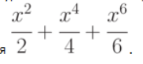
1. **«ПРОЦЕДУРЫ И ФУНКЦИИ. ПЕРЕГРУЗКА ФУНКЦИЙ (МЕТОДОВ) И ОПЕРАТОРОВ»**

Задание 1. Разработать метод f(n, x), который для заданного натурального числа n и вещественного x находит значение выражения (x^n)/(n). Вычислить с помощью данного метода значение выражения (x^2)/(2) + (x^4)/(4) + (x^6)/(6).



Листинг программы:

using System;

namespace ConsoleApp1

{

class Program

{

static double f(int n, double x)

{

double result = 0;

for (int i = 2; i <= n; i += 2)

{

result += Math.Pow(x, i) / i;

}

return result;

}

static void Main(string[] args)

{

Console.WriteLine("Введите значение n:");

int n = int.Parse(Console.ReadLine());

double x = 5;

double expressionValue = f(n, x);

Console.WriteLine($"Значение выражения: {expressionValue}");

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 5 | 168,75 |

Анализ результатов:

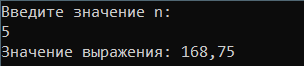
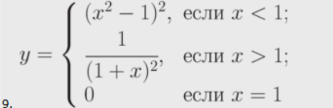


Рисунок 1.1 – Результат работы программы

Задание 2. Построить таблицу значений функции y = f(x) для x∈|a, b| с шагом h. Для решения задачи использовать вспомогательный метод:



Листинг программы:

using System;

namespace ConsoleApp

{

class Program

{

static double f(double x)

{

double y;

if (x < 0) y = Math.Pow(x \* x - 1, 2);

else if (x > 5 && x < 20) y = 1 / Math.Pow(1 + x, 2);

else y = 0;

return y;

}

static void Main(string[] args)

{

Console.Write("Введите значение a: ");

double a = double.Parse(Console.ReadLine());

Console.Write("Введите значение b: ");

double b = double.Parse(Console.ReadLine());

Console.Write("Введите значение h: ");

double h = double.Parse(Console.ReadLine());

Console.WriteLine("Таблица значений функции y = f(x):");

Console.WriteLine("------------------------------------");

for (double x = a; x <= b; x += h)

{

double y = f(x);

Console.WriteLine($"x = {x:f2}, y = {y:f4}");

}

Console.ReadLine();

}

}

}

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| -5 25 2.5 | x = -5,00, y = 576,0000  x = -2,50, y = 27,5625  x = 0,00, y = 0,0000  x = 2,50, y = 0,0000  x = 5,00, y = 0,0000  x = 7,50, y = 0,0138  x = 10,00, y = 0,0083  x = 12,50, y = 0,0055  x = 15,00, y = 0,0039  x = 17,50, y = 0,0029  x = 20,00, y = 0,0000  x = 22,50, y = 0,0000  x = 25,00, y = 0,0000 |

Анализ результатов:

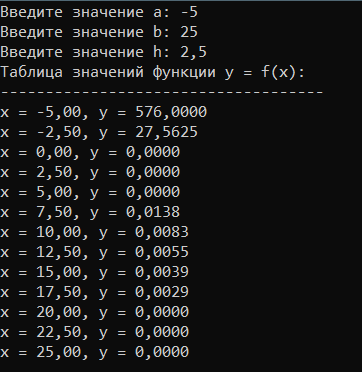


Рисунок 1.2 – Результат работы программы

Задание 3. Перегрузи метод f так, чтобы его сигнатура соответствовала виду static void f (double x, out double y).

Листинг программы:

using System;

namespace ConsoleApp

{

class Program

{

static void f(double x, out double y)

{

if (x < 0)

{

y = Math.Pow(x \* x - 1, 2);

}

else if (x > 5 && x < 20)

{

y = 1 / Math.Pow(1 + x, 2);

}

else

{

y = 0;

}

}

static void Main(string[] args)

{

Console.Write("Введите значение a: ");

double a = double.Parse(Console.ReadLine());

Console.Write("Введите значение b: ");

double b = double.Parse(Console.ReadLine());

Console.Write("Введите значение h: ");

double h = double.Parse(Console.ReadLine());

Console.WriteLine("Таблица значений функции y = f(x):");

Console.WriteLine("------------------------------------");

for (double x = a; x <= b; x += h)

{

f(x, out double y);

Console.WriteLine($"x = {x:f2}, y = {y:f4}");

}

Console.ReadLine();

}

}

}

Таблица 1.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| -5 25 2.5 | x = -5,00, y = 576,0000  x = -2,50, y = 27,5625  x = 0,00, y = 0,0000  x = 2,50, y = 0,0000  x = 5,00, y = 0,0000  x = 7,50, y = 0,0138  x = 10,00, y = 0,0083  x = 12,50, y = 0,0055  x = 15,00, y = 0,0039  x = 17,50, y = 0,0029  x = 20,00, y = 0,0000  x = 22,50, y = 0,0000  x = 25,00, y = 0,0000 |

Анализ результатов:

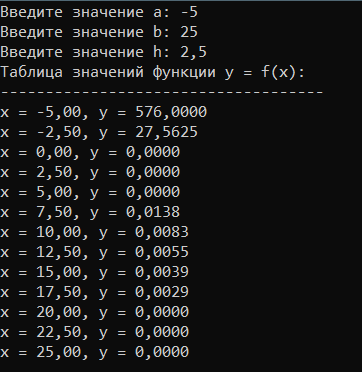


Рисунок 1.3 – Результат работы программы

Задание 4. Описать класс для работы с двумерным массивом вещественных чисел. Обеспечить добавление к первому столбцу столбца заданного номера (перегрузка операции +).

Листинг программы:

using System;

class Program

{

static void Main(string[] args)

{

// Создаем двумерный массив

DoubleArray da = new DoubleArray(3, 3);

da[0, 0] = 1.0; da[0, 1] = 2.0; da[0, 2] = 3.0;

da[1, 0] = 4.0; da[1, 1] = 5.0; da[1, 2] = 6.0;

da[2, 0] = 7.0; da[2, 1] = 8.0; da[2, 2] = 9.0;

Console.WriteLine("Original array:");

PrintArray(da);

// Создаем столбец для добавления

DoubleArray column = new DoubleArray(3, 1);

column[0, 0] = 0.1; column[1, 0] = 0.2; column[2, 0] = 0.3;

// Добавляем столбец

DoubleArray newDa = da + column;

Console.WriteLine("New array:");

PrintArray(newDa);

Console.ReadKey();

}

static void PrintArray(DoubleArray da)

{

for (int i = 0; i < da.Rows; i++)

{

for (int j = 0; j < da.Cols; j++)

{

Console.Write(da[i, j] + " ");

}

Console.WriteLine();

}

}

}

Класс

using System;

class DoubleArray

{

private double[,] array;

public DoubleArray(int rows, int cols)

{

array = new double[rows, cols];

}

public double this[int i, int j]

{

get { return array[i, j]; }

set { array[i, j] = value; }

}

public int Rows

{

get { return array.GetLength(0); }

}

public int Cols

{

get { return array.GetLength(1); }

}

public static DoubleArray operator +(DoubleArray da, DoubleArray column)

{

if (da.Rows != column.Rows)

throw new ArgumentException("Number of rows in added column must match number of rows in array.");

DoubleArray result = new DoubleArray(da.Rows, da.Cols + 1);

for (int i = 0; i < da.Rows; i++)

{

result[i, 0] = column[i, 0];

for (int j = 0; j < da.Cols; j++)

{

result[i, j + 1] = da[i, j];

}

}

return result;

}

}

Таблица 1.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Original array:  1 2 3  4 5 6  7 8 9 | New array:  0,1 1 2 3  0,2 4 5 6  0,3 7 8 9 |

Анализ результатов:

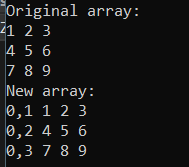


Рисунок 1.4 – Результат работы программы

Задание 5. Описать класс «предметный указатель». Каждый компонент указателя содержит слово и номера страниц, на которых это слово встречается.

Количество номеров страниц, относящихся к одному слову, от одного до десяти. Предусмотреть возможность формирования указателя с клавиатуры и из файла, вывода указателя, вывода номеров страниц для заданного слова, удаления элемента из указателя. Написать программу, демонстрирующую все разработанные элементы класса.

Листинг программы:

class Program

{

static void Main(string[] args)

{

Index index = new Index();

index.ReadFromConsole();

index.Display();

Console.WriteLine("Введите слово для поиска:");

string word = Console.ReadLine();

index.DisplayPages(word);

Console.WriteLine("Введите слово для удаления:");

word = Console.ReadLine();

index.Remove(word);

Console.WriteLine("Введите имя файла для загрузки индекса:");

string filename = Console.ReadLine();

index.ReadFromFile(filename);

index.Display();

Console.ReadLine();

}

}

Класс

using System;

using System.Collections.Generic;

using System.IO;

class Index

{

// Ключ - слово, значение - список номеров страниц, на которых это слово встречается

private Dictionary<string, List<int>> index = new Dictionary<string, List<int>>();

// Добавление слова и номера страницы в индекс

public void Add(string word, int page)

{

if (!index.ContainsKey(word))

{

index.Add(word, new List<int>());

}

index[word].Add(page);

}

// Вывод индекса на экран

public void Display()

{

Console.WriteLine("Предметный указатель:");

foreach (var entry in index)

{

Console.Write(entry.Key + ": ");

foreach (int page in entry.Value)

{

Console.Write(page + " ");

}

Console.WriteLine();

}

}

// Вывод номеров страниц для заданного слова

public void DisplayPages(string word)

{

if (index.ContainsKey(word))

{

Console.WriteLine("Страницы, на которых встречается слово \"" + word + "\":");

foreach (int page in index[word])

{

Console.Write(page + " ");

}

Console.WriteLine();

}

else

{

Console.WriteLine("Слово \"" + word + "\" не найдено в предметном указателе.");

}

}

// Удаление элемента из индекса

public void Remove(string word)

{

if (index.ContainsKey(word))

{

index.Remove(word);

Console.WriteLine("Слово \"" + word + "\" удалено из предметного указателя.");

}

else

{

Console.WriteLine("Слово \"" + word + "\" не найдено в предметном указателе.");

}

}

// Формирование индекса из файла

public void ReadFromFile(string filename)

{

try

{

using (StreamReader reader = new StreamReader(filename))

{

while (!reader.EndOfStream)

{

string line = reader.ReadLine();

string[] parts = line.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

string word = parts[0];

for (int i = 1; i < parts.Length; i++)

{

int page = int.Parse(parts[i]);

Add(word, page);

}

}

}

Console.WriteLine("Индекс успешно загружен из файла " + filename);

}

catch (Exception ex)

{

Console.WriteLine("Ошибка при чтении файла " + filename + ": " + ex.Message);

}

}

// Формирование индекса с клавиатуры

public void ReadFromConsole()

{

Console.WriteLine("Введите слова и номера страниц через пробел. Для окончания ввода введите пустую строку.");

while (true)

{

string line = Console.ReadLine();

if (line == "") break;

string[] parts = line.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

string word = parts[0];

for (int i = 1; i < parts.Length; i++)

{

int page = int.Parse(parts[i]);

Add(word, page);

}

}

}

}

Таблица 1.5 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| пока 4  норм 6  марк 96 | пока: 4  марк: 96 |

Анализ результатов:

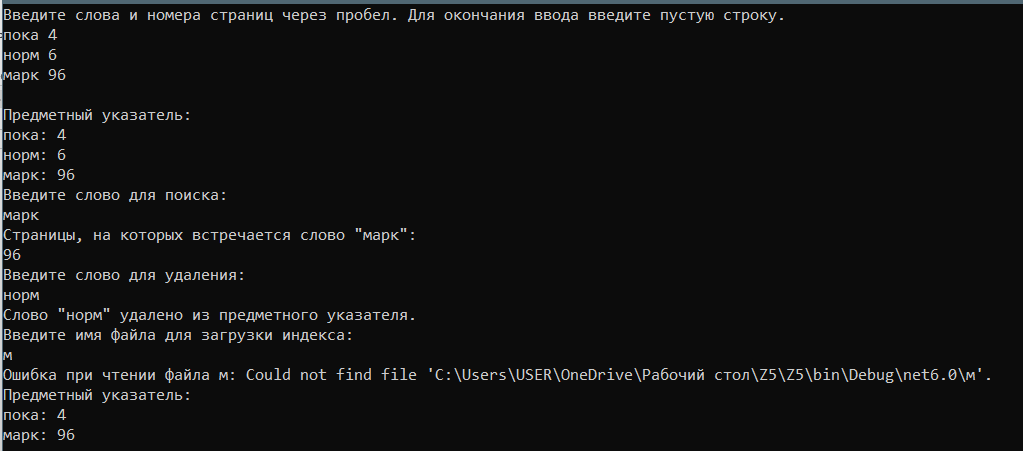


Рисунок 1.5 – Результат работы программы