**УО «Белорусский государственный университет информатики и**

**радиоэлектроники»**

**Кафедра ПОИТ**

**Отчет по лабораторной работе №2.3**

**по предмету**

**Основы Алгоритмизации и Программирования**

**Вариант 1**

**Выполнил**

**Андросов И.С.**

**Проверила**

**Данилова Г.В.**

Группа:

**8**51001

**Минск 2018**

**Задание**

Дана квадратная матрица A порядка n. Найти сумму положительных элементов матрицы, стоящих под главной диагональю.

**Код программы**

**(Delphi)**

**program** laba\_2\_3;

{$APPTYPE CONSOLE}

**uses**

SysUtils;

**type**

TShortIntMatrix = **array of array of** ShortInt;

**const**

MinNumber: ShortInt = -128;

MaxNumber: ShortInt = 127;

MinSize: ShortInt = 2;

MaxSize: ShortInt = 127;

**procedure** GetOutputToConsole(Sum: Integer);

**begin**

WriteLn('The sum is:', Sum);

**end**;

**function** GetAnswer(): Char;

**var**

Answer: Char;

IsCorrect: Boolean;

**begin**

**repeat**

ReadLn(Answer);

Answer := UpCase(Answer);

**if** (Answer = 'Y') **or** (Answer = 'N') **then**

IsCorrect := true

**else**

**begin**

IsCorrect := false;

WriteLn('Incorrect input. Enter Y(Yes) or N(No):');

**end**;

**until** IsCorrect;

GetAnswer := Answer;

**end**;

**procedure** GetOutputToFile(Sum: Integer);

**var**

IsCorrect: Boolean;

NewFile: TextFile;

NameOfFile: String;

**begin**

WriteLn('Enter the name of file Name.txt:');

IsCorrect := false;

**repeat**

ReadLn(NameOfFile);

**if** (**not** FileExists(NameOfFile)) **then**

WriteLn('File does not exist. Try again:')

**else**

**begin**

IsCorrect := true;

Assign(NewFile, NameOfFile);

writeln('Would you like to rewrite the file? Press Y(Yes) or N(No):');

**if** GetAnswer = 'Y' **then**

**try**

Rewrite(NewFile);

WriteLn(NewFile, 'The sum is:', Sum);

**except**

WriteLn('Access is not allowed. Try again:');

IsCorrect:= false;

**end**

**else**

**try**

Append(NewFile);

WriteLn(NewFile, 'The sum is:', Sum);

**except**

WriteLn('Access is not allowed. Try again:');

IsCorrect:= false;

**end**;

**end**;

**until** IsCorrect;

Close(NewFile);

**end**;

**function** GetSum(Matrix: TShortIntMatrix; Size: ShortInt): Integer;

**var**

Sum: Integer;

i, j: ShortInt;

**begin**

Sum := 0;

**for** j := 0 **to** Size **do**

**for** i := j + 1 **to** Size **do**

**if** (Matrix[i, j] > 0) **then**

Sum := Sum + Matrix[i, j];

GetSum := Sum;

**end**;

**function** CheckInput(Min, Max: ShortInt): ShortInt;

**var**

IsCorrect: Boolean;

Number: ShortInt;

**begin**

IsCorrect := false;

**repeat**

**try**

ReadLn(Number);

**if** (Number >= min) **and** (Number <= max) **then**

IsCorrect := true

**else**

WriteLn('Enter number from interval ', Min, '..', Max, ':');

**except**

WriteLn('Check entered data. Enter number from interval ', Min, '..', Max, ':');

**end**;

**until** IsCorrect;

CheckInput := Number;

**end**;

**function** GetMatrixConsole(**var** Size: ShortInt): TShortIntMatrix;

**var**

i, j: ShortInt;

Matrix: TShortIntMatrix;

IsCorrect: Boolean;

**begin**

WriteLn('Enter size of matrix ', MinSize, '..', MaxSize, ':');

Size := CheckInput(MinSize, MaxSize);

SetLength(Matrix, Size, Size);

Dec(Size);

**for** i := 0 **to** Size **do**

**for** j := 0 **to** Size **do**

**begin**

WriteLn('Enter [', i, '][', j, '] number of matrix ', MinNumber, '..',

MaxNumber, ':');

Matrix[i, j] := CheckInput(MinNumber, MaxNumber);

**end**;

GetMatrixConsole := Matrix;

**end**;

**function** CheckInputFile(**const** MinNumber, MaxNumber: ShortInt; **var** MyFile: TextFile): Boolean;

**var**

IsCorrect: boolean;

Number: ShortInt;

**begin**

IsCorrect := true;

**while** (**not** SeekEof(MyFile)) **and** (IsCorrect) **do**

**try**

ReadLn(MyFile, Number);

**if** (Number <= MinNumber) **or** (Number >= MaxNumber) **then**

IsCorrect := false;

**except**

IsCorrect := false;

**end**;

CheckInputFile := IsCorrect;

**end**;

**function** ReadFromFile(**var** MyFile: TextFile; **var** Matrix: TShortIntMatrix; **var** Size: ShortInt): Boolean;

**var**

Number, i, j: ShortInt;

**begin**

**if** CheckInputFile(MinNumber, MaxNumber, MyFile) **then**

**begin**

reset(MyFile);

i := 0;

**while not** Eof(MyFile) **do**

**begin**

SetLength(Matrix, i + 1);

j := 0;

**while not** Eoln(MyFile) **do**

**begin**

SetLength(Matrix[i], j + 1);

Read(MyFile,Matrix[i,j]);

WriteLn('Elemet [', i, '][', j, ']:', Matrix[i,j]);

inc(j);

**end**;

ReadLn(MyFile);

inc(i);

**end**;

**if** (i = j) **then**

**begin**

Size := i - 1;

ReadFromFile := true;

**end**

**else**

**begin**

WriteLn('This is not a square matrix. Try again:');

ReadFromFile := false;

**end**;

**end**

**else**

**begin**

CloseFile(MyFile);

ReadFromFile := false;

WriteLn('Check entered data. Enter number from interval ', MinNumber, '..',

MaxNumber, '. Try Again:');

**end**;

**end**;

**function** GetMatrixFile(**var** Size: ShortInt): TShortIntMatrix;

**var**

IsCorrect: Boolean;

NameOfFile: String;

MyFile: TextFile;

Matrix: TShortIntMatrix;

**begin**

WriteLn('Enter file name(Name.txt):');

**repeat**

ReadLn(NameOfFile);

**if** (**not** FileExists(NameOfFile)) **then**

**begin**

WriteLn('File does not exist. Try again:');

IsCorrect := false;

**end**

**else**

**begin**

AssignFile(MyFile, NameOfFile);

reset(MyFile);

**if** SeekEof(MyFile) **then**

**begin**

WriteLn('File is empty. Try again:');

IsCorrect := false;

**end**

**else**

IsCorrect := ReadFromFile(MyFile, Matrix, Size);

**end**;

**until** IsCorrect;

CloseFile(MyFile);

GetMatrixFile := Matrix;

**end**;

**procedure** Main();

**var**

Matrix: TShortIntMatrix;

Size: ShortInt;

Sum: Integer;

**begin**

WriteLn('This program calculates the sum of positive elements of a square matrix under

the main diagonal.', #10, 'Would you like to open the file? Press Y(Yes) or

N(No):');

**if** GetAnswer = 'Y' **then**

Matrix := GetMatrixFile(Size)

**else**

Matrix := GetMatrixConsole(Size);

Sum := GetSum(Matrix, Size);

WriteLn('Would you like to write down the answer to file? Press Y(Yes) or N(No):');

**if** GetAnswer = 'Y' **then**

GetOutputToFile(Sum)

**else**

GetOutputToConsole(Sum);

WriteLn('Press "Enter" to exit the console.');

ReadLn;

**end**;

**begin**

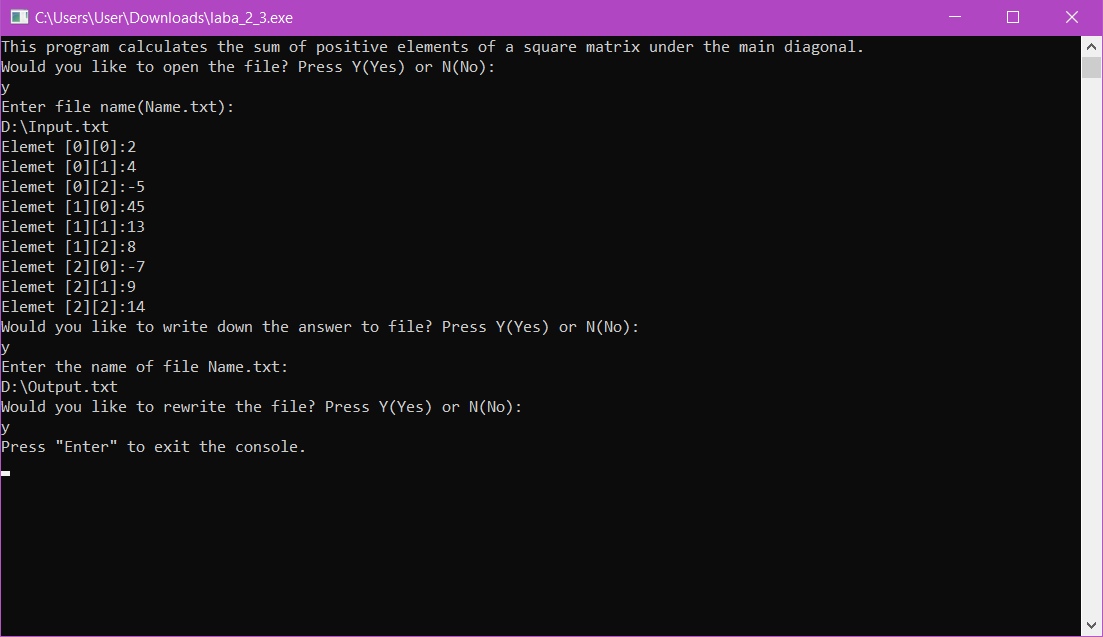
Main();

**end**.

**(С)**

**Скриншоты**

**(Delphi)**



**(С++)**

**Схема алгоритма**





