

ВВЕДЕНИЕ

Цель – разработать программу, которая добавляет новый элемент в заданную матрицу. Программа должна корректно работать с матрицами любого размера, вставляя новый элемент в указанную позицию и сдвигая остальные элементы.

Задачи:

- Составить IDEF0-диаграмму проекта;
- Составить блок-схемы алгоритмов;
- Реализовать алгоритмы на языке Object Pascal;
- Протестировать реализации алгоритмов.

Аналитическая часть

Структура представлена на рисунке 1 в виде IDEF0-нотации. Задача разделяется на несколько блоков, представленных на рисунке 2:

- Ввод данных с клавиатуры;
- Инициализация матрицы ;
- Вставка элемента в матрицу;
- Вывод данных на экран.

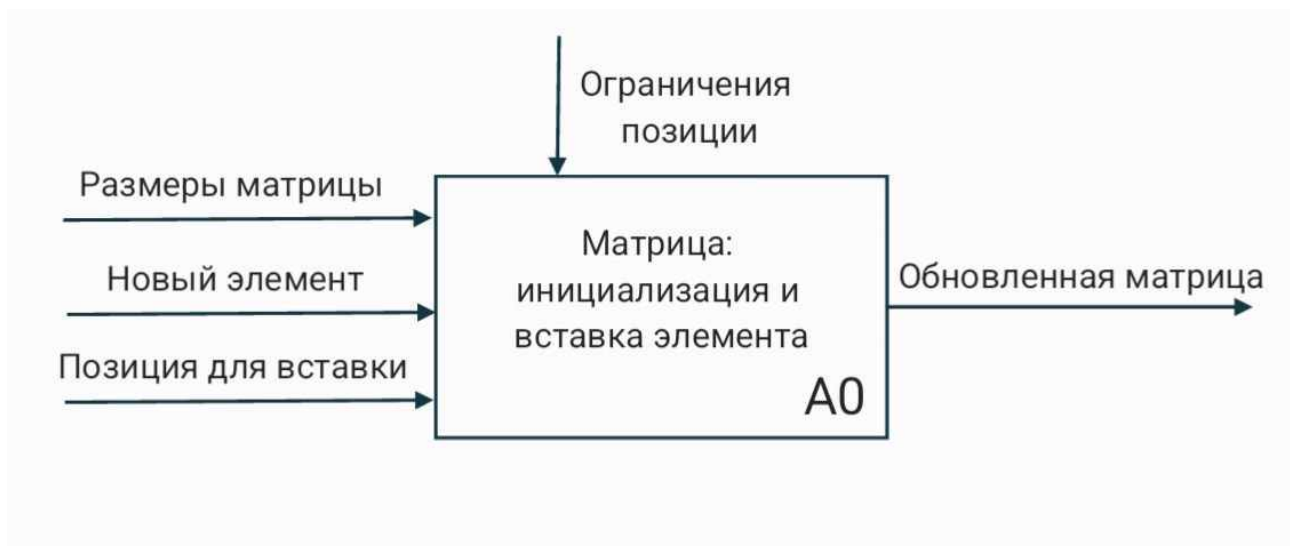


Рисунок 1 – Общая IDEF0-нотация



Рисунок 2 – Подробная IDEF0-нотация

Конструкторская часть

Блоки IDEF0-диаграммы представляют собой 1-2 действия, ввиду чего рациональнее отобразить алгоритм всего проекта целиком без разбиения каждого блока на отдельные процедуры.

Блок-схема алгоритма представлена на рисунке 1. Элементы, отвечающие за интерфейс пользователя, на блок-схеме не отображены; текстовые сообщения, ввиду малозначимости их дословного приведения, представлены сокращенно.

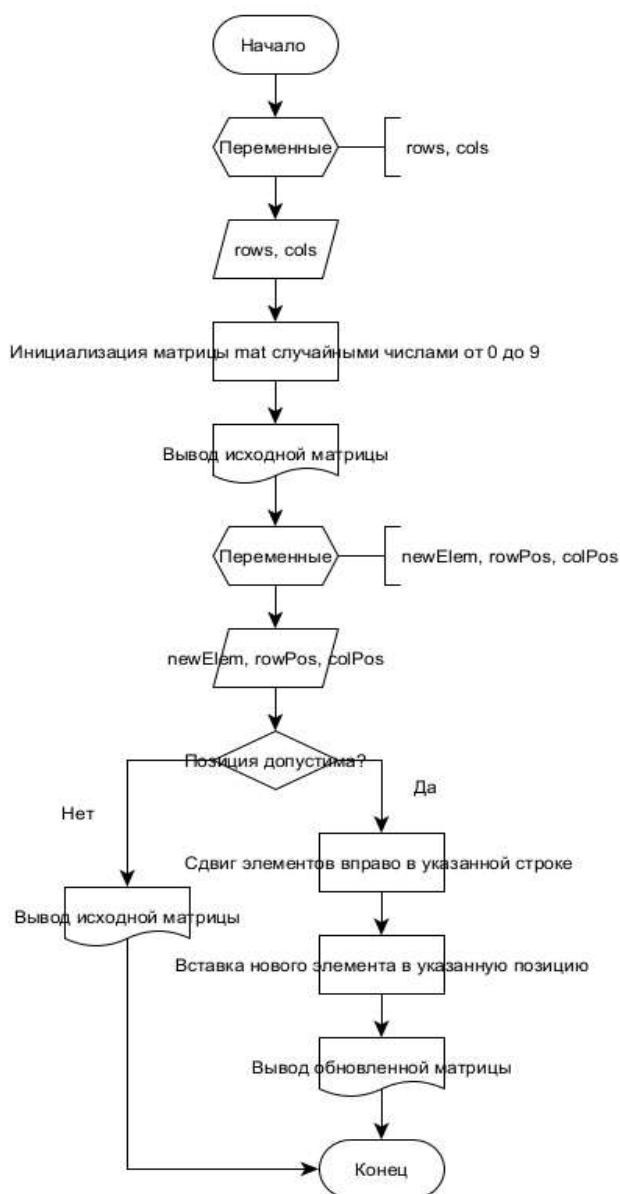


Рисунок 1 – Блок-схема алгоритма программы

Технологическая часть

Реализация алгоритма

В настоящем разделе представлена реализация алгоритма, чья блок-схема представлена на рисунке 1. Реализация была произведена с помощью языка программирования Pascal и представлена в листинге 1.

Листинг 1 – Программа алгоритма

```

program MatrixInsertion;
uses SysUtils;

type
  Matrix = array of array of Integer;

var
  mat: Matrix;
  rows, cols, newElem, rowPos, colPos, i, j: Integer;

procedure InitializeMatrix(var m: Matrix; r, c: Integer);
var
  i, j: Integer;
begin
  SetLength(m, r, c);
  for i := 0 to r - 1 do
    for j := 0 to c - 1 do
      m[i][j] := Random(10);
    end;
  end;

procedure PrintMatrix(m: Matrix);
var
  i, j: Integer;
begin
  for i := 0 to High(m) do
    begin
      for j := 0 to High(m[i]) do
        write(m[i][j]:4);
      end;
      writeln;
    end;
  end;

procedure InsertElement(var m: Matrix; elem, row, col: Integer);
var
  j: Integer;
begin
  if (row < 0) or (row >= Length(m)) or (col < 0) or (col >=
Length(m[0])) then
  begin
    writeln('Некорректная позиция для вставки элемента. ');
    Exit;
  end;

  for j := High(m[row]) downto col + 1 do
    m[row][j] := m[row][j - 1];

  m[row][col] := elem;
end;

begin
  Randomize;
  writeln('Введите количество строк и столбцов матрицы :');
  readln(rows, cols);
  InitializeMatrix(mat, rows, cols);

  writeln('Исходная матрица :');
  PrintMatrix(mat);

  writeln('Введите новый элемент и позицию для вставки :');
  readln(newElem, rowPos, colPos);

  InsertElement(mat, newElem, rowPos - 1, colPos - 1);

  writeln('Матрица после вставки элемента :');
  PrintMatrix(mat);
end.

```

Тестирование реализации

Для данной реализации были использованы методы тестирования черным ящиком – метод эквивалентного разбиения – и белым ящиком – метод комбинаторного покрытия условий и решений. Были выделены следующие эквивалентные классы возможных решений:

1. Введены корректные целые числа, программа корректно выполняет вставку элемента в матрицу. Рассмотрены случаи для различных позиций вставки.
2. Нарушение ввода – введены некорректные значения, такие как нецелые числа или нечисловые значения.
3. Нарушение диапазона значений – введена позиция для вставки элемента за пределами матрицы.
4. Пустая матрица или неверные размеры матрицы для тестирования корректности обработки пустых массивов.

Результаты тестов представлены в таблице 1. Сообщения о системной ошибке предусмотрены для некорректного ввода данных.

Таблица 1 – Тесты реализации программы

№	Входные данные	Выходные данные
1	Матрица 3x3, элемент 5, позиция (1,1)	Матрица после вставки: 1,1,6 5,1,6 7,6,1 >> 7,6,1 0,1,4 0,1,4
2	Матрица 4x4, элемент 12, позиция (1,3)	Матрица после вставки: 2,5,4,9 2,5,12,9 5,5,4,7 >> 5,5,4,7 5,6,4,5 5,6,4,5 1,7,4,6 1,7,4,6
3	Матрица 3x3, элемент 5, позиция (-2,1)	Системная ошибка: Некорректная позиция для вставки элемента (значение вне диапазона)
4	Матрица 3x'с'	Системная ошибка
5	Матрица 0x0, элемент 10, позиция (0,0)	Пустой матричный вывод

ЗАКЛЮЧЕНИЕ

В настоящей работе была составлена программа для добавления нового элемента в указанную позицию матрицы с учетом сдвига элементов..

Задачи:

- Составлена IDEF0-диаграмма проекта;
- Составлены блок-схемы алгоритмов;
- Алгоритмы реализованы на языке Object Pascal;
- Реализации алгоритмов протестированы, были подобраны классы ошибок.