

14, 邹博总结

2018年4月27日 17:09

主要内容

- 古典概型
 - 找零钱/麻将
- 众数
 - 绝对众数
- 链表(部分)逆序
 - 附: 链表去重及其变种
- 金钗赛诗II
 - 带约束的排序
- 附: 猜数字
 - 面试常考类型——逻辑思维能力
- 附: 跳跃问题
 - 贪心/动态规划/广度优先搜索

例1、古典概型

/*

问题1: 现有一叠纸币, 其中5元面值的纸币6张, 10元面值的纸币5张, 20元面值的纸币4张, 从袋子中任意取4张纸币, 则每种面值至少取到一张的概率是多少?

分析: 首先 $c(15, 4)$ 为总的可能数

然后有效的事件有三种, 从5, 10, 20中任选一种面值取两张, 其他两种各取一张

$c(6, 2)*5*4 + c(5, 2)*6*4 + c(4, 2)*6*5$,

然后相除化简即可

附加: “国粹”麻将是集技巧与运气的在我国开展广泛的娱乐项目。去除花牌后的标准麻将, 由1到9的“万、条、饼”各4张, $9 \times 3 \times 4$

以及“东南西北中发白”各4张 (4×7), 共计136张组成。我们把两张内容一样的牌叫一幅“将”。

请问, 庄家起手摸14张牌, 则他起手没有“将”的概率是多少?

此外, 可以算下摸13张牌没有“将”的概率, 摸13张牌没有“风”的概率。

分析: 首先, $c(136, 14)$ 为总的取法数, 其中有效事件数目为:

一共 $3 \times 9 + 7 = 34$ 组牌, 选择某个组, 然后在该组的四张牌中任选一张: $c(34, 14) * 4^{14}$

然后相除, 求约分

*/

例2、绝对众数

/*

问题：已知给定的N个整数存在绝对众数，以最低的是空复杂度计算该绝对众数。

见2_5.cpp

分析：众数出现的次数大于所有其余数出现次数之和。因此，当你每次删除两个不同的数，众数不变为什么呢？如果扔掉一个众数一个非众，不变。扔掉两个非众，仍不变。

整体的思想就是，维护一个x，再来一个y，不同，则都丢掉。相同，则记录x出现的次数

```
int count = 0, x;
for(int i = 0; i < n; i++) //士兵守阵地的思想
{
    if(count == 0) { x = a[i], count = 1; } //count为0了，更新x
    else if(x == a[i]) ++count; //相同，则计数++
    else --count; //不相同，则count--，表示丢掉了x，同时a[i]也没存贮
} //最后x就是我们想要的
```

注意，有些题目要数一下x的出现次数是否真的大于1/2，

拓展题：如何找到出现次数严格大于1/k的数？（众数是1/2）

提示：保留k-1个数，来一个数，和这k-1个数比较，相同的话，对应count+1，不同，每个数出现的次数减1。

如何维持这k-1个数，用hash or map?

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

例3、链表的部分翻转

/*

问题：给定一个链表，翻转该链表从m到n的位置。要求直接翻转而非申请新空间。leetcode 3-3

附加：给定排序的链表，删除重复元素，只保留重复元素第一次出现的结点。leetcode 3-5

附加2：若题目变成：若发现重复元素，则重复元素全部删除，代码应该怎么实现呢？

leetcode 3-6

*/

例4、金钗赛诗II

□ 赛诗会上十二金钗各赋诗一首，12人各自随机挑选一首后，李纨曰：“大家通过两两交换的方式，换回自己的律诗；但要求只能跟我交换”。现已知：

黛玉、宝钗、元春、探春、湘云、妙玉、迎春、惜春、熙凤、巧姐、李纨、可卿

各自拿到的律诗作者为：

熙凤、黛玉、迎春、惜春、湘云、可卿、探春、元春、宝钗、巧姐、妙玉、李纨

□ 试计算至少需要多少次交换，才能使得所有人交换得到自己的律诗？

/*

问题：赛诗会上十二金钗各赋诗一首，12人各自随机挑选一首后，李纨曰：“大家通过两两交换的方式，换回自己的律诗；但要求只能跟我交换”。现已知：

黛玉、宝钗、元春、探春、湘云、妙玉、

迎春、惜春、熙凤、巧姐、李纨、可卿

各自拿到的律诗作者为：

熙凤、黛玉、迎春、惜春、湘云、可卿、

探春、元春、宝钗、巧姐、妙玉、李纨

试计算至少需要多少次交换，才能使得所有人交换得到自己的律诗？

分析：谁拿谁的，这个关系画出来，定能形成环。

比如

十二金钗分别标号为0, 1, 2, ..., 11，其中，李纨为0号。

题目实际上给定了12个整数的一个排列，要求只和0交换，最终形成升序数组，求最少的交换次数。

统计每个“环”的长度L：

若包含0，则该环的最少交换次数为L-1；

若不包含0，则该环的最少交换次数为1+L-1+1 = L+1次

将所有环的交换次数累积即可。

时间复杂度O(N)，空间复杂度O(N)

附加：两个聪明人A和B玩猜数字的游戏。他们在脑门上各贴一个正整数数字，两个数字只相差1，A和B只能看到对方的数而看不到自己的

以下是两人的对话：

A：我不知道

B：我也不知道

A：我知道了

B：我也知道了

上述4句对话结束后，聪明的你帮助A、B推算下，他们的数字各是多少呢？

分析：

若A1, B2, 那么A会怀疑自己是1or3, 说自己不知道，但B看到A，肯定确定自己是2, so, 这种情况pass

若A2, B1, 那么A肯定知道，so pass

若A4, B3, 那么A看到3, 猜测自己是2or4, 说自己不知道，B会看到4, 以为自己是3or5, 所以也说不知道，复杂的逻辑题可以用二叉树(N叉树)做辅助推理，原理是：只要某个结点的两个孩子(所有孩子)都不可能，则这个结点不可能。

注意：两人的说话顺序是有决定作用的，是不对称消息：

A说话，则A是在看到B的内容后做判断，B可根据A的内容在自己的推理树上做剪枝。

*/

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```
class Solution
```

```
{
```

```
public:
```

```
    int exchange(vector<int> a, int n)
```

```
    {
```

```
        vector<bool> visited(n, 0); //是否被访问过
```

```
        int j;
```

```
        int res = (a[0] == 0) ? 2 : 0; //最开始
```

```

    for(int i = 0;i<n;i++)
    {
        if(visited[i] || a[i] == i)
            continue;
        j = a[i];//j为a[i]的值，即黛玉最开始拿的诗，比如最开始为熙凤
        while(j!=i)//如果熙凤拿得诗不是黛玉
        {
            res++;//交换次数++
            visited[j] = true;//被访问过
            j = a[j];//看熙凤的诗
        }
        res+=2;//这么多环，只有一个带李纨的，所以统统加2
    }
    res-=2;//减去带李纨这个环的
    return res;
}
};

int main()
{
    int N = 12;
    vector<int> a;
    for(int i = 0;i<N;i++)
        a.push_back(i);
    random_shuffle(a.begin(), a.end());
    Solution s;
    for(int i = 0;i<N;i++)
        cout<<a[i]<<" ";
    cout<<endl;
    cout<<s.exchange(a, N)<<endl;
    return 0;
}

```

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

例5：Jump

/*

问题：给定非负整数数组，初始时在数组起始位置放置一机器人，数组的每个元素表示在当前位置机器人最大能够跳跃的数目。

它的目的是用最少的步数到达数组末端。例如：给定数组A=[2,3,1,1,2]，最少跳步数目是2，对应的跳法是：2→3→2。

□如：2,3,1,1,2,4,1,1,6,1,7，最少需要几步？

分析：leetcode13.1

*/

一点总结

□ 重视思想，重视动手能力；

■ 计算机是实践学科，掌握至少一门编程语言。

- 提高内涵是赢取所有对战平台的终极法宝。
- 重视教科书上的基本内容；
- 实现无bug的快速排序需要几分钟？
- 掌握内容要全面、透彻。
- 优先队列是什么，教科书上在哪部分介绍的？
- 计算字符串的全排列如何设计算法及代码实现？
- 字符有重复怎么办？
- 递归？非递归？

2016春招公开课 29/32