Министерство высшего образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет» (ПНИПУ)**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

Объектно-ориентированное программирование

«Лабораторная работа №5»

Выполнил

Студент группы РИС-22-2б

Прядеин И.А.

Проверил доцент кафедры

ИТАС

Викентьева О.Л.

Пермь 2023

**Постановка задачи:**

1. Сформировать динамический одномерный массив, заполнить его случайными числами и вывести на печать.
2. Выполнить указанное в варианте задание и вывести полученный массив на печать.
3. Сформировать динамический двумерный массив, заполнить его случайными числами и вывести на печать.
4. Выполнить указанное в варианте задание и вывести полученный массив на печать.
5. Сформировать динамический двумерный массив, заполнить его случайными числами и вывести на печать.
6. Выполнить указанное в варианте задание и вывести полученный массив на печать.
7. При реализации функций необходимо продемонстрировать использование параметров разных типов и различные способы организации функций (параметры по умолчанию, перегрузку функций, и т.д.)

**Вариант 20:**

|  |  |  |  |
| --- | --- | --- | --- |
| 20 | Удалить элемент с заданным номером | Добавить строки после каждой чётной строки матрицы | Добавить строку в конец массива |

**Анализ задачи:**

**Текст программы:**

using System;

namespace lab\_5

{

class Input

{

public static int IntInput(string message)

{

int number;

bool isConvert;

string buf;

do

{

Console.Write(message);

buf = Console.ReadLine();

isConvert = int.TryParse(buf, out number);

if (!isConvert)

{

Console.Write("Неверный формат данных, введите новое значение: ");

}

} while (!isConvert);

return number;

}

}

internal class Program

{

static void Main()

{

int choose;

do

{

choose = Input.IntInput(GetMainMenu());

choose = CheckSelection(choose, 1, 4);

int secondChoose;

switch (choose)

{

case 1:

int[] arr = new int[0];

do

{

Console.WriteLine("\n1. Создать массив");

Console.WriteLine("2. Напечатать массив");

Console.WriteLine("3. Удалить элемент с заданным номером");

Console.Write("4. Назад");

secondChoose = Input.IntInput("\n> ");

secondChoose = CheckSelection(secondChoose, 1, 4);

switch (secondChoose)

{

case 1:

arr = CreateOneDimArray();

break;

case 2:

if (arr.Length == 0)

Console.WriteLine("Размер массива равен 0!");

else

PrintArray(arr);

break;

case 3:

if (arr.Length == 0)

{

Console.WriteLine("Размер массива равено 0!");

}

else

{

int pos = Input.IntInput("Введите номер удаляемого элемента: ");

arr = DeleteWithPos(arr, pos);

}

break;

}

} while (secondChoose != 4);

break;

case 2:

int[,] twoDimArr = new int[0, 0];

do

{

Console.WriteLine("1. Создать массив");

Console.WriteLine("2. Напечатать массив");

Console.WriteLine("3. Добавить строки после каждой четной строки матрицы");

Console.Write("4. Назад");

secondChoose = Input.IntInput("\n> ");

secondChoose = CheckSelection(secondChoose, 1, 4);

switch (secondChoose)

{

case 1:

twoDimArr = CreateTwoDimArray();

break;

case 2:

if (twoDimArr.GetLength(0) == 0)

Console.WriteLine("Размер массива равен 0!");

else

PrintArray(twoDimArr);

break;

case 3:

if (twoDimArr.GetLength(0) == 0)

{

Console.WriteLine("Размер массива равен 0!");

}

else

{

twoDimArr = InsertRows(twoDimArr);

}

break;

}

} while (secondChoose != 4);

break;

case 3:

int[][] tornArr = new int[0][];

do

{

Console.WriteLine("1. Создать массив");

Console.WriteLine("2. Напечатать массив");

Console.WriteLine("3. Добавить строку в конец массива");

Console.Write("4. Назад");

secondChoose = Input.IntInput("\n> ");

secondChoose = CheckSelection(secondChoose, 1, 4);

switch (secondChoose)

{

case 1:

tornArr = CreateTornArray();

break;

case 2:

if (tornArr.GetLength(0) == 0)

Console.WriteLine("Размер массива равен 0!");

else

PrintArray(tornArr);

break;

case 3:

if (tornArr.GetLength(0) == 0)

{

Console.WriteLine("Размер массива равен 0!");

}

else

{

tornArr = AddRowAtEnd(tornArr);

}

break;

}

} while (secondChoose != 4);

break;

}

} while (choose != 4);

}

static int[] DeleteWithPos(int[] arr, int pos)

{

while (pos <= 0 || pos > arr.Length)

{

pos = Input.IntInput("Неверный номер элемента, введите новое значение: ");

}

int[] newArr = new int[arr.Length - 1];

int j = 0;

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

{

if (i + 1 != pos)

{

newArr[j] = arr[i];

j++;

}

}

return newArr;

}

static int[,] InsertRows(int[,] arr)

{

int newRows = 0;

if (arr.GetLength(0) % 2 == 1)

{

newRows = arr.GetLength(0) / 2 + 1 + arr.GetLength(0);

}

else

{

newRows = arr.GetLength(0) / 2 + arr.GetLength(0);

}

int[,] newArr = new int[newRows, arr.GetLength(1)];

Random rnd = new Random();

int row = 0;

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

{

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

{

newArr[row, j] = arr[i, j];

}

row++;

if ((i + 1) % 2 == 1)

{

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

{

newArr[row, j] = rnd.Next(0, 20);

}

row++;

}

}

return newArr;

}

static int[][] AddRowAtEnd(int[][] arr)

{

int[][] newArr = new int[arr.Length + 1][];

Random rnd = new Random();

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

{

newArr[i] = new int[arr[i].Length];

for (int j = 0; j < arr[i].Length; j++)

{

newArr[i][j] = arr[i][j];

}

}

int columns = Input.IntInput("Введите количество столбцов: ");

while (columns <= 0)

{

columns = Input.IntInput("Количество столбцов не может быть меньше или равно 0, введите новое значение: ");

}

newArr[newArr.Length - 1] = new int[columns];

for (int i = 0; i < columns; i++)

{

newArr[newArr.Length - 1][i] = rnd.Next(0, 20);

}

return newArr;

}

static int CheckSelection(int number, int min, int max)

{

while (number < min || number > max)

{

number = Input.IntInput("Такого выбора не существует, введите новое значение:\n> ");

}

return number;

}

static string GetMainMenu()

{

string menu = "1. Работа с одномерными массивами\n" +

"2. Работа с двумерными массивами\n" +

"3. Работа с рваными массивами\n" +

"4. Выход\n> ";

return menu;

}

static void PrintArray(int[] arr)

{

foreach (int i in arr)

Console.Write(i + " ");

Console.WriteLine();

}

static void PrintArray(int[,] arr)

{

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

{

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

{

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

}

Console.WriteLine();

}

}

static void PrintArray(int[][] arr)

{

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

{

for (int j = 0; j < arr[i].Length; j++)

{

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

}

Console.WriteLine();

}

}

static int[] CreateOneDimArray()

{

Console.WriteLine("1. Ввести значения вручную");

Console.WriteLine("2. Заполнить случайными числами");

int choose = Input.IntInput("> ");

choose = CheckSelection(choose, 1, 2);

int size = Input.IntInput("Введите размер массива: ");

while (size <= 0)

{

size = Input.IntInput("Размер массива не может быть меньше или равен 0, введите новое значение: ");

}

int[] newArr = new int[size];

if (choose == 1)

{

Console.WriteLine("Введите элементы массива:");

for (int i = 0; i < size; i++)

{

newArr[i] = Input.IntInput("");

}

}

if (choose == 2)

{

Random rnd = new Random();

for (int i = 0; i < size; i++)

{

newArr[i] = rnd.Next(0, 20);

}

}

return newArr;

}

static int[,] CreateTwoDimArray()

{

Console.WriteLine("1. Ввести значения вручную");

Console.WriteLine("2. Заполнить случайными числами");

int choose = Input.IntInput("> ");

choose = CheckSelection(choose, 1, 2);

int rows = Input.IntInput("Введите количество строк: ");

while (rows <= 0)

{

rows = Input.IntInput("Количество строк не может быть меньше или равено 0: ");

}

int columns = Input.IntInput("Введите количество столбцов: ");

while (columns <= 0)

{

columns = Input.IntInput("Количество столбцов не может быть меньше или равено 0: ");

}

int[,] newArr = new int[rows, columns];

if (choose == 1)

{

Console.WriteLine();

for (int i = 0; i < rows; i++)

{

Console.WriteLine($"Строка {i + 1}:");

for (int j = 0; j < columns; j++)

{

newArr[i, j] = Input.IntInput("");

}

}

}

if (choose == 2)

{

Random rnd = new Random();

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < columns; j++)

{

newArr[i, j] = rnd.Next(0, 20);

}

}

}

return newArr;

}

static int[][] CreateTornArray()

{

Console.WriteLine("1. Ввести значения вручную");

Console.WriteLine("2. Заполнить случайными числами");

int choose = Input.IntInput("> ");

choose = CheckSelection(choose, 1, 2);

int rows = Input.IntInput("Введите количество строк: ");

while (rows <= 0)

{

rows = Input.IntInput("Количество строк не может быть меньше или равено 0: ");

}

int[][] newTornArr = new int[rows][];

Random rnd = new Random();

for (int i = 0; i < rows; i++)

{

int columns = Input.IntInput("Введите количество столбцов: ");

while (columns <= 0)

{

columns = Input.IntInput("Количество столбцов не может быть меньше или равно 0: ");

}

newTornArr[i] = new int[columns];

for (int j = 0; j < columns; j++)

{

if (choose == 1)

{

newTornArr[i][j] = Input.IntInput("");

}

else

{

newTornArr[i][j] = rnd.Next(0, 20);

}

}

}

return newTornArr;

}

}

}

Результат работы программы: