

## Системное программирование

### Практическая работа 3

Используя классы для работы с двоичными файлами BinaryReader (<https://learn.microsoft.com/ru-ru/dotnet/api/system.io.binaryreader?view=net-5.0>) и BinaryWriter (<https://learn.microsoft.com/ru-ru/dotnet/api/system.io.binarywriter?view=net-5.0>) реализовать программный продукт, реализующий чтение и запись файлов с учетом различных кодировок, предусмотренных операционной системой. Примерный вид программного продукта представлен на рисунке 1.

Выбор файла для чтения и записи реализовать с применением диалоговых компонентов OpenFileDialog и SaveFileDialog.

Для выбора формата кодировки использовать класс Encoding.

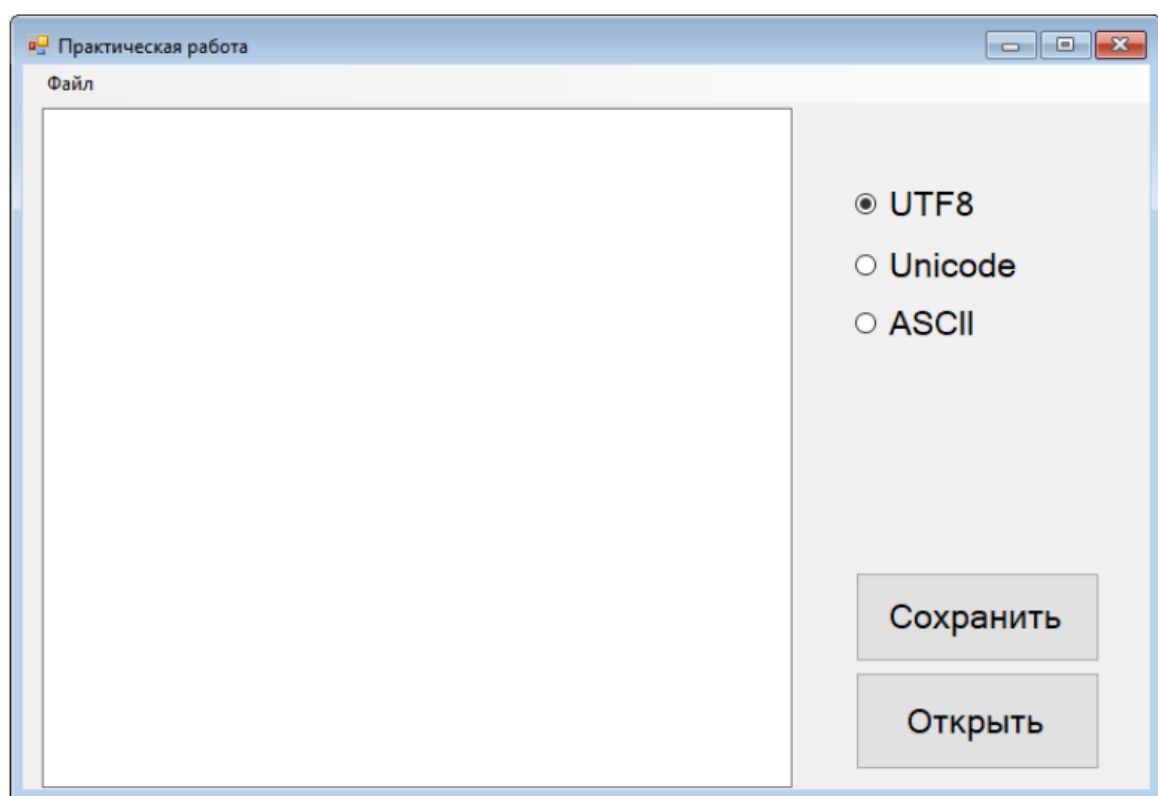


Рисунок 1 –Примерный интерфейс программного продукта

#### Листинг - 1

```
SaveFileDialog saveFileDialog1 = new SaveFileDialog();
Encoding Cod = Encoding.UTF8;
if (radioButton1.Checked)
{
```

```

        Cod = Encoding.UTF8;
    }
    else if (radioButton2.Checked)
    {
        Cod = Encoding.Unicode;
    }
    else if (radioButton3.Checked)
    {
        Cod = Encoding.ASCII;
    }

    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        string s = saveFileDialog1.FileName;

        using (var stream = File.Open(s, FileMode.Create))
        {
            using (var writer = new BinaryWriter(stream, Cod, false))
            {
                writer.Write(textBox1.Text);
            }
        }
    }
}

```

## Листинг – 2

```

string tempDirectory;
OpenFileDialog openFileDialog = new OpenFileDialog();
Encoding Cod = Encoding.UTF8;
if (radioButton1.Checked)
{
    Cod = Encoding.UTF8;
}
else if (radioButton2.Checked)
{
    Cod = Encoding.Unicode;
}
else if (radioButton3.Checked)
{
    Cod = Encoding.ASCII;
}
if (openFileDialog.ShowDialog() == DialogResult.OK)
{
    string s = openFileDialog.FileName;

    using (var stream = File.Open(s, FileMode.Open))
    {
        using (var reader = new BinaryReader(stream, Cod, false))
        {
            tempDirectory = reader.ReadString();
            textBox1.Text = tempDirectory;
        }
    }
}
}

```

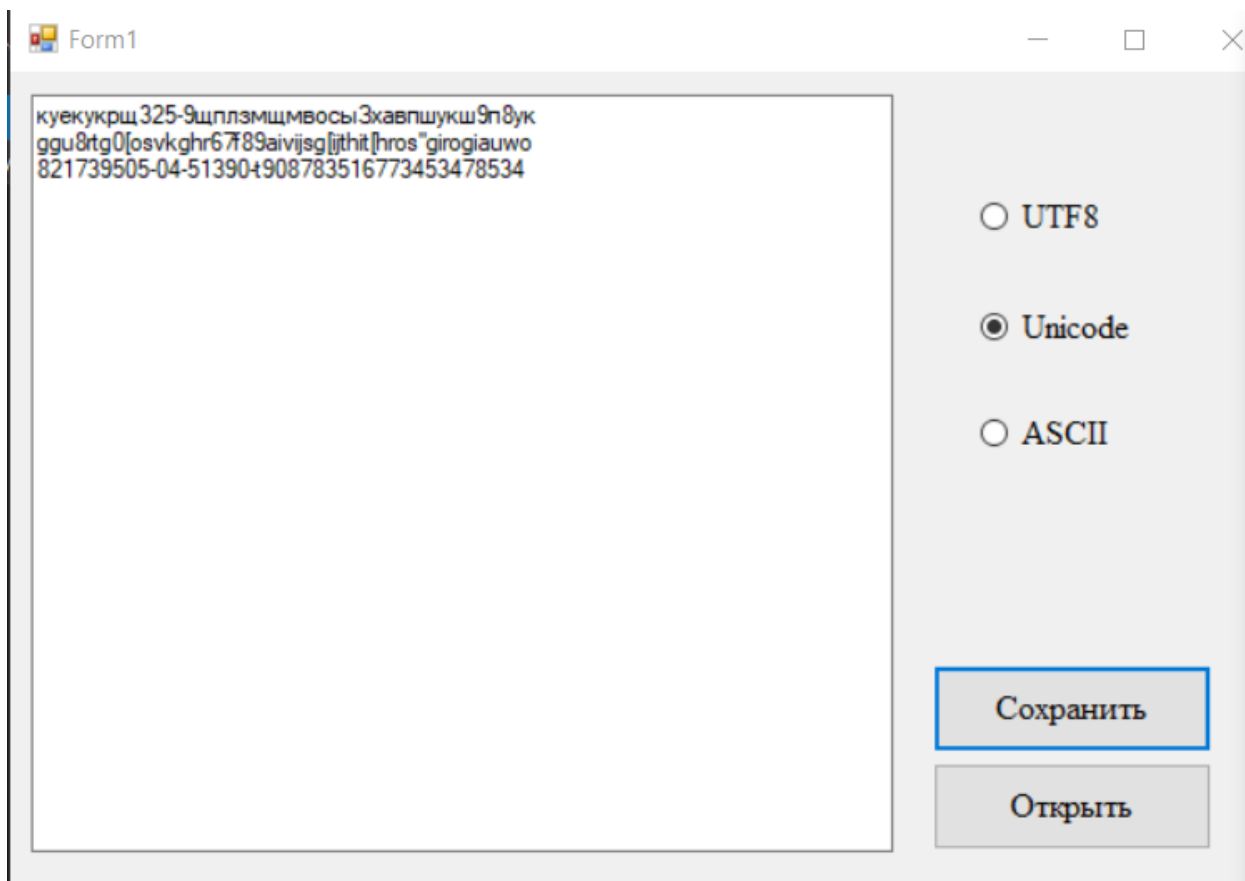


Рисунок 2 – Интерфейс программы

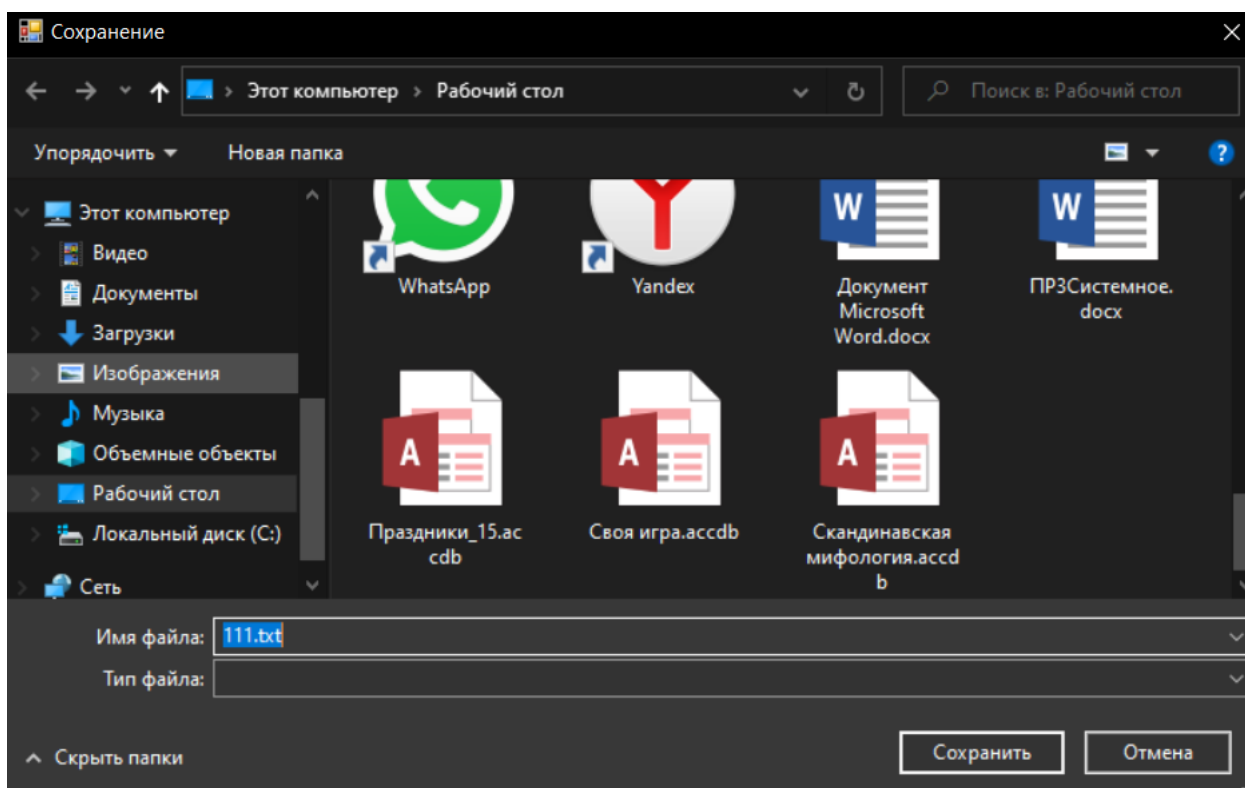


Рисунок 3 – Диалоговое окно для сохранения файла

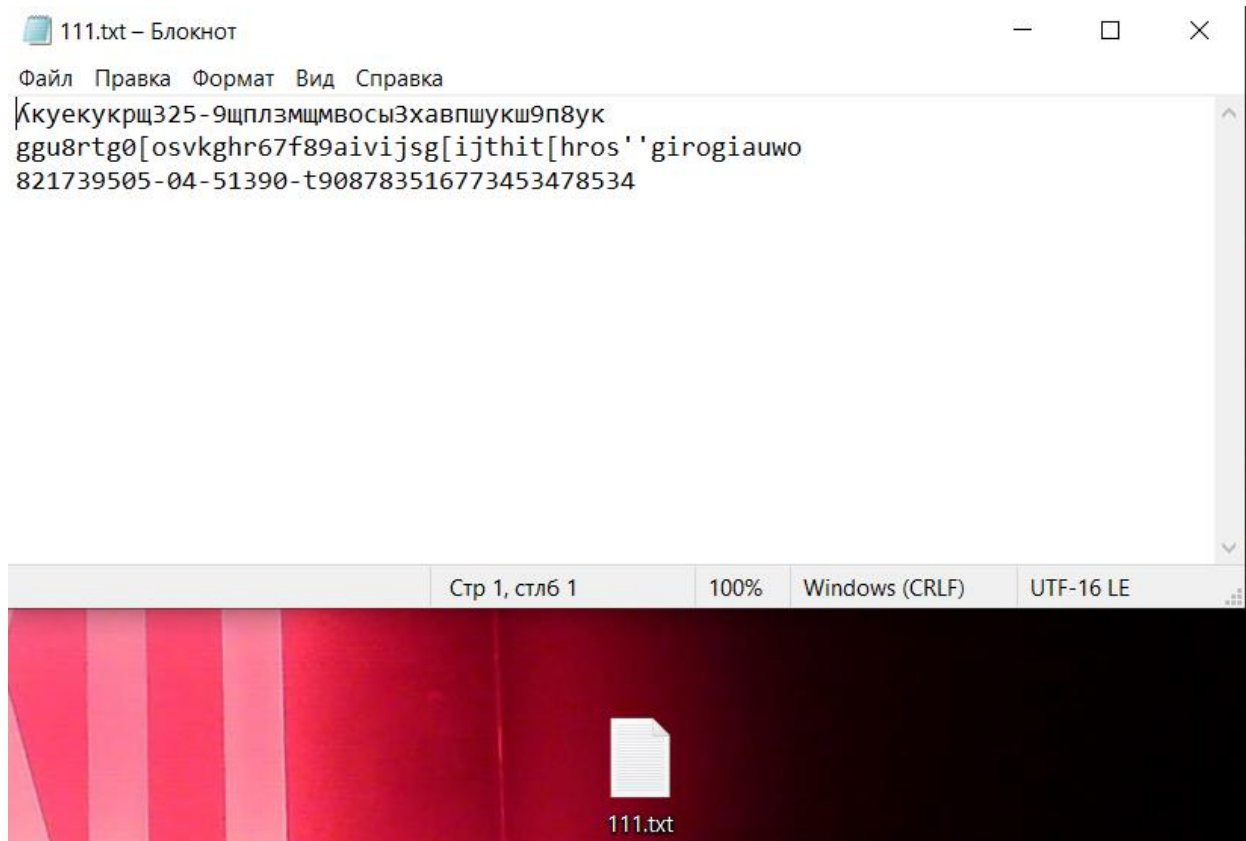


Рисунок 4 – Сохранение файла с кодировкой Unicode

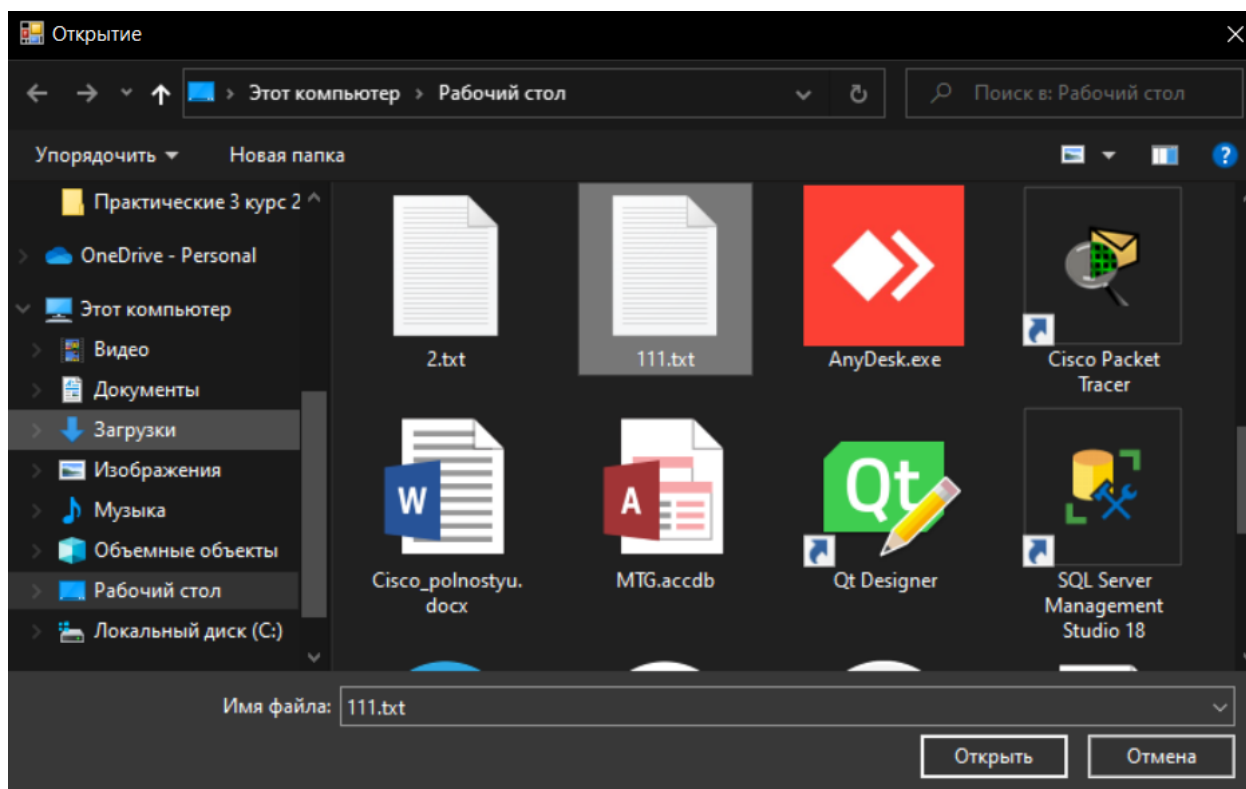


Рисунок 5 – Открытие файла

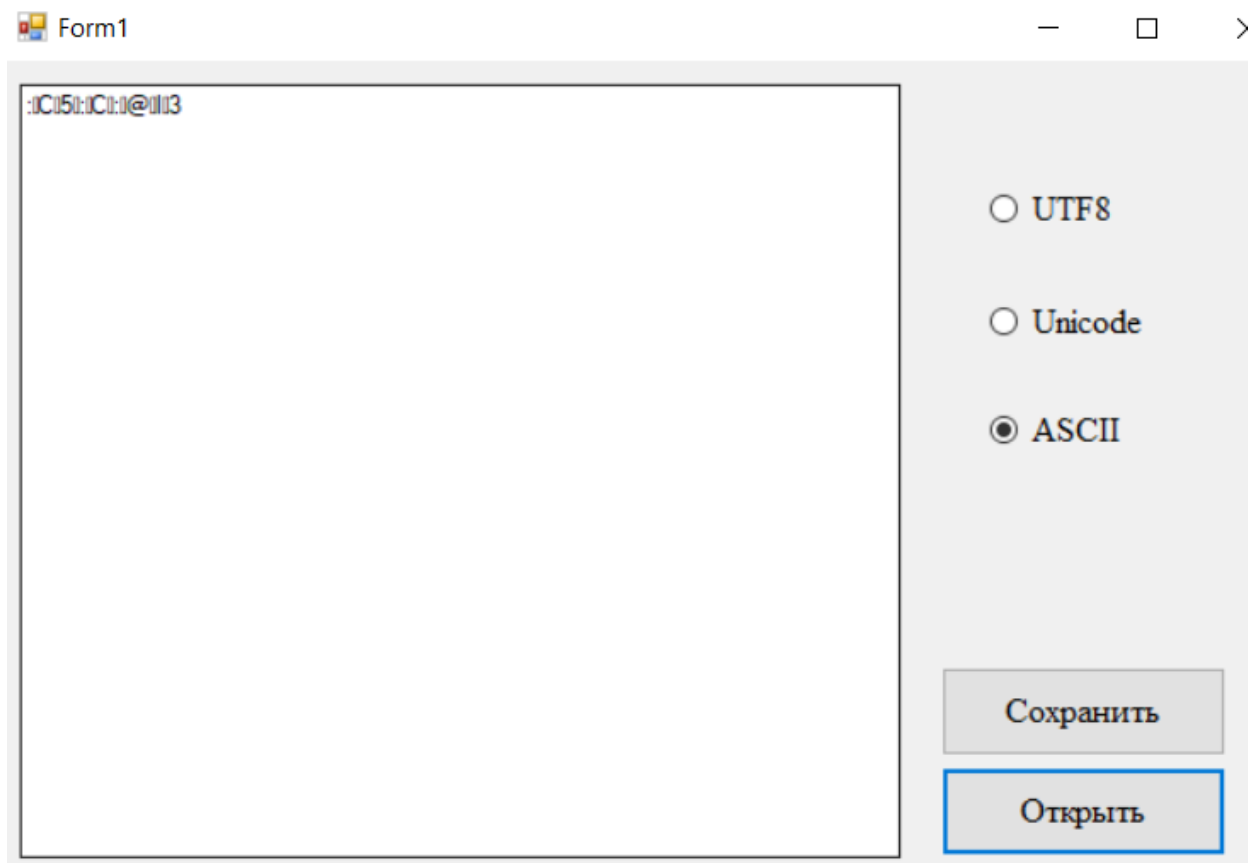


Рисунок 6 – Результат конвертирования файла с кодировкой Unicode в TextBox с кодировкой ASCII

Ссылка на гитхаб:

[https://github.com/Alexandrov911/Pr3Sistem\\_Binary26.01.2023.git](https://github.com/Alexandrov911/Pr3Sistem_Binary26.01.2023.git)