Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по лабораторной работе № 1

Тема работы: Действия над матрицами

Выполнил

студент: гр.551004 Довыдёнок М.А.

Проверил: Фадеева Е.П.

Минск 2016

Содержание

[1 Постановка задачи 3](#_Toc445836552)

[2 Описание алгоритмов 4](#_Toc445836553)

[3 Основные расчетные формулы 5](#_Toc445836554)

[4 Структура данных 6](#_Toc445836555)

[4.1 Структура данных основной программы 6](#_Toc445836556)

[4.2 Структура данных подпрограмм 6](#_Toc445836557)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 8](#_Toc445836558)

[6 Результаты расчетов и тестирование программы 10](#_Toc445836559)

[Приложение А 11](#_Toc445836560)

# Постановка задачи

Разработать программу для расчета матричного выражения:

2\*(A+B)\*(2\*B-A) , где

# Описание алгоритмов

Таблица 1 – Описание основной программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование  алгоритма | Назначение  алгоритма | Формальные  параметры | Рекомендуемый тип |
|  | Основной алгоритм | Вызов необходимых подпрограмм:  inputAB, productMM, productNM, addMM, ouputMatrix |  |  |
|  | inputAB(a, b) | Заполнение матриц исходными данными | a, b – array [1..3, 1..3] of Real. Возвращаемые параметры: a, b | Процедура |
|  | productMM(a, b, result) | Возвращает матрицу result, равную произведению матриц a и b | a, b, result – array [1..3, 1..3] of Real.  Возвращаемый параметр: result | Функция.  Возвращаемый параметр: result |
|  | productNM(num, a, result) | Возвращает матрицу result, равную произведению числа num и матрицы a | num - Real, a, result – array [1..3, 1..3] of Real.  Возвращаемый параметр: result | Функция.  Возвращаемый параметр: result |
|  | addMM(a, b, sign, result) | Возвращает матрицу result, равную сумме матрицы a и матрицы b, умноженной на число sign | a, b – array [1..3, 1..3] of Real., sign - Integer, result – array [1..3, 1..3] of Real.  Возвращаемый параметр: result | Функция.  Возвращаемый параметр: result |
|  | outputMatrix(note, a) | Выводит строку note, а затем матрицу a | note – String, a - array [1..3, 1..3] of Real | Процедура |

# Основные расчетные формулы

Умножение матрицы на число:

Сложение матриц:

Вычитание матриц:

Произведение матриц:

# Структура данных

## Структура данных основной программы

Таблица 2 – Структура данных основной программы

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| a | Array [1..3,1..3] of Real | Массив элементов первой матрицы |
| b | Array [1..3,1..3] of Real | Массив элементов второй матрицы |
| c | Array [1..3,1..3] of Real | Массив элементов матрицы результата |

## Структура данных подпрограмм

Таблица 4 – Структура данных алгоритма productMM(a, b, result)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| a | Array [1..3,1..3] of Real | Массив элементов первой матрицы |
| b | Array [1..3,1..3] of Real | Массив элементов второй матрицы |
| result | Array [1..3,1..3] of Real | Массив элементов матрицы результата |
| i, j, k | Integer | Счетчики циклов |

Таблица 5 – Структура данных алгоритма productNM(num, a, result)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| num | Real | Первый множитель |
| a | Array [1..3,1..3] of Real | Массив элементов матрицы, второй множитель |
| result | Array [1..3,1..3] of Real | Массив элементов результирующей матрицы |
| i, j | Integer | Счетчики циклов |

Таблица 6 – Структура данных алгоритма addMM(a, b, sign, result)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| a | Array [1..3,1..3] of Real | Массив элементов первой матрицы |
| b | Array [1..3,1..3] of Real | Массив элементов второй матрицы |
| sign | Integer | Число, отвечающее за знак второй матриц |
| result | Array [1..3,1..3] of Real | Массив элементов результирующей матрицы |
| i, j | Integer | Счетчики циклов |

Таблица 7 – Структура данных алгоритма outputMatrix(note, a)

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| note | String | Строка, выводимая до вывода матрицы |
| a | Array [1..3,1..3] of Real | Массив элементов матрицы |
| i, j | Integer | Счетчики циклов |

# Схема алгоритма решения задачи по ГОСТ 19.701-90

|  |  |
| --- | --- |
| Схема основного алгоритма | Схема алгоритма inputAB(a, b) |
|  |  |

|  |  |
| --- | --- |
| Схема алгоритма  productMM(a, b, result) | Схема алгоритма  productNM(num, a, result) |
|  |  |

|  |  |
| --- | --- |
| Схема алгоритма  addMM(a, b, sign, result) | Схема алгоритма  outputMatrix(note, a) |
|  |  |

# Результаты расчетов и тестирование программы

Ожидаемый результат:

Рисунок 2 – Расчёт в MathCad

Полученный результат:

Рисунок 1 – Результат работы программы

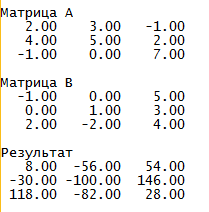


Рисунок 1 – Результат работы программы

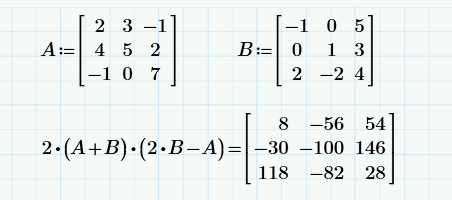


Рисунок 2 – Расчет в MathCad

Приложение А

(обязательное)

Исходный код программы

program L9;

{$APPTYPE CONSOLE}

uses

Windows;

type

TMatrix = array [1..3, 1..3] of real;

var

a, b: TMatrix;

{ Ввод матриц А и B }

procedure inputAB(var a, b: TMatrix);

begin

a[1, 1] := 2;

a[1, 2] := 3;

a[1, 3] := -1;

a[2, 1] := 4;

a[2, 2] := 5;

a[2, 3] := 2;

a[3, 1] := -1;

a[3, 2] := 0;

a[3, 3] := 7;

b[1, 1] := -1;

b[1, 2] := 0;

b[1, 3] := 5;

b[2, 1] := 0;

b[2, 2] := 1;

b[2, 3] := 3;

b[3, 1] := 2;

b[3, 2] := -2;

b[3, 3] := 4;

end;

{ Произведение матриц }

function productMM(const a, b: TMatrix):TMatrix;

var

i, j, k: integer;

begin

for i := 1 to 3 do

begin

for j := 1 to 3 do

begin

result[i, j] := 0;

for k := 1 to 3 do

begin

result[i, j] := result[i, j] + a[i, k] \* b[k, j];

end;

end;

end;

end;

{ Произведение матрицы и числа }

function productNM(const num: Real;

const a: TMatrix):TMatrix;

var

i, j: integer;

begin

for i := 1 to 3 do

begin

for j := 1 to 3 do

begin

result[i, j] := num \* a[i, j];

end;

end;

end;

{ Сумма матриц }

function addMM(const a, b: TMatrix;

const sign: integer):TMatrix;

var

i, j: integer;

begin

for i := 1 to 3 do

begin

for j := 1 to 3 do

begin

result[i, j] := a[i, j] + sign \* b[i, j];

end;

end;

end;

{ Вывод матрицы }

procedure outputMatrix(const note: String;

const a: TMatrix);

var

i, j: integer;

begin

writeln(note);

for i := 1 to 3 do

begin

for j := 1 to 3 do

begin

write(a[i, j]:7:2, ' ');

end;

writeln;

end;

writeln;

end;

begin

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

inputAB(a, b);

outputMatrix('Матрица A', a);

outputMatrix('Матрица B', b);

c := productMM(productNM(2, addMM(a, b, +1)),

addMM(productNM(2, B), A, -1));

outputMatrix('Результат', c);

readln;

end.