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 №10

Тема работы: Текстовые файлы

Выполнил

студент: гр. 251003 Панкратьев Е.С.

Проверил: Фадеева Е.П.

Минск 2023

Содержание

[1 Постановка задачи 4](#_Toc136795929)

[2 Методика решения 5](#_Toc136795930)

[3 Описание алгоритмов решения задачи 6](#_Toc136795931)

[4 Структура данных 7](#_Toc136795932)

[5 Результаты расчетов 9](#_Toc136795933)

[Приложение А 10](#_Toc136795934)

[Приложение Б 13](#_Toc136795935)

# Постановка задачи

Создать текстовый файл, напечатав первые N-строк треугольника Паскаля в следующем виде:

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

# Методика решения

Треугольник Паскаля, также известный как арифметический треугольник, представляет собой бесконечную таблицу чисел, известных как биномиальные коэффициенты. Он получает свое название в честь французского математика Блеза Паскаля. Треугольник имеет форму треугольника, где в вершине и по бокам расположены единицы.

Каждое число в треугольнике Паскаля равно сумме двух чисел, расположенных над ним. Таким образом, каждое число в треугольнике можно вычислить как комбинацию чисел выше него. Строки треугольника симметричны относительно вертикальной оси.

Треугольник Паскаля имеет множество интересных свойств и находит применение в различных областях математики, включая комбинаторику, теорию вероятностей, алгебру и численные методы. Он является важным инструментом при работе с биномиальными разложениями, построении полиномов и решении задач, связанных с сочетаниями и вероятностями.

В программе вычисляется и представляется треугольник Паскаля с помощью матрицы. Пользователь вводит количество строк треугольника, которое определяет размерность матрицы. Затем вычисляется значение для каждой ячейки матрицы, используя соответствующие формулы. Наконец, полученный треугольник Паскаля записывается в текстовый файл в виде прямоугольного треугольника для дальнейшего отображения и использования.

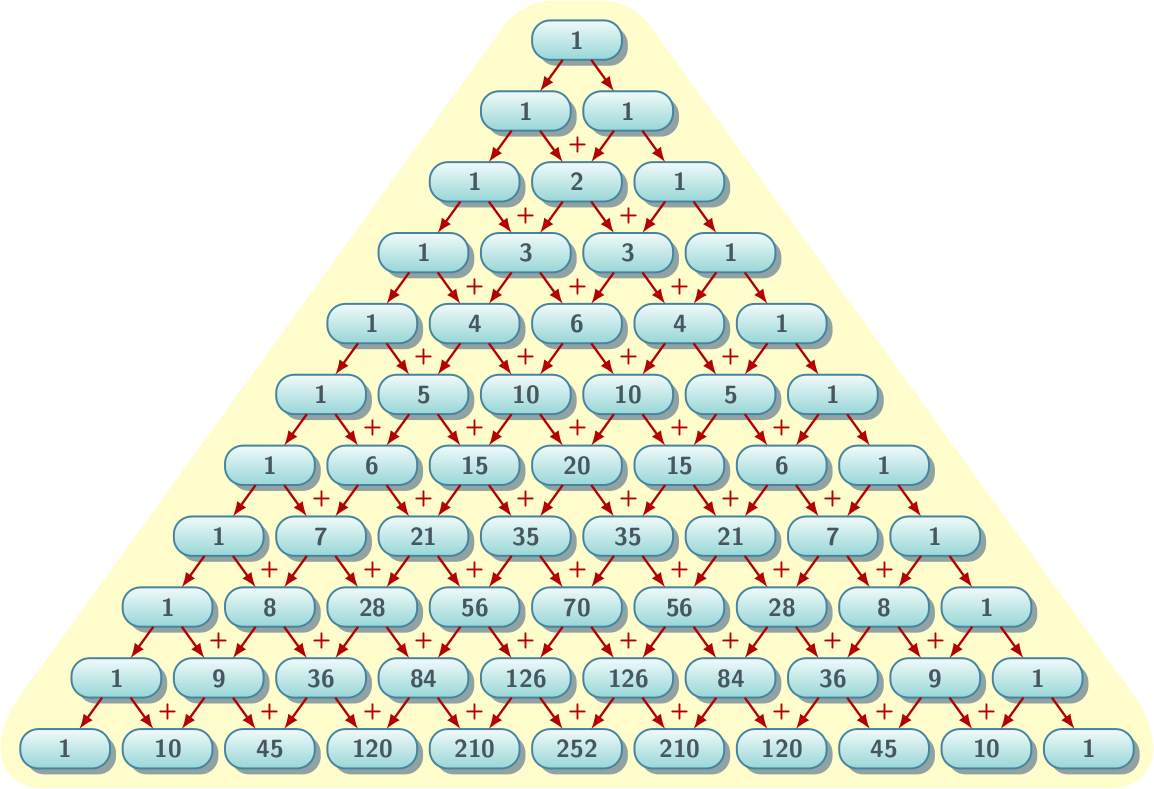


Рисунок 1 – Треугольник паскаля

# Описание алгоритмов решения задачи

Таблица 1 – Описание алгоритмов решения задачи

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п.п. | Наименование алгоритма | Название алгоритма | Формальные  параметры | Предпола-гаемый тип реализации |
| 1 | Основной алгоритм | Запрашивает ввод. Вызов следующих подпрограмм:  CalculatePascalTriangle и WriteToFile |  |  |
| 2 | CalculatePascalTriangle  (  AAmount,  Result  ) | Создает треугольник Паскаля с заданным количеством строк AAmount и сохраняет его в переменной Result | AAmount – получает от фактического параметра адрес с защитой;  Result – получает от фактического параметра адрес с защитой | Функция,  Result – возвращаемый функцией параметр |
| 3 | OutputInFile  (  AOutput,  AMatrix  ) | Формирует прямоугольный треугольник на основе матрицы AMatrix и записывает его в текстовый файл AOutput | aMatrix – получает от фактического параметра адрес с защитой;  TextFile – получает от фактического параметра адрес с защитой | Процедура |
| 4 | WriteToFile  (  AMatrix  ) | Создает текстовый файл, связывает его с вычисленным абсолютным путем, вызывает OutputIn-File для записи прямоугольного треугольника. В конце закрывает файл | aMatrix – получает от фактического параметра адрес с защитой | Процедура |

# Структура данных

Таблица 2 – Структура типов программы

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| TMatrix | array of array of Integer | Двумерный массив для представления треугольника |

Таблица 3 – Структура данных основной программы

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| MaxAmount | Integer | Максимальное количество строк |
| MinAmount | Integer | Минимальное количество строк |
| PascalTriangle | TMatrix | Массив для записи треугольника |
| Amount | Integer | Количество строк в треугольнике |
| IsCorrect | Boolean | Индикатор правильного ответа |

Таблица 4 – Структура данных алгоритма CalculatePascalTriangle(AAmount, Result)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | | Тип параметра |
| AAmount | Integer | Количество строк в треугольнике | | Формальный |
| Result | TMatrix | Полученный треугольник | Формальный | |
| I, J | Integer | Счетчики циклов | Локальный | |

Таблица 5 – Структура данных алгоритма OutputInFile(AOutput, AMatrix)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | | Тип параметра |
| AOutput | TextFile | Текстовый файл для записи | | Формальный |
| AMatrix | TMatrix | Треугольник паскаля, представленный в виде матрицы | Формальный | |
| I, J | Integer | Счетчики циклов | Локальный | |
| numStr | String | Строка для записи в файл | Локальный | |
| MaxLengthNum | Integer | Максимальная длина числа в треугольника | Локальный | |

Таблица 6 – Структура данных алгоритма WriteToFile(AMatrix)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | | Тип параметра |
| AMatrix | TMatrix | Треугольник паскаля, представленный в виде матрицы | Формальный | |
| Output | TextFile | Текстовый файл для записи | Локальный | |

# Результаты расчетов

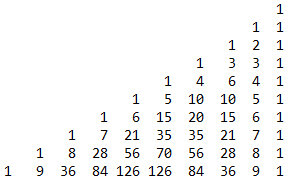


Рисунок 2 – Результаты расчетов

Приложение А

(обязательное)

Исходный код программы

program Lab10;

{

Output N lines of Pascal's triangle

}

uses

System.SysUtils;

type

TMatrix = array of array of Integer;

// Calculate AAmount lines of Pascal's triangle

function CalculatePascalTriangle(const AAmount:

Integer):TMatrix;

var

I, J: Integer;

// I, J - loop counter

begin

// Set length for Result

SetLength(Result, AAmount);

// Calculate Pascal's triangle

for I := 0 to AAmount do

begin

SetLength(Result[I], I + 1);

// First element in each row is always 1

Result[I, 0] := 1;

// Last element in each row is always 1

Result[I, I] := 1;

// Calculate each element in the row based on the values

// above it

for J := 1 to I - 1 do

Result[I, J] := Result[I-1, J-1] +

Result[I-1, J];

end;

end;

// Outputs the matrix to a file in the desired format

procedure OutputInFile(const AOutput: TextFile; const AMa

trix: TMatrix);

var

MaxLengthNum: Integer;

i, j: Integer;

numStr: string;

begin

Reset(AOutput);

Rewrite(AOutput);

// Calculate the maximum length of a number, taking into

// account additional padding

MaxLengthNum := Length(IntToStr(aMatrix[High(AMatrix),

(High(AMatrix) + 1) shr 1]));

for i := Low(AMatrix) to High(AMatrix) do

begin

Write(AOutput, StringOfChar(' ', High(AMatrix) - i +

MaxLengthNum \* (High(AMatrix) - i)));

for j := Low(AMatrix[i]) to High(AMatrix[i]) do

begin

// Convert the current number to a string

numStr := IntToStr(AMatrix[i][j]);

// Add padding spaces to align the number to the right

numStr := StringOfChar(' ', MaxLengthNum - Length(num-

Str)) + numStr;

// Write the formatted number and a space to the output

// file

Write(AOutput, numStr, ' ');

end;

// Write a newline character to move to the next line in

// the output file

WriteLn(AOutput);

end;

end;

procedure WriteToFile(const AMatrix: TMatrix);

var

Output: TextFile;

// Output - output text file where the Pascal's triangle

// will be written

begin

AssignFile(Output, ExtractFilePath(ParamStr(0)) +

'PascalTriangle.txt');

OutputInFile(Output, AMatrix);

CloseFile(Output);

end;

const

MaxAmount = 35;

MinAmount = 0;

var

PascalTriangle: TMatrix;

Amount: Integer;

IsCorrect: Boolean;

// PascalTriangle - store the Pascal's triangle

// Amount - amount of rows to be calculated in Pascal's

// triangle

// IsCorrect - flag indicating the correctness of user

// input

// Output - output text file where the Pascal's trian-

// gle will be written

begin

WriteLn('Please enter a positive number');

IsCorrect := False;

repeat

// Prompt the user to enter a number and validate

// the input

IsCorrect := True;

try

Readln(Amount);

except

IsCorrect := False;

WriteLn('Incorrect data entered. Try again');

end;

// Check if the entered number is negative

if IsCorrect and (Amount < MinAmount) or (Amount >

MaxAmount) then

begin

IsCorrect := False;

WriteLn('Incorrect data entered. Try again');

end;

until IsCorrect;

// Calculate Pascal's triangle based on the entered

// amount

PascalTriangle := CalculatePascalTriangle(Amount);

// Output file and write Pascal's triangle to it

WriteToFile(PascalTriangle);

end.

Приложение Б

(обязательное)

Тестовые наборы

**Тест 1**

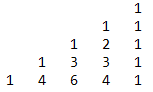
****

Рисунок 3 – Результаты расчетов

**Тест 2**

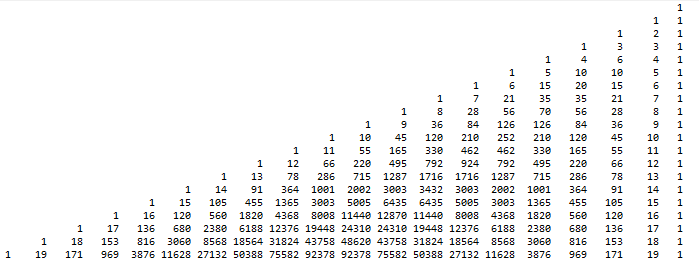
****

Рисунок 4 – Результаты расчетов



Рисунок 5 – Результаты расчетов

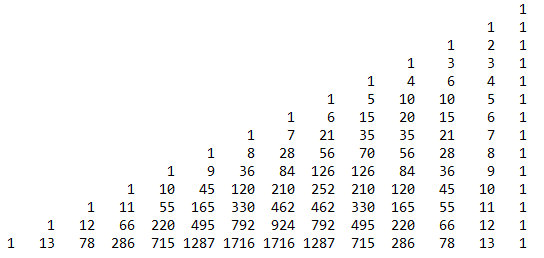


Рисунок 6 – Результаты расчетов