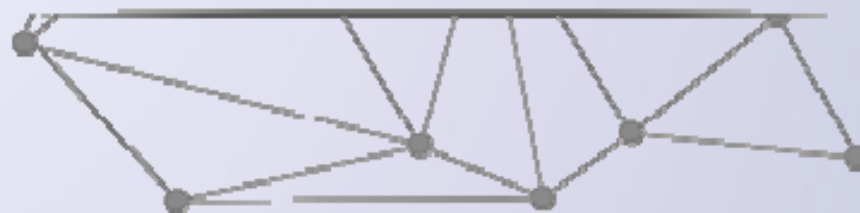


π 的计算



嵩 天

北京理工大学



- 圆周率 π 是一个无理数，没有任何一个精确公式能够计算 π 值， π 的计算只能采用近似算法。
- 国际公认的PI值计算采用蒙特卡洛方法。



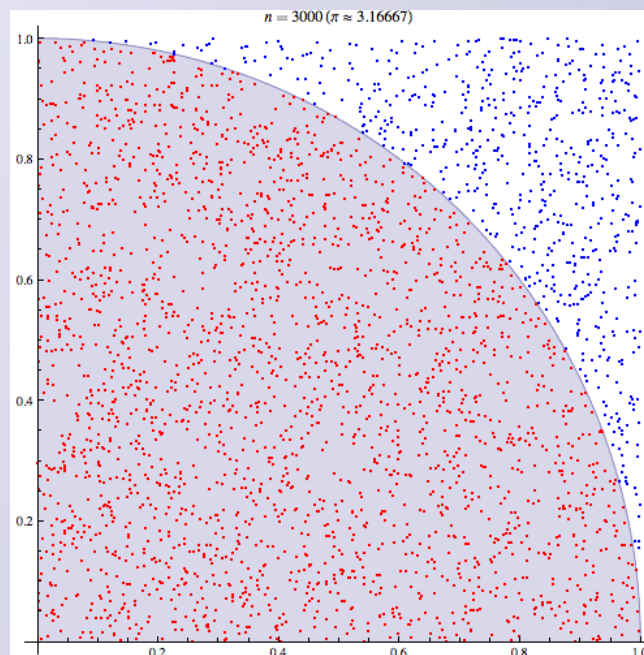


- 蒙特卡洛(Monte Carlo)方法，又称随机抽样或统计试验方法。当所求解问题是某种事件出现的概率，或某随机变量期望值时，可以通过某种“试验”的方法求解。
- 简单说，蒙特卡洛是利用随机试验求解问题的方法。



π 的计算

- 应用蒙瑞特卡罗方法求解 π 的步骤是这样的：
- 首先构造一个单位正方形 和 $1/4$ 圆





- 随机向单位正方形和圆结构抛洒大量点，对于每个点，可能在圆内或者圆外，当随机抛点数量达到一定程度，圆内点将构成圆的面积，全部抛点将构成矩形面积。圆内点数除以圆外点数就是面积之比，即 $\pi/4$ 。随机点数量越大，得到的 π 值越精确。





π 计算问题的IPO表示如下：

- 输入：抛点的数量
- 处理：对于每个抛洒点，计算点到圆心的距离，通过距离判断该点在圆内或是圆外。统计在圆内点的数量
- 输出： π 值





示例程序1： π 的计算

```
# pi.py
from random import random
from math import sqrt
from time import clock
DARTS = 1200
hits = 0
clock()
for i in range(1,DARTS):
    x, y = random(), random()
    dist = sqrt(x**2 + y**2)
    if dist <= 1.0:
        hits = hits + 1
pi = 4 * (hits/DARTS)
print("Pi的值是 %s" % pi)
print("程序运行时间是 %-5.5ss" % clock())
```





示例程序1： π 的计算

- 采用IDLE打开pi.py文件，F5运行该程序，程序运行结果如下：

```
>>>  
Pi的值是 3.1433333333333335  
程序运行时间是 0.013s
```

- 由于DARTS点数量较少， π 的值不是很精确。



示例程序1： π 的计算

■通过增加DARTS数量继续试验，同时，运行时间也逐渐增加。

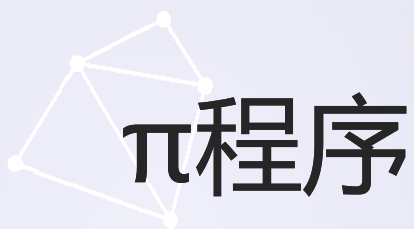
表 3.2 不同抛点数产生的精度和运行时间

| DARTS | π | 运行时间 |
|----------|--------------|---------|
| 2^{10} | 3.109375 | 0.011 s |
| 2^{11} | 3.138671 | 0.012 s |
| 2^{12} | 3.150390 | 0.014 s |
| 2^{13} | 3.143554 | 0.018 s |
| 2^{14} | 3.141357 | 0.030 s |
| 2^{15} | 3.147827 | 0.049 s |
| 2^{16} | 3.141967 | 0.116 s |
| 2^{18} | 3.144577 | 0.363 s |
| 2^{20} | 3.1426696777 | 1.255 s |
| 2^{25} | 3.1416978836 | 40.13 s |



示例程序1： π 的计算

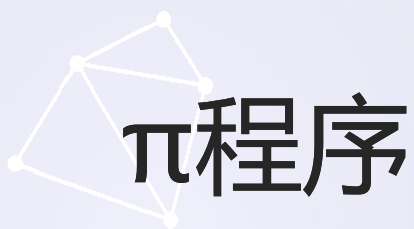
- 在 2^{30} 数量级上， π 的值就相对准确了。
- 进一步增加DARTS数量，能够进一步增加精度。



- 输入部分通过初始变量DARTS给出；
- 输出部分通过print函数打印到屏幕上
- 操作部分主要代码如下：

```
for i in range(1,DARTS):  
    x, y = random(), random()  
    dist = sqrt(x**2 + y**2)  
    if dist <= 1.0:  
        hits = hits + 1  
pi = 4 * (hits/DARTS)
```





- 代码主体是一个循环，模拟抛洒多个点的过程
- 对于一个抛点，通过`random()`函数给出随机的坐标值(x,y)，然后利用开方函数`sqrt()`计算抛点到原点距离
- 然后通过if语句判断这个距离是否落在圆内
- 最终，根据总抛点落入圆内的数量，计算比值，从而得到 π 值





蒙特卡洛方法

- 蒙特卡洛方法提供了一个利用计算机中随机数和随机试验解决现实中无法通过公式求解问题的思路。
- 它广泛应用在金融工程学，宏观经济学，计算物理学（如粒子输运计算、量子热力学计算、空气动力学计算）等领域中。

