

Учреждение образования
«Белорусский государственный университет информатики и
радиоэлектроники»

ОТЧЕТ О ПРОДЕЛАННОЙ РАБОТЕ

«Функции»

Выполнил:
Студент гр. 114302
Голод И.О.

Проверил:
Кабариха В.А.

Минск 2021

Для выполнения этого задания я выбрал практическую работу No4, так как она самая объемная. Эта работа состоит из 5 задач, которые чем-то похожи друг на друга, например, во всех задачах надо создавать массив, заполнять его, выводить на экран, удалять, что дает еще один повод максимально разбить эти задачи на функции. К сожалению, некоторые из них (недетерминированные) нельзя считать чистыми, так как в них используется функция rand().

Вот несколько примеров моих функций:

```
void showArray(int* array, int size_of_array)
{
    for (int i = 0; i < size_of_array; i++)
    {
        cout<<array[i]<<"\t";
    }
}
```

```
void createMatrix(int** matrix, int rows, int columns)
{
    for (int i = 0; i < rows; i++)
    {
        matrix[i] = new int[columns];
    }
}
```

```
void randMatrix(int** matrix, int rows, int columns)
{
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
            matrix[i][j] = rand() % 21 - 10;
        }
    }
}
```

```
void showMatrix(int** matrix, int rows, int columns)
{
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
            cout << matrix[i][j] << "\t";
        }
        cout << endl;
    }
}
```

```

void deleteMatrix(int** matrix, int rows)
{
    for (int i = 0; i < rows; i++)
    {
        delete[] matrix[i];
    }
    delete[] matrix;
}

```

```

void createArray(int* array, int size_of_array)
{
    for (int i = 0; i < size_of_array; i++)
    {
        array[i] = rand() % 21 - 10;
    }
}

```

```

void findSumAndAvgOfMatrix(int** matrix, int rows, int columns)
{
    double avg;
    int sum;
    int counter;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
            sum = sum + matrix[i][j];
            counter++;
        }
    }
    avg = sum / counter;
    cout<<"Сумма всех элементов матрицы : "<<sum<<endl;
    cout<<"Среднее значение элементов матрицы : "<<avg<<endl<<endl;
}

```

```

void multiplicationOfMatrix(int** first_matrix, int** second_matrix, int first_rows, int first_columns, int second_rows, int second_columns)
{
    int third_rows = first_rows;
    int third_columns = second_columns;
    int** third_matrix = new int* [third_rows];
    createMatrix(third_matrix, third_rows, third_columns);
    for (int i = 0; i < first_rows; i++)
    {
        for (int j = 0; j < second_columns; j++)
        {
            third_matrix[i][j] = 0;
            for (int a = 0; a < first_columns; a++)
            {
                third_matrix[i][j] += first_matrix[i][a] * second_matrix[a][j];
            }
        }
    }
    cout<<"Результат перемножения матриц : "<<endl;
}

```

Для сравнения код без функций:

```
cout<<"Введите количество строк и столбцов : "<<endl;
cin>>rows>>columns;
int** matrix = new int* [rows];
createMatrix(matrix, rows, columns);
randMatrix(matrix, rows, columns);
showMatrix(matrix, rows, columns);
for (int i = 0; i < rows; i++)
{
    for (int j = 0; j < columns; j++)
    {
        sum = sum + matrix[i][j];
        counter++;
    }
}
avg = sum / counter;
cout<<"Сумма всех элементов матрицы : "<<sum<<endl;
cout<<"Среднее значение элементов матрицы : "<<avg<<endl<<endl;
```

```
cout<<"Введите количество строк и столбцов второй матрицы : "<<endl;
cin>>second_rows>>second_columns;
int** second_matrix = new int* [second_rows];
createMatrix(second_matrix, second_rows, second_columns);
randMatrix(second_matrix, second_rows, second_columns);
showMatrix(second_matrix, second_rows, second_columns);
if (first_rows == second_columns)
{
    int third_rows = first_rows;
    int third_columns = second_columns;
    int** third_matrix = new int* [third_rows];
    createMatrix(third_matrix, third_rows, third_columns);
    for (int i = 0; i < first_rows; i++)
    {
        for (int j = 0; j < second_columns; j++)
        {
            third_matrix[i][j] = 0;
            for (int a = 0; a < third_rows; a++)
            {
                third_matrix[i][j] += first_matrix[i][a] * second_matrix[a][j];
            }
        }
    }
    cout<<"Результат перемножения матриц : "<<endl;
    showMatrix(third_matrix, third_rows, third_columns);
    deleteMatrix(third_matrix, third_rows);
}
else cout<<"Количество столбцов первой матрицы и строк второй матрицы должно совпадать!"<<endl;
```