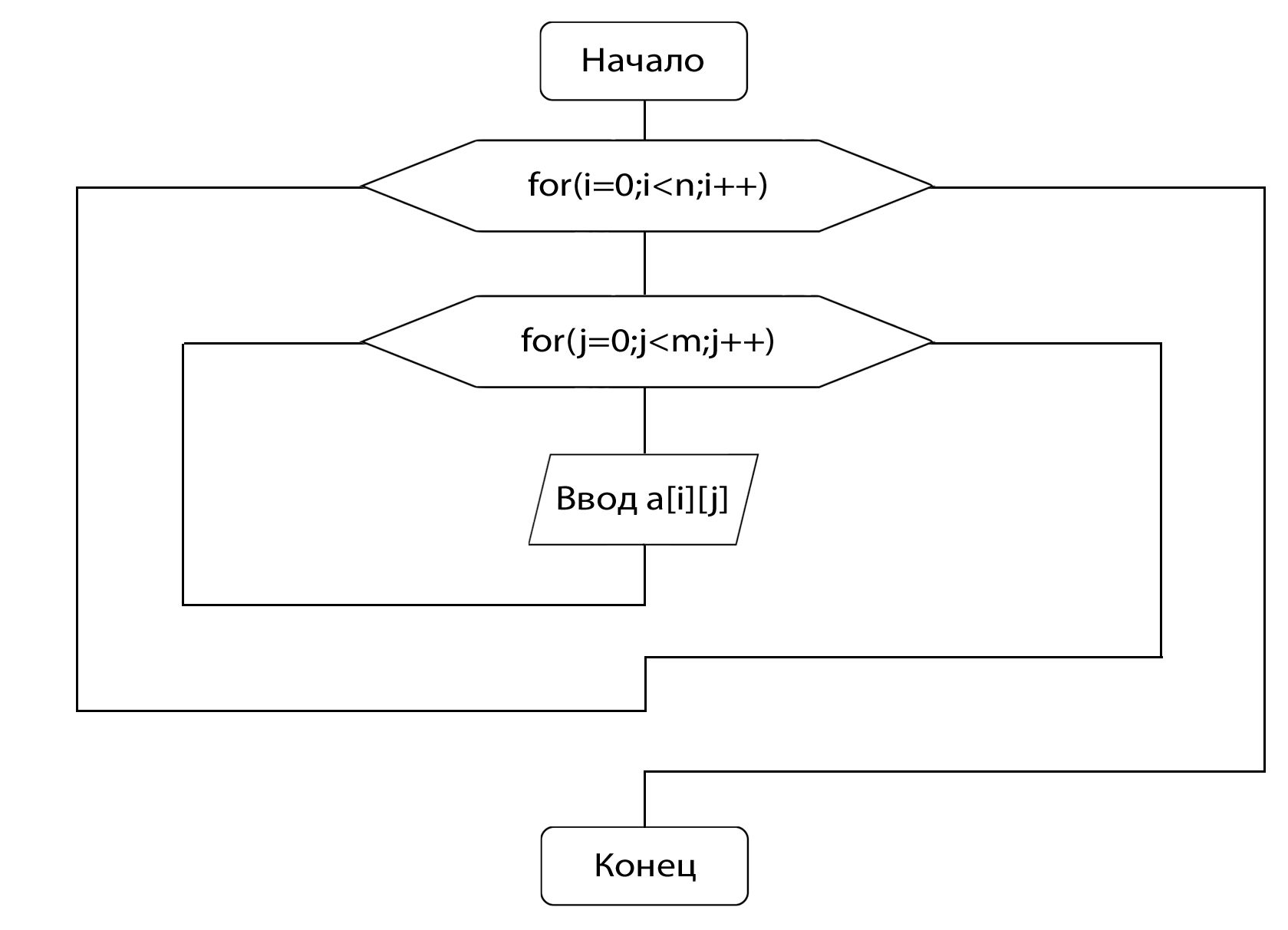
Если все элементы главной диагонали матрицы А отрицательны, разделить все элементы матрицы на максимальный по абсолютной величине элемент матрицы.

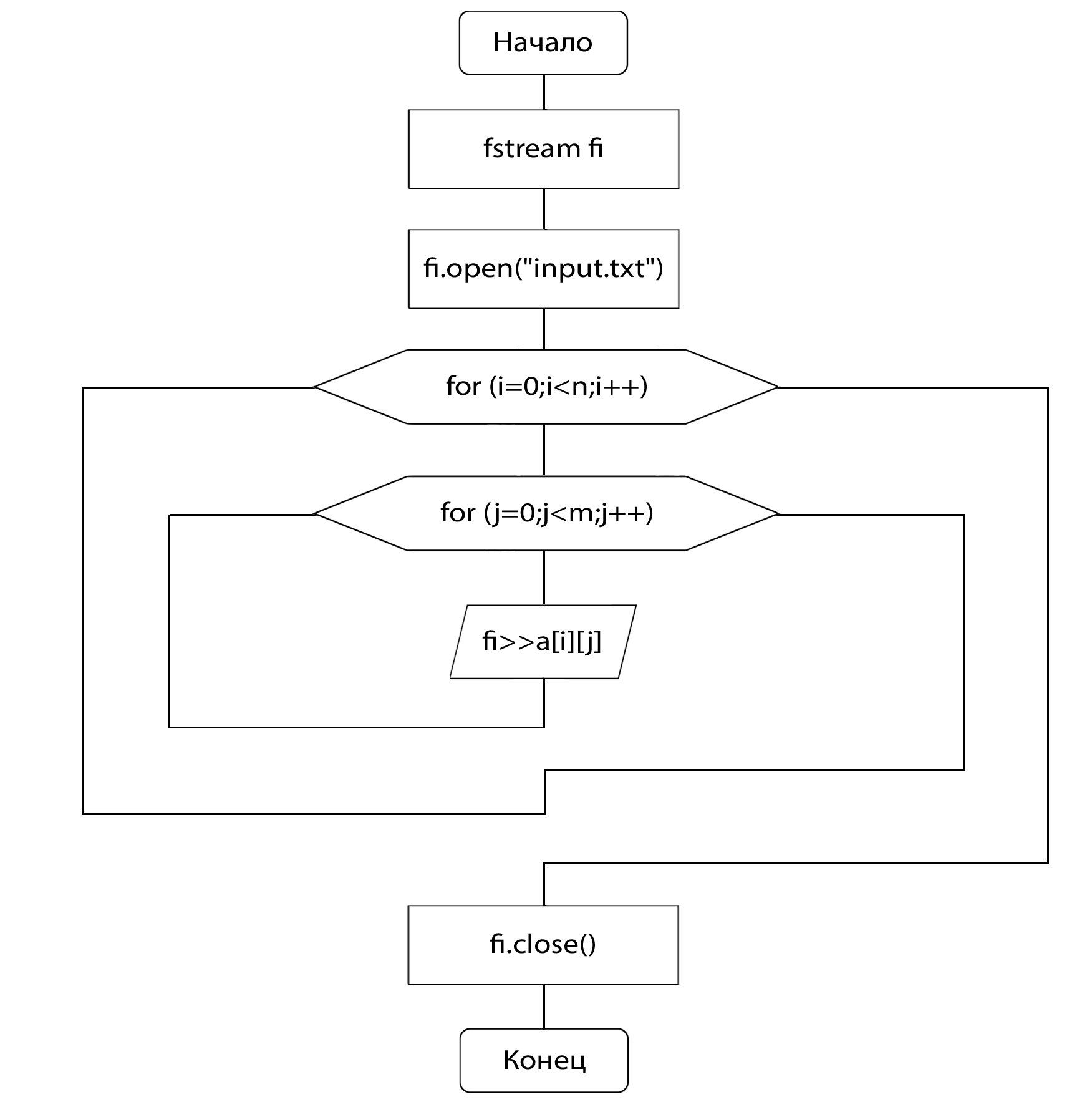
Алгоритм процедуры initarray:

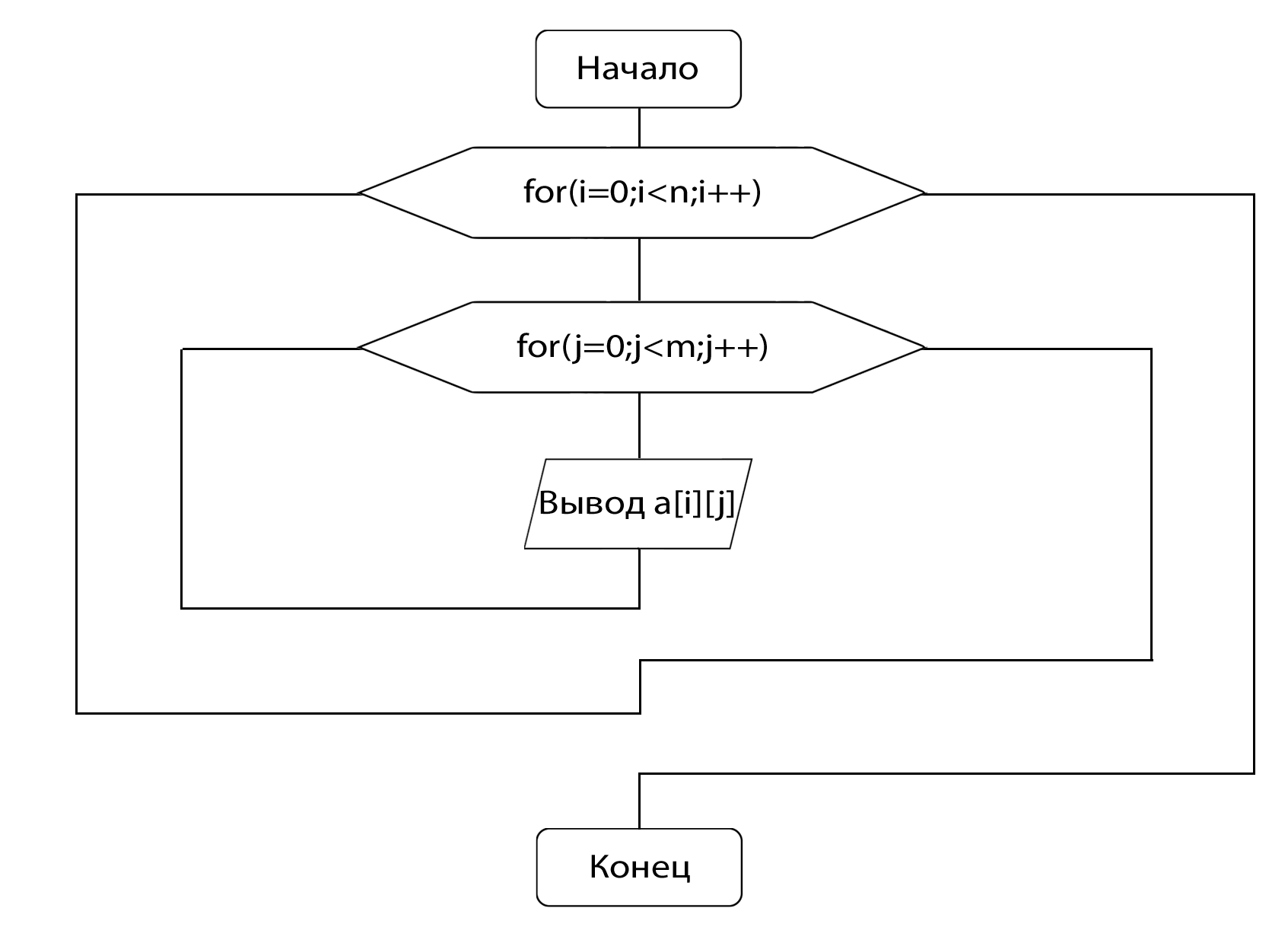


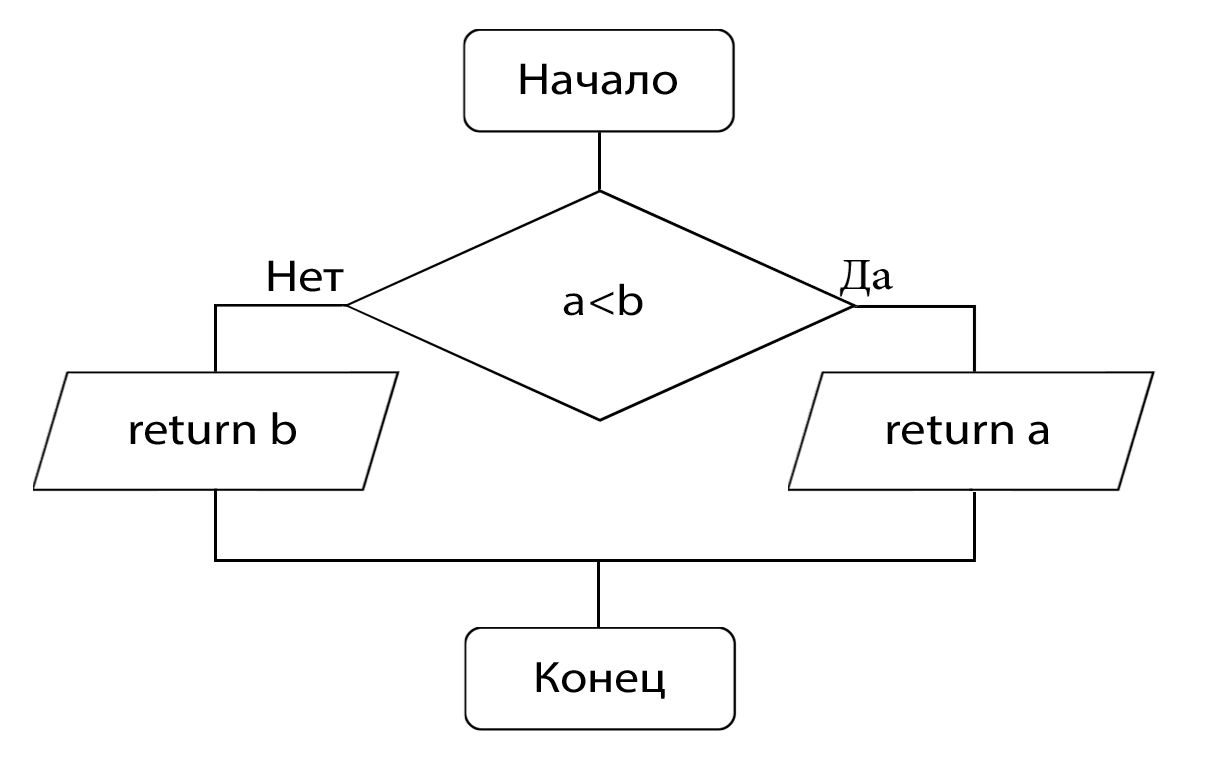
Алгоритм процедуры generate:

Алгоритм процедуры enter:

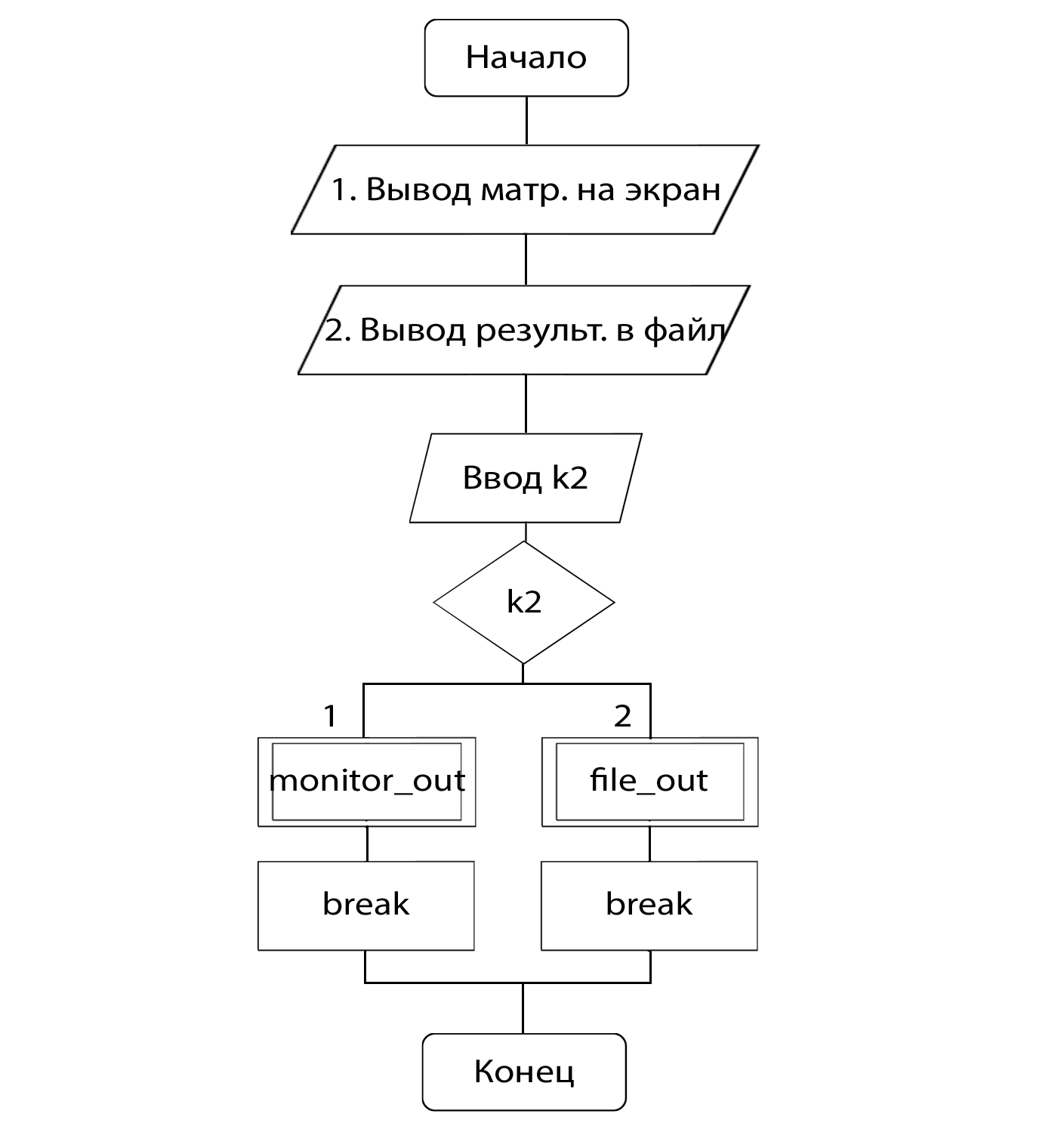


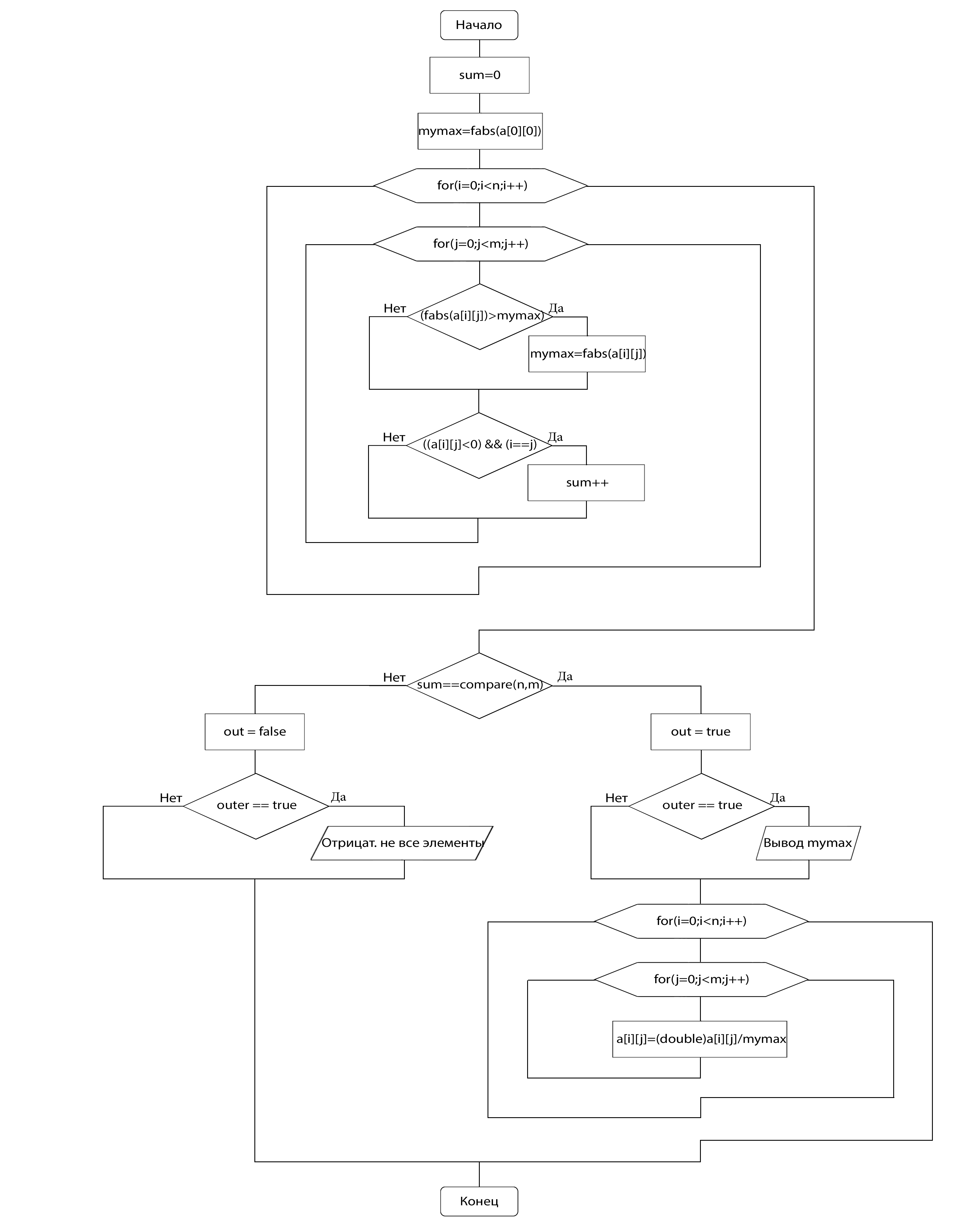
Алгоритм процедуры file\_in:

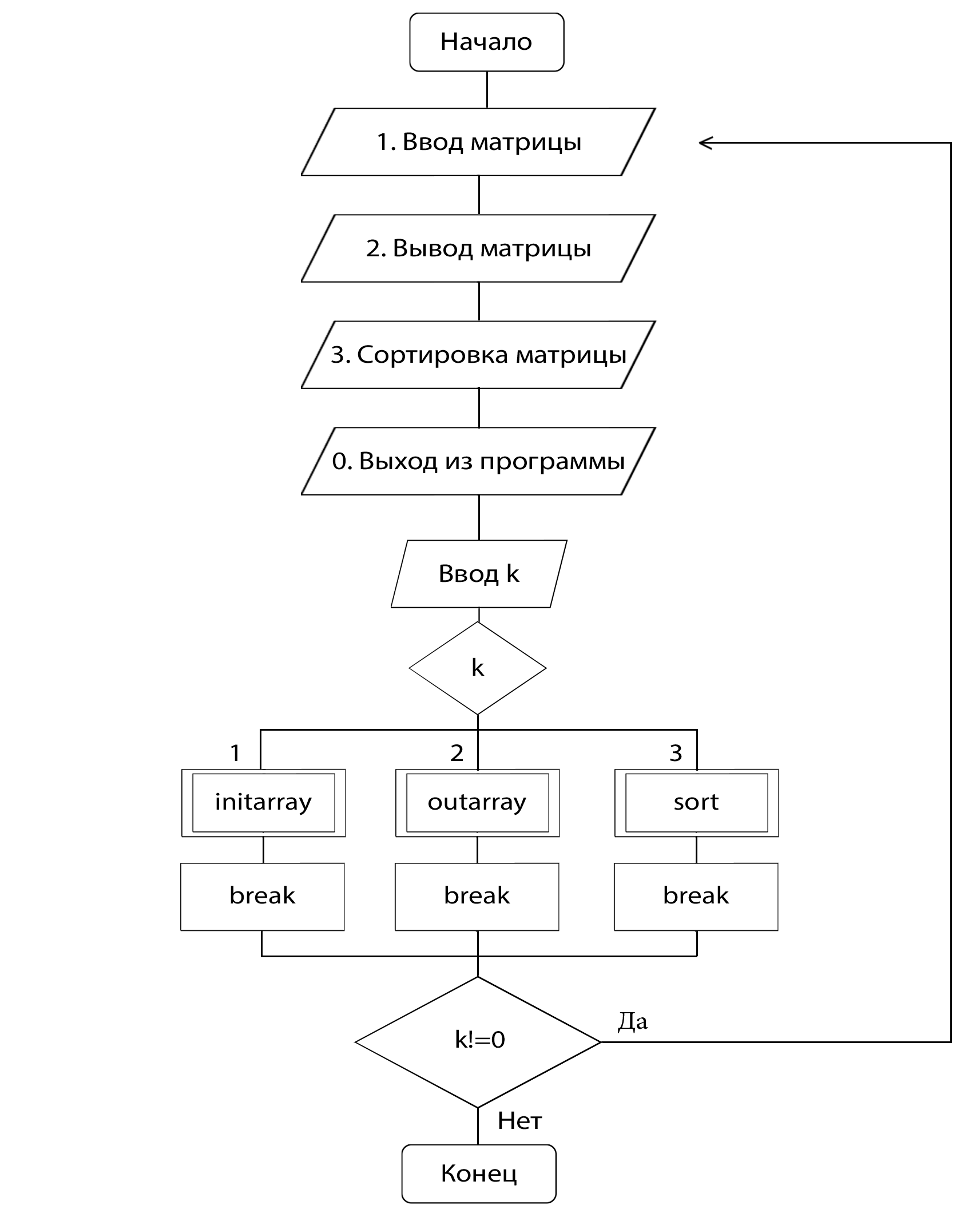
Алгоритм процедуры monitor\_out:

Алгоритм функции compare:

Алгоритм процедуры file\_out:

Алгоритм процедуры outarray:

Алгоритм процедуры sort:

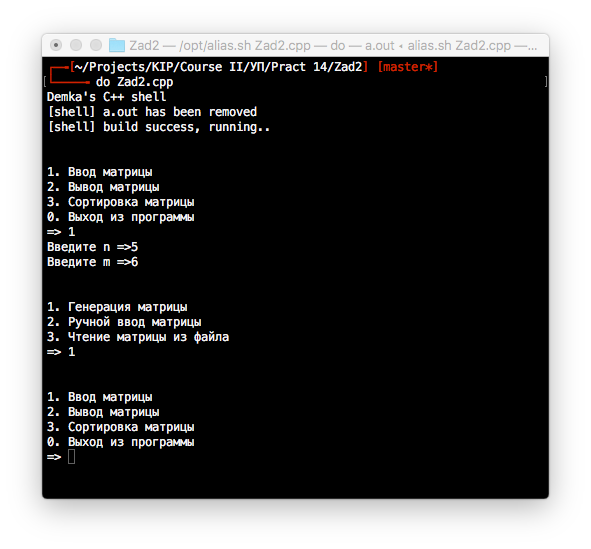
Алгоритм основной программы:

Исходный код программы:

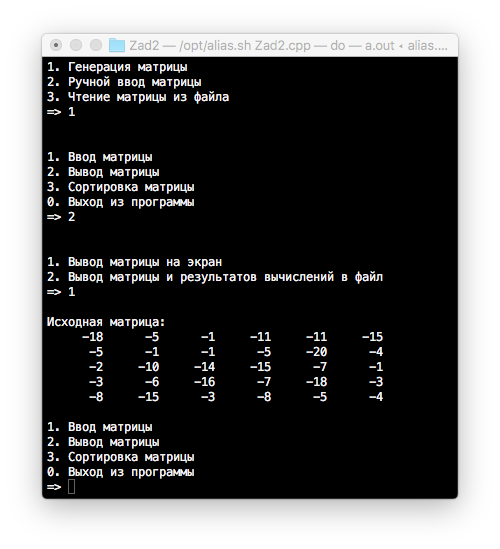
#include **<iostream>**#include **<iomanip>**#include **<fstream>**#include **<cmath>  
  
using namespace** std;  
  
**int** i, j, n, m, k, k1, k2, mymax, sum;  
**double** \*\*a;  
**bool** out;  
FILE \*outfile;  
  
*//Простая функция сравнения***int** compare(**int** a, **int** b){  
 **if** (a<b) **return** a;  
 **return** b;  
 }  
  
*//Генерация матрицы***void** generate(){  
  
 srand((**unsigned**)time(0));  
 **for**(i=0;i<n;i++)  
 **for**(j=0;j<m;j++)  
 a[i][j]=-20+rand()%20;  
}  
  
*//Ручной ввод матрицы***void** enter(){  
  
 **for**(i=0;i<n;i++)  
 **for**(j=0;j<m;j++){  
 cout<<**"Введите элемент ["**<<i<<**"]["**<<j<<**"]: "**;  
 cin>>a[i][j];  
 }  
  
}  
  
*//Ввод матрицы из файла***void** file\_in(){  
  
 fstream fi;  
 fi.open(**"input.txt"**);  
 **for** (i=0;i<n;i++)  
 **for** (j=0;j<m;j++)  
 fi>>a[i][j];  
 fi.close();  
  
}  
  
  
*//Меню выбора ввода***void** initarray(){  
  
 cout<<**"Введите n =>"**; cin>>n;  
 cout<<**"Введите m =>"**; cin>>m;  
  
 a = **new double** \*[n];  
 **for**(i=0;i<n;i++)  
 a[i]=**new double**[m+1];  
  
 cout<<**"\n\n1. Генерация матрицы\n2. Ручной ввод матрицы\n3. Чтение матрицы из файла\n=> "**;  
 cin>>k1;  
 **switch** (k1)  
 {  
 **case** 1:  
 generate();  
 **break**;  
  
 **case** 2:  
 enter();  
 **break**;  
  
 **case** 3:  
 file\_in();  
 **break**;  
 }  
  
}  
  
*//Сортировка матрицы***void** sort(**bool** outer){  
  
 sum=0;  
 mymax=fabs(a[0][0]);  
 **for**(i=0;i<n;i++)  
 **for**(j=0;j<m;j++){  
 **if** (fabs(a[i][j])>mymax)  
 mymax=fabs(a[i][j]);  
 **if** ((a[i][j]<0) && (i==j))  
 sum++;  
 }  
  
 **if** (sum==compare(n,m)){  
 out = **true**;  
 **if** (outer == **true**)  
 cout<<**"Все элементы диагонали отрицательны, макс. элемент: "**<<mymax;  
 **for**(i=0;i<n;i++)  
 **for**(j=0;j<m;j++)  
 a[i][j]=(**double**)a[i][j]/mymax;  
 }  
 **else**{  
 out = **false**;  
 **if** (outer == **true**)  
 cout<<**"Отрицательны не все элементы диагонали, завершаем работу"**;  
 }  
}  
  
*//Процедура вывода на экран***void** monitor\_out(){  
  
 cout<<**"\nИсходная матрица:"**;  
 **for**(i=0;i<n;i++){  
 cout<<**"\n"**;  
 **for**(j=0;j<m;j++)  
 cout<<setw(8)<<a[i][j];  
 }  
  
}  
  
*//Процедура вывода в файл* **void** file\_out(){  
  
 outfile=fopen(**"out.txt"**,**"w+"**);  
 fprintf(outfile,**"Исходная матрица:\n"**);  
  
 **for** (i=0;i<n;i++){  
 **for** (j=0;j<m;j++)  
 fprintf(outfile,**"%15f"**,a[i][j]);  
 fprintf(outfile,**"\n"**);  
 }  
 sort(**false**);  
 **if** (out == **true**){  
 fprintf(outfile,**"\nВсе элементы диагонали отрицательны, макс. элемент: %d\n"**,mymax);  
 fprintf(outfile,**"\nПреобразованная матрица:\n"**);  
 **for** (i=0;i<n;i++){  
 **for** (j=0;j<m;j++)  
 fprintf(outfile,**"%15f"**,a[i][j]);  
 fprintf(outfile,**"\n"**);  
 }  
 }  
 **else** fprintf(outfile,**"\nОтрицательны не все элементы диагонали, завершаем работу\n"**);  
 fclose(outfile);  
  
}  
  
*//Меню выбора вывода***void** outarray(){  
  
 cout<<**"\n\n1. Вывод матрицы на экран\n2. Вывод матрицы и результатов вычислений в файл\n=> "**;  
 cin>>k2;  
 **switch** (k2)  
 {  
 **case** 1:  
 monitor\_out();  
 **break**;  
  
 **case** 2:  
 file\_out();  
 **break**;  
 }  
  
}  
  
**int** main(){  
  
 **do** {  
 cout<<**"\n\n1. Ввод матрицы\n2. Вывод матрицы\n3. Сортировка матрицы\n0. Выход из программы\n=> "**;  
 cin>>k;  
 **switch** (k)  
 {  
 **case** 1:  
 initarray();  
 **break**;  
  
 **case** 2:  
 outarray();  
 **break**;  
  
 **case** 3:  
 sort(**true**);  
 **break**;  
  
 }  
  
 }  
 **while** (k!=0);  
  
cout<<**"\n"**;  
**return** 0;  
}

Скриншоты программы:

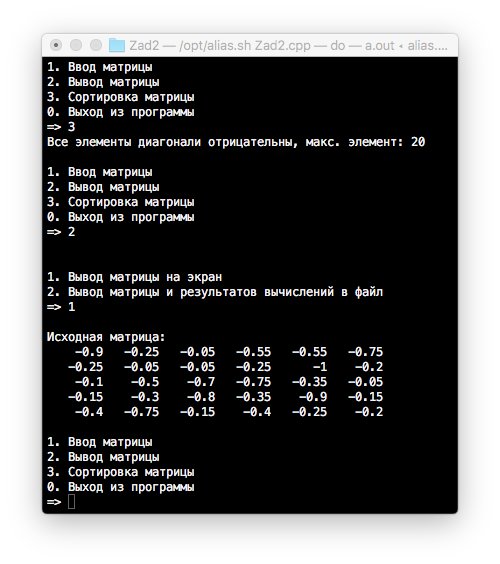
а) Ввод исходной матрицы через генератор случайных чисел



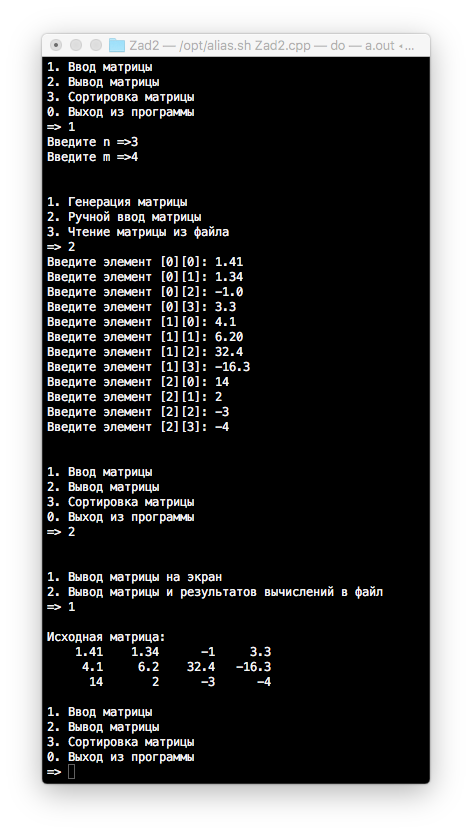
б) Вывод матрицы на экран

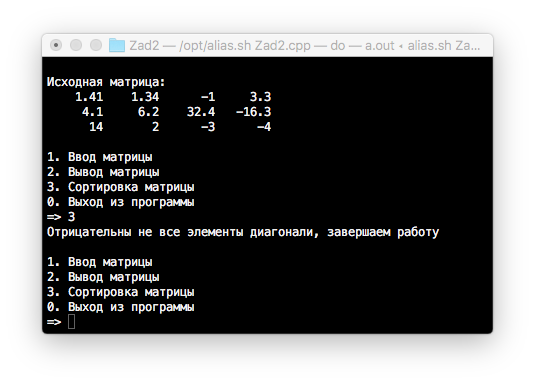


в) Если все элементы диагонали отрицательны, делим элементы матрицы на максимальный по абсолютной величине элемент

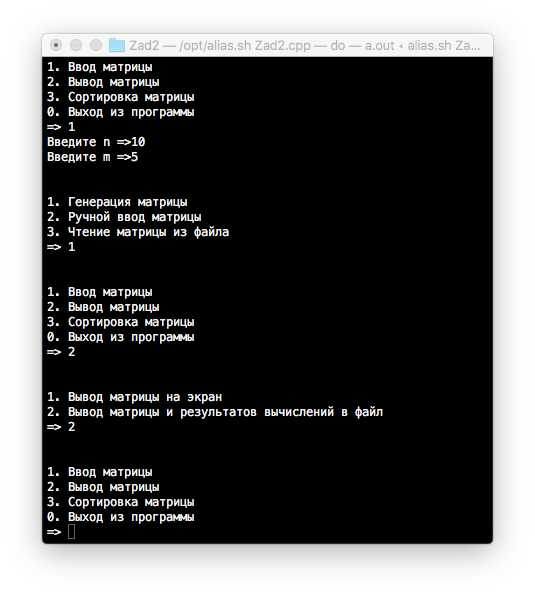


г) Ручной ввод/вывод матрицы

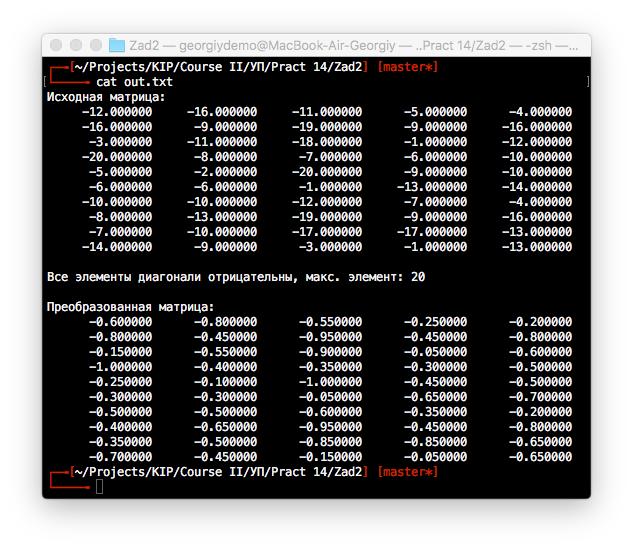


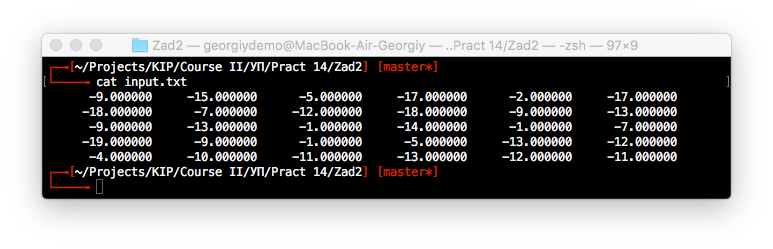
д) При выполнении условия задания не все элементы отрицательны, условие не выполняется

е) Вывод результатов всех операций в файл



ё) Исходный файл на выходе



ж) Исходный файл с матрицей

з) Чтение матрицы из файла

