6，概率

2018年4月27日 19:43

- 简介
- 题外话
- 面试题总体分析
- 一些例题
  - 例1 关于独立的理解
  - 例2 构造随机数发生器
  - 例3 不均匀随机数发生器构造均匀
  - 例4 随机变量的和
  - 例5 水库采样
  - 例6 随机排列产生——random_shuffle
  - 例7 带权采样问题
- 总结
- 简介
- 概率
  - 对“独立”事件的理解
  - 古典概率 （计数、除法）
  - 条件概率
  - 期望
  - 随机数产生和利用 （采样）*
- 题外话
- 随机数
  - 随机数生成并不容易
    - “随机性”和“不可预测性”
      - 固定m，自然数n % m 是“均匀”的，具有一定随机性，但密码学不采用它

- 一般假设已有一个均匀的随机数生成器

- 期望的计算
  - 一般转化为方程组
    - E(A) = E(A1) * p1 + E(A2) * P2 +…+1

- 面试题总体分析
- 概率（简单）
  - 概率、期望的计算： 笔试
  - 随机数
    - 产生：笔试、面试
    - 利用：采样
  - 相关算法（快排） 面试，具有一定的随机性，期望为nlogn

- 一些例题
  - 例1 关于独立的理解

```
/*
问题： X1,X2,都是二元随机变量，取值0和1的概率各一半，则X3＝ X1 ^ X2,它与X1、X2独立。

为什么这样的呢，因为分析可知，x1为1的时候，x3可能为0,也可能为1,同样的x2也是，所以说它们独立

分析：
因此，分析是否独立不能靠直觉，而要看它们同时出现的概率是否等于单独出现的概率之积。即P(A&B) = P(A)*P(B)
*/

来自 <http://tool.oschina.net/highlight>
```

  - 例2 构造随机数发生器

```
/*
问题： 假设一个随机数发生器rand7均匀产生1到7之间的随机整数，如何构造rand10，均匀产生1-10之间的随机整数？

分析： 关键在于，不想要的数可以扔，但要保证等概率。

方法1：（稍微笨一点的代码）1-7之间有4个奇数三个偶数，丢掉一个奇数，然后就形成3奇3偶的01发生器，
    用其产生4个bit，对应表示整数0-15,保留1-10即可。
```

方法2：（稍微聪明些的）使用7进制，我们把1-7减去1,变为0-6,然后产生一个两位的七进制数，对应0-48,我们把40-48扔掉，
其余按个位数字分类，0-9对应1-10
*/

```
class Solution1
{
public:
    int getBit()
    {
        int x;
        while((x = rand7()) == 7)
        ;//是7的话循环运行
        return x&1;//相当于x%2。奇数返回1,偶数返回0
    }
    int rand10()
    {
        int x;
        do{
            x = 0;
            for(int i = 0;i<4;i++)//循环四次，得到四个getBit，然后把它们接起来
                x = (x<<1)|getBit();
        }while(x<1 || x>10)
        return x;
    }
};

class Solution2
{
public:
    int rand10()
    {
        //代码超级精妙
        while((x = (rand7()-1)*7+rand7()-1)>=40);//40-48扔了，留0-39
        return x%10+1;
    }
};
```

来自 <http://tool.oschina.net/highlight>

○ 例3 不均匀随机数发生器构造均匀

```
/*
```

问题：一个随机数发生器，不均匀，以概率p产生0，以(1-p)产生1， (0<p<1),构造一个均匀的随机数发生器 （算法导论）

分析：将不均匀的随机数发生器发生两次，产生01和10的概率均为p(1-p),是均匀的了

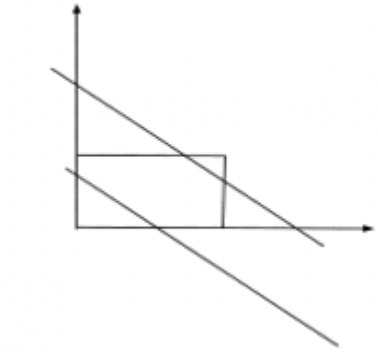
```
*/
class Solution
{
public:
```

```
    int gen()
    {
        int x,y;
        while((x = rand()) == (y=rand()));//产生00,11时继续重新产生。
        return x;//只有x=0， y=1和x=1,y=0两种情况， 等概率。
    }
};
```

来自 <http://tool.oschina.net/highlight>

-

- ## 例4 随机变量的和

- 

```
/*
问题：实数随机变量x和y分别在[0,a]与[0,b]之间均匀分布（a和b是给定的实数），再给一个实数z,问x + y <= z的概率？

分析：x和y分布在图形上为一个矩形，这里是求直线x+y =z下边在矩形内的面积和矩形本身的面积比

*/
```

来自 <http://tool.oschina.net/highlight>

- ## 例5 水库采样

```
/*
问题：水库采样 流入若干个对象（整数），事先不知道个数。如何随机取出k个 （k小于总数）？
```

分析：算法：用一个数组a保存k个数 a[0..k -1]
对于第i个元素(i = 1,2,…)
如果i <= k： 则a[i -1]存放这个元素，（前k个元素，直接放）
否则：产生随机数x = rand() % i，（第i个时，1/i的概率产生一个随机数，随机替换a[0,k-1]范围内的数）
若x < k,则用a[x]存放这个元素（扔掉之前的元素）

证明：，假设目前已经流入n > k个元素，
第i(i <= k)个元素被选中的可能性
明显，最开始我们是要它的，其概率为1,然后每次替换被保留下的可能性为k/k+1,k+1/k+2一直到n-1/n，约分后变成k/n

1 * k / (k + 1) * (k + 1) / (k + 2) *…*(n -1) / n = k / n

第i (i > k)个元素被选中的可能性

什么情况下i会被留下？首先要它的可能性为k/i,以后每次被留下的可能分别为不被替换掉的可能，约分后也是k/n

k / i * i / (i + 1) * (i + 1) / (i + 2) *…* (n - 1) / n = k / n

拓展：k == 1的特殊性

1,用来在一个若干行的大文件中，随机选择一行

2,在一个不知道长度的链表中，随机选择一个or多个元素

*/

```
#include<bits/stdc++.h>
using namespace std;
class Solution
{
public:
    vector<int> poolsampling(vector<int> a,int k)
    {
        vector<int> res;
        int len = a.size();
        for(int i = 0;i<k;i++)//0-k-1直接放
            res.push_back(a[i]);
        for(int i = k;i<len;i++)//k及以后
        {
            int idx = rand()%(i+1);//产生0-i的随机数
            if(idx<k)//随机数在0-k-1范围内，替换，
                res[idx] = a[i];
        }
        return res;
    }
};
int main()
{
    int a[10] = {1,2,3,4,5,6,7,8,9,10};
    vector<int> arr(a,a+10);
    Solution s;
    vector<int> b = s.poolsampling(arr,3);
    for(int i = 0;i<3;i++)
        cout<<b[i]<<endl;
    return 0;
}
```

来自 <http://tool.oschina.net/highlight>

○ 例6 随机排列产生——random_shuffle

/*

问题：用数组a[0..n -1]随机产生一个全排列

分析：方法

```
    for(int i =0;i<n;i++)
        a[i]=i;//赋下初值
    for(int i = 0;i<n;i++)//将a[i]和a[i,n-1]范围内的随机位置交换
        swap(a[i],a[rand()%(n-i)+i]);
*/
```

来自 <http://tool.oschina.net/highlight>

- 例7 带权采样问题

```
/*

问题：带权采样问题 给定n种元素，再给定n个权值，按权值比例随机抽样一个元素。为了方便我们可以假设权值全是整数。

分析:方法1：每份元素复制权值那么多份，然后使用蓄水池抽样n个元素即可

    方法2,每个元素按权值对应一个区间，比如3个a，2个b，6个c，
    a对应[0,2],b对应[3,4],c对应[5,10],随机产生一个0-10的随机数，然后二分查找对应的元素是哪个

    方法3，假设有m中元素，先以1/m的概率随机选择一个元素a，
        第二步再生成一个权值总和大小的随机数，如果该数在wa范围内，则a保留，否则继续重复上述步骤

*/
```

来自 <http://tool.oschina.net/highlight>

- 应用
  - 按照分数给用户推荐歌曲、产品等

- 总结
- 采样
- 概率算法
  - 快速排序 pivot的选择——避免最差情况
  - 在线雇佣问题 （算法导论）
    - 不假设输入分布情况
    - Hash函数解决碰撞
      - 一致性hash
    - 多次尝试

- 如一个算法有一半的可能性得到正确（最优）解，——尝试30次，几乎能得到正确（最优）解