|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

|  |  |
| --- | --- |
| **ФАКУЛЬТЕТ** | **ИУК «Информатика и управление»** |
| **КАФЕДРА** | **ИУК2 «Информационные системы и сети»** |
|  | |

**Практическая работа 1.1**

**«Функции и массивы»**

**ДИСЦИПЛИНА: «Объектно-ориентированное программирование»**



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил: студент гр. ИУК4-21Б | |  |  | ( | Суриков Н.С | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |
| Проверил: | |  |  | ( | Дерюгина Е. О. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |

|  |  |
| --- | --- |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | |
|  | - Балльная оценка:  - Оценка: |

**Цель:** реализовать работу с функциями и массивами на языке C#.

**Задачи:**

1. Составить подпрограмму определения минимального элемента в одномерном массиве X=(x1, x2, …, xn). Используя её, найти максимум функции z=-x2-y2-4x+6y-8 в следующих сечениях:

при x=-1, 2<=y<=4, hy=0,2

при y=3, -2<=x<=0, hx=0,2

для вычисления z рекомендуется использовать функцию.

2. Составить подпрограмму для определения максимума функции у = f(х), заданной в дискретных, точках отрезка [а, b] с постоянным шагом h. Используя ее, найти максимумы функций:

y1 = ех\*соs2х - 1 на отрезке [1; 2], h = 0.1;

у2 = х\*1n|х + 1| на отрезке [0; 2], h = 0.2.

3. Пусть дано натуральное число n и вещественные числа а1 … аn. В последовательности а1 … аn все отрицательные члены увеличьте на 0,5, а положительные, меньшие среднего арифметического, замените на 0,1.

4.Найти количество простых чисел в массиве.

Перевернуть все нечетные строки матрицы.

Перевернуть каждое четное слово в строке.

5.Удалить элемент с заданным ключом (значением)

Добавить строку в конец матрицы

**Листинг программы:**

1 using System;

2 using System.Linq;

3 using System.Collections.Generic;

4

5

6 namespace PR1

7 {

8 class Program

9 {

10 static void Main()

11 {

12 Task1();

13 Task2();

14 Task3();

15 Task4();

16 Task5();

17 }

18 static void Task1()

19 {

20 int[] X = [5, 3, 8, 1, 9, 2];

21 Console.WriteLine($"Минимальный элемент в массиве: {FindMinElement(X)}");

22 Console.WriteLine($"Максимум функции z: {FindMaxZ()}");

23

24 }

25

26 static void Task2()

27 {

28 *// Максимум функции y1 = e^x \* cos(2x) - 1 на отрезке [1; 2], h = 0.1*

29 double max1 = FindMax(Function1, 1, 2, 0.1, 1, double.MinValue);

30 Console.WriteLine("Максимум функции y1 на отрезке [1; 2] равен: " + max1);

31

32 *// Максимум функции y2 = x \* ln(|x| + 1) на отрезке [0; 2], h = 0.2*

33 double max2 = FindMax(Function2, 0, 2, 0.2, 0, double.MinValue);

34 Console.WriteLine("Максимум функции y2 на отрезке [0; 2] равен: " + max2);

35 }

36

37 static void Task3()

38 {

39 int n = 5;

40 double[] arr = { -2.5, 1.3, -0.7, 0.8, -1.2 };

41

42 Console.WriteLine("Исходный массив:");

43 foreach (double num in arr)

44 {

45 Console.Write(num + ", ");

46 }

47 Console.WriteLine();

48

49 ProcessArray(arr);

50

51 Console.WriteLine("Преобразованный массив:");

52 foreach (double num in arr)

53 {

54 Console.Write(num + ", ");

55 }

56 Console.WriteLine();

57 }

58

59 static void Task4()

60 {

61 int[] numbers = { 2, 3, 5, 7, 10, 11, 13, 17 };

62 int primeCount = CountPrimes(numbers);

63 Console.WriteLine("Количество простых чисел в массиве: " + primeCount);

64

65 int[,] matrix = new int[,]

66 {

67 {1, 2, 3},

68 {4, 5, 6},

69 {7, 8, 9},

70 {10, 11, 12}

71 };

72

73 ReverseOddRows(matrix);

74

75 Console.WriteLine("Матрица с перевернутыми нечетными строками:");

76 for (int i = 0; i < matrix.GetLength(0); i++)

77 {

78 for (int j = 0; j < matrix.GetLength(1); j++)

79 {

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

81 }

82 Console.WriteLine();

83 }

84

85 string sentence = "Hello world, this is a test sentence";

86 string[] words = sentence.Split(' ');

87

88 ReverseEvenWords(words);

89

90 string reversedSentence = string.Join(" ", words);

91 Console.WriteLine("Строка с перевернутыми четными словами: " + reversedSentence);

92 }

93

94 static void Task5()

95 {

96 List<int> numbers = new List<int> { 1, 2, 3, 4, 5 };

97 int key = 3;

98

99 if (numbers.Contains(key))

100 {

101 numbers.Remove(key);

102 Console.WriteLine("Элемент с ключом " + key + " удален из списка.");

103 }

104 else

105 {

106 Console.WriteLine("Элемент с ключом " + key + " не найден в списке.");

107 }

108

109 List<List<int>> matrix = new List<List<int>>

110 {

111 new List<int> {1, 2, 3},

112 new List<int> {4, 5, 6},

113 new List<int> {7, 8, 9}

114 };

115

116 List<int> newRow = new List<int> { 10, 11, 12 };

117 matrix.Add(newRow);

118

119 Console.WriteLine("Матрица после добавления строки:");

120 foreach (var row in matrix)

121 {

122 foreach (int num in row)

123 {

124 Console.Write(num + " ");

125 }

126 Console.WriteLine();

127 }

128 }

129

130 static bool IsPrime(int num)

131 {

132 if (num <= 1)

133 {

134 return false;

135 }

136

137 for (int i = 2; i <= Math.Sqrt(num); i++)

138 {

139 if (num % i == 0)

140 {

141 return false;

142 }

143 }

144

145 return true;

146 }

147

148 static int CountPrimes(int[] arr)

149 {

150 int count = 0;

151 foreach (int num in arr)

152 {

153 if (IsPrime(num))

154 {

155 count++;

156 }

157 }

158 return count;

159 }

160

161 static void ReverseOddRows(int[,] matrix)

162 {

163 for (int i = 0; i < matrix.GetLength(0); i++)

164 {

165 if (i % 2 != 0)

166 {

167 Array.Reverse(matrix, i \* matrix.GetLength(1), matrix.GetLength(1));

168 }

169 }

170 }

171

172 static void ReverseEvenWords(string[] words)

173 {

174 for (int i = 0; i < words.Length; i++)

175 {

176 if (i % 2 == 1)

177 {

178 char[] charArray = words[i].ToCharArray();

179 Array.Reverse(charArray);

180 words[i] = new string(charArray);

181 }

182 }

183 }

184 static double FindAveragePozitive(double[] arr)

185 {

186 double sum = 0;

187 int count = 0;

188

189 *// Находим сумму всех положительных чисел в массиве и их количество*

190 foreach (double num in arr)

191 {

192 if (num > 0)

193 {

194 sum += num;

195 count++;

196 }

197 }

198

199 return sum / count;

200 }

201

202 static void ProcessArray(double[] arr)

203 {

204 double average = FindAveragePozitive(arr);

205

206 for (int i = 0; i < arr.Length; i++)

207 {

208 if (arr[i] < 0)

209 {

210 arr[i] += 0.5; *// Увеличиваем отрицательные числа на 0.5*

211 }

212 else if (arr[i] < average)

213 {

214 arr[i] = 0.1; *// Заменяем положительные числа, меньшие среднего, на 0.1*

215 }

216 }

217 }

218

219 static double Function1(double x)

220 {

221 return Math.Exp(x) \* Math.Cos(2 \* x) - 1;

222 }

223

224 static double Function2(double x)

225 {

226 return x \* Math.Log(Math.Abs(x) + 1);

227 }

228

229 static double FindMax(Func<double, double> func, double a, double b, double step, double currentX, double max)

230 {

231 if (currentX > b)

232 {

233 return max;

234 }

235

236 double y = func(currentX);

237

238 if (y > max)

239 {

240 max = y;

241 }

242

243 return FindMax(func, a, b, step, currentX + step, max);

244 }

245

246 static int FindMinElement(int[] array)

247 {

248 int min = array[0];

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

250 {

251 if (array[i] < min)

252 {

253 min = array[i];

254 }

255 }

256 return min;

257 }

258

259 static double FunctionZ(double x, double y)

260 {

261 return -Math.Pow(x, 2) - Math.Pow(y, 2) - 4 \* x + 6 \* y - 8;

262 }

263

264 static double FindMaxZ()

265 {

266 double maxZ = double.MinValue;

267

268 *// При x=-1, 2<=y<=4, hy=0,2*

269 for (double y = 2; y <= 4; y += 0.2)

270 {

271 double currentZ = FunctionZ(-1, y);

272 if (currentZ > maxZ)

273 {

274 maxZ = currentZ;

275 }

276 }

277

278 *// При y=3, -2<=x<=0, hx=0,2*

279 for (double x = -2; x <= 0; x += 0.2)

280 {

281 double currentZ = FunctionZ(x, 3);

282 if (currentZ > maxZ)

283 {

284 maxZ = currentZ;

285 }

286 }

287

288 return maxZ;

289 }

290 }

291 }

**Результат работы:**

Минимальный элемент в массиве: 1

Максимум функции z: 5

Максимум функции y1 на отрезке [1; 2] равен: -2,1312043837568133

Максимум функции y2 на отрезке [0; 2] равен: 2,197224577336219

Исходный массив:

-2,5, 1,3, -0,7, 0,8, -1,2,

Преобразованный массив:

-2, 1,3, -0,19999999999999996, 0,1, -0,7,

Количество простых чисел в массиве: 7

Матрица с перевернутыми нечетными строками:

1 2 3

6 5 4

7 8 9

12 11 10

Строка с перевернутыми четными словами: Hello ,dlrow this si a tset sentence

Элемент с ключом 3 удален из списка.

Матрица после добавления строки:

1 2 3

4 5 6

7 8 9

10 11 12

**Вывод:** в результате работы мы познакомились с массивами и функциями в языке C#.

**Основная литература**

1. Зыков, С. В. Введение в теорию программирования. Объектно-ориентированный подход : учебное пособие / С. В. Зыков. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 187 c. — ISBN 978-5-4497-0926-4. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/102007.html.

2. Павловская, Т. А. Программирование на языке высокого уровня C# : учебное пособие / Т. А. Павловская. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 245 c. — Текст : электронный — URL: http://www.iprbookshop.ru/102051.html.

3. Биллиг, В. А. Основы объектного программирования на C# (С# 3.0, Visual Studio 2008) : учебник / В. А. Биллиг. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 409 c. — Текст : электронный — URL: http://www.iprbookshop.ru/102029.html.

4. Горелов, С. В. Современные технологии программирования: разработка Windows-приложений на языке С#. В 2 томах. Т.I : учебник / С. В. Горелов ; под редакцией П. Б. Лукьянова. — Москва : Прометей, 2019. — 362 c. — Текст : электронный — URL: http://www.iprbookshop.ru/94532.html.

5. Горелов, С. В. Современные технологии программирования: разработка Windows-приложений на языке С#. В 2 томах. Т.II : учебник / С. В. Горелов ; под редакцией П. Б. Лукьянова. — Москва : Прометей, 2019. — 378 c. — Текст : электронный — URL: http://www.iprbookshop.ru/94533.html.