**КОЛЛЕКЦИИ И КЛАССЫ ПРОТОТИПЫ**

Задание 1.

Листинг программы:

namespace task1;  
  
class Program  
{  
 static void Main()  
 {  
 string inputString = "abcasd#af##asdasd##asdasd#d##c";  
 string outputString = TransformString(inputString);  
 Console.WriteLine(outputString);   
 }  
  
 static string TransformString(string input)  
 {  
 Stack<char> stack = new Stack<char>();  
  
 foreach (char c in input)  
 {  
 if (c == '#')  
 {  
 if (stack.Count > 0)  
 {  
 stack.Pop();   
 }  
 }  
 else  
 {  
 stack.Push(c);   
 }  
 }  
  
 char[] result = new char[stack.Count];  
 int index = stack.Count - 1;  
 while (stack.Count > 0)  
 {  
 result[index--] = stack.Pop();  
 }  
  
 return new string(result);  
 }  
}

Входные и выходные данные:

|  |  |
| --- | --- |
| Входные | Выходные |
| abcasd#af##asdasd##asdasd#d##c | abcasasdaasdac |

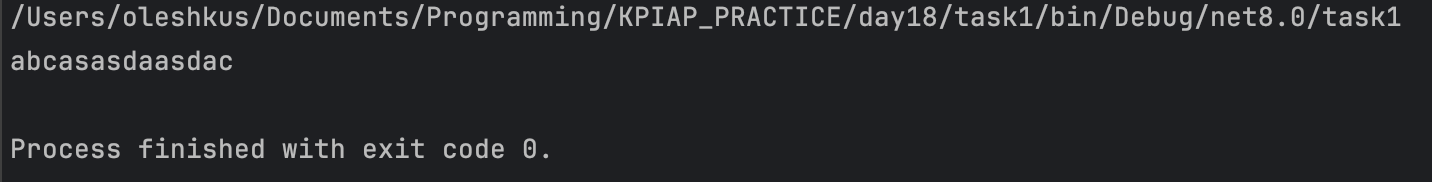
Анализ результатов: 

Рисунок 1 – Результат работы программы

Задание 2.

Листинг программы:

namespace task2;  
  
class Program  
{  
 static void Main(string[] args)  
 {  
 string filePath = "random\_text.txt";  
 CreateRandomFile(filePath, 100);  
  
   
 Queue<char> nonDigitQueue = new Queue<char>();  
 Queue<char> digitQueue = new Queue<char>();  
  
   
 foreach (char c in File.ReadAllText(filePath))  
 {  
 if (char.IsDigit(c))  
 {  
 digitQueue.Enqueue(c);   
 }  
 else  
 {  
 nonDigitQueue.Enqueue(c);   
 }  
 }  
  
 while (nonDigitQueue.Count > 0)  
 {  
 Console.Write(nonDigitQueue.Dequeue());  
 }  
  
 while (digitQueue.Count > 0)  
 {  
 Console.Write(digitQueue.Dequeue());  
 }  
 }  
  
 static void CreateRandomFile(string filePath, int length)  
 {  
 Random random = new Random();  
 char[] randomChars = new char[length];