



C#语言基础（第2讲）





本堂课内容

- 检查学习视频的情况
- 熟悉C#基本语句
- 编程：随机数应用：蒙特卡罗法



视频回顾：数组

- 数组的使用要两步
- `int[] a = new int[10];`
- `for(int i=0; i<a.Length; i++) ...` 可访问每个元素
- 或 `foreach(int x in a)...` 这是只读式遍历

关于随机数



- 伪随机数

- 线性同余算法

- $X_{n+1} = (X_n * a + c) \bmod m,$

- 其中 $m > 0$ 称为模数, $0 \leq a < m$ 称为乘数, $0 \leq c < m$ 称为增量, $0 \leq X_0 < m$ 称为初始值或种子

- 如 $X = (X * 123 + 59) \bmod 65535$

- 如 $X = (X * 2053 + 13849) \bmod 65535$

- 如 a 取 69069或 1664525(即&H19660D)



C++(vs2003)中的公式

- `((holdrand = holdrand * 214013L + 2531011L) >> 16) & 0x7fff);`



Java语言中用的公式

- `synchronized public void setSeed(long seed) {`
 - ▣ `this.seed = (seed ^ 0x5DEECE66DL) & ((1L << 48) - 1);`
- `}`
- `synchronized protected int next(int bits) {`
 - ▣ `seed = (seed * 0x5DEECE66DL + 0xBL) & ((1L << 48) - 1);`
 - ▣ `return (int)(seed >>> (48 - bits));`
- `}`
- Donald Knuth (高德纳) , *The Art of Computer Programming, Volume 2, Section 3.2.1.*

C#语言中用的公式



- C# core中
 - Numerical Recipes in C (2nd Ed.)
 - <https://github.com/dotnet/corefx>
 - <https://github.com/dotnet/corefx/blob/master/src/Common/src/CoreLib/System/Random.cs>
- C#中
 - <https://referencesource.microsoft.com/>
 - <https://referencesource.microsoft.com/#mscorlib/system/random.cs,bb77e610694e64ca>
- C++boost中用的算法
 - Mersenne twister -- 目前为止最好的随机数算法



生成标准正态分布的随机数

- 产生方法很多。
- 近似方法（利用中心极限定理）
- 即用 n 个 $N(0,1)$ 变量产生一个 $U(0,1)$ 变量。 $N(0,1)$ $\bar{u} = \frac{1}{n} \sum_{i=1}^n u_i$
- 其中 u_i 是抽自 $U(0,1)$ 的随机数， $x = \sqrt{12n}(\bar{u} - 1/2)$ 可近似为一个 $N(0,1)$ 变量。
- 简单地说：6个均匀分布的随机数的平均值是一个正态分布的随机数



蒙特卡罗 (Monte Carlo) 方法

- 蒙特卡洛方法是一种应用随机数来进行计算机模拟的方法。此方法对研究的系统进行随机观察抽样，通过对样本值的观察统计，求得所研究系统的某些参数。
- 蒙特卡罗方法的主要应用范围包括：粒子输运问题，统计物理，典型数学问题，真空技术，激光技术以及医学，生物，探矿等方面。随着科学技术的发展，其应用范围将更加广泛。
- 蒙特卡罗方法在粒子输运问题中的应用范围主要包括：实验核物理，反应堆物理，高能物理等方面。
- 蒙特卡罗方法在实验核物理中的应用范围主要包括：通量及反应率，中子探测效率，光子探测效率，光子能量沉积谱及响应函数，气体正比计数管反冲质子谱，多次散射与通量衰减修正等方面。

第2次作业（随机数）



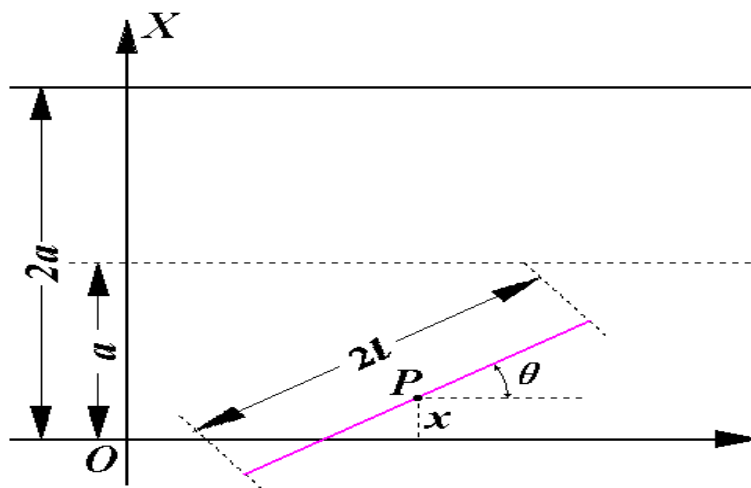
- 两个色子，分别为1到6
- 每次执出两个色子，问它们的和最大可能性为多少？



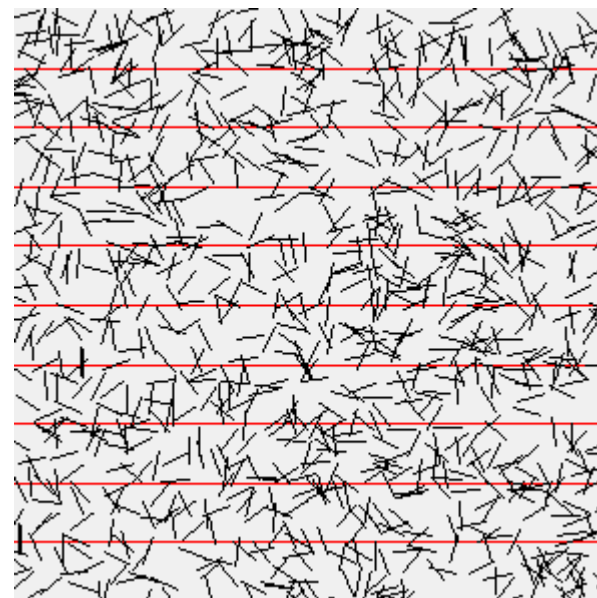
蒲丰氏(Buffon)问题

$$\pi = \frac{2l}{aP} \approx \frac{2l}{a\bar{s}_N}$$

- 如果针的长度是平行间的距离的一半，则：
Pi = 总数 / 相交的针



针在平行线间的位置





波斯公主选驸马

- 波斯公主到了适婚年龄，要选驸马。候选男子100名，都是公主没有见过的。
- 百人以随机顺序，从公主面前逐一经过。每当一位男子在公主面前经过时，公主要么选他为驸马，要么不选。
- **如果选他**，其余那些还没有登场的男子就都遣散回家，选驸马的活动也 over 了。
- **如果不选**，当下这名男子就离开，也就是 pass 掉此人，下一人登场。
- 被pass 掉的，公主不可以反悔再从选。
- 规则是，公主必须在这百人中选出一人做驸马，也就是说，如果前99人公主都看不中我的话，她必须选择第100名男子为驸马，不管他有多么丑陋。
- 任务是，给公主设计选择方法，让她有最高概率选到百人中最英俊的男子为驸马。



- 最佳选法是 pass 掉最开始的 n 个男子。但是记录下这 n 名男子中最英俊者。
- 之后鱼贯而来的男子中，出现的第一位英俊程度超越所有前 n 人者，即为驸马。
- 如果人都走光了，也没出现这么一位 Mr. Right，那么就只好选择第 100 位男子。



- 在我们所生活的世界上，充满了不确定性



- 需要什么样的类
 - 有什么样的变量
 - 有什么样的方法
- 需要什么样的界面
 - 美观
 - 易实现



几个技术点

- 数学函数

- ▣ Math.Sin, Math.Sqrt

- 字符串连接 使用 + 号

- //字符串与变量连接的三种写法
 - `this.label1.Text = x+"以及"+y;`
 - `this.label1.Text = $"{x}以及{y}";`
 - `this.label1.Text = string.Format("{0}以及{1}", x, y);`



几个技术点

- 随机数

- `Random rnd = new Random();` 一般定义在函数外（类中）
- `double a = rnd.NextDouble();`

- Timer对象（在组件中）

- Interval及Enabled
- Tick事件

- 绘图

- `Graphics g = this.CreateGraphics();`
- `g.DrawLine(Pens.Red, 1, 1, 20, 20);`

- `Graphics g = this.CreateGraphics();`
- `g.DrawLine(Pens.Red, 10, 10, 300, 200);`
- 关于长时间的任务
 - `new System.Threading.Thread(() => {`
 - `for (int i = 0; i < 100; i++)`
 - `{`
 - `System.Threading.Thread.Sleep(100);`
 - `this.BeginInvoke(new Action(() => {`
 - `g.DrawLine(Pens.Red, rnd.Next(300), rnd.Next(300), rnd.Next(300), rnd.Next(300));`
 - `});`
 - `}`
 - `}).Start();`



作业：选一个

- 随机数与数组
 - 投针求pi
 - 投色子
 - 选驸马
 - 23人的生日