# **Задание 2. Разработка описания функции**

**ВЫПОЛНЕНИЕ ЗАДАНИЯ:**

**Мысли:**

Так как функция изложенная в тестовом задании уже содержит все коментарии и по которым итак понятно что функция выполняет, я понял что задание состоит в том чтобы переписать данную функцию на C#.

**Код на C#:**

// подключаем

**using** System.Text.RegularExpressions;

**...**

**string** FormatFilePath(string filePath, string mask)

{

// локальные переменные

string disc;

string directory;

string fileName;

string extension;

// удаляем пробелы

filePath = filePath.Trim();

try

{

// В полном имени файла FilePath находим:

// 1) имя диска (не работает, если FilePath - сетевой путь)

disc = Regex.Match(filePath,

@"^([A-Z]{1}):\\").Value;

// 2) директорию

directory = Regex.Match(filePath,

@"^[A-Z]{1}:\\(.\*)\\[^\\]+").Value;

// 3) имя файла

fileName = Regex.Match(filePath,

@"^[A-Z]{1}:\\.\*\\([^\\]+)\.[^\.\\]+$").Value;

// 4) расширение файла

extension = Regex.Match(filePath,

@"^[A-Z]{1}:\\.\*\\[^\\]+\.([^\.\\]+)$").Value;

}

catch

{

// если полное имя файла задано не верно, возвращаем пустую строку

return "";

}

// компонуем полученные значения (диск, директорию, имя файла и расширение)

// в строку, как указано в маске mask

string result = mask;

// заменить %D на имя диска

result = result.Replace("%D", disc);

// заменить %P на путь

result = result.Replace("%P", directory);

// заменить %N на имя файла

result = result.Replace("%N", fileName);

// заменить %E на расширение

result = result.Replace("%E", extension);

return result;

}