2\_4**random.random(）**

（1）用于生成一个0到1之间的随机符点数（注意不包含0和1）

（2）用于生成一个（a，b）之间的浮点数

2\_5字符串是用于表示文本的字符的有序集合, C#支持两种形式的字符文字：常规字符串文本和逐字字符串文本。

**2\_6**string类的concat方法的拼接速度远小于stringBuffer类的Append方法

Concat方法整体是一个数组的拷贝。Append方法都在做字符数组的处理、加长、拷贝等。这些都是基本的数据处理，整个方法内没有生成对象。

3-2

using System;

using namespace ;

{

class Program

{

static void Main(string[] args)

{

string s = "asdasd";

Console.Write("Before deletion: s=");

Console.WriteLine(s);

Console.WriteLine("\nInput the char that you want to delete in s.");

string a = Console.ReadLine().ToString();

Console.Write("\nAfter deletion: s=");

s = s.Replace(a, "");

Console.WriteLine(s);

Console.WriteLine("\nPress any key to quit.");

Console.ReadKey();

}

}

}

3\_3

using System;

using System.Collections.Generic;

using namespace ;

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("请输入要进行判断的字符串");

string str = Console.ReadLine();

Stack<char> stack = new Stack<char>();

Queue<char> queue = new Queue<char>();

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

{

stack.Push(str[i]);

queue.Enqueue(str[i]);

}

bool isPlalindrome = true;//isPlalindrome

while (stack.Count > 0)

{

if (stack.Pop() != queue.Dequeue())isPlalindrome

{

isPlalindrome = false;

break;

}

}

Console.WriteLine("字符串是回文串：" + isPlalindrome);

Console.ReadKey();

}

}

}

3\_13

using System;

using namespace

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Input 10 integers...");

int[] a = new int[10];

for (int i = 0; i < 10; i++)

a[i] = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Before exchange :");

foreach (var i in a)

Console.Write(i + ",");

for (int i = 0; i < 5; i++)

{

int t = a[i];

a[i] = a[9 - i];

a[9 - i] = t;

}

Console.WriteLine("\nAfter exchange :");

foreach (var i in a)

Console.Write(i + ",");

Console.WriteLine("\nPress any key to quit.");

Console.ReadKey();

}

}

}