# Acm 不归路

最近我学到了什么。

Acm的题型：

Acm的题型：

**1、动态规划 2、贪心策略 3、枚举法**

**4、最短路径 5、回溯法 6、最小生成树**

**7、背包问题 8、计算几何学 9、网络流**

**10、欧拉回路 11、大数问题（素数测试）**

**12、启发式搜索 13、近似搜索 14、线段树**

**15、字符串处理 16、数论 17、组合数学**

**学**

**18、杂题、博弈、模拟题 ….**

**1、动态规划 2、贪心策略 3、枚举法**

**4、最短路径 5、回溯法 6、最小生成树**

**7、背包问题 8、计算几何学 9、网络流**

**10、欧拉回路 11、大数问题（素数测试）**

**12、启发式搜索 13、近似搜索 14、线段树**

**15、字符串处理 16、数论 17、组合数学**

**18、杂题、博弈、模拟题 ….**

1：Acm的基本输入输出

While（scanf==m）这种格式

Scanf（n） while（n--）；这样的格式

2：一种写题的递归思想

他的条件；

1：知道这种情况与上一种情况的关系，相当于从上一个条件推出这个条件

2：终止条件，什么时候结束递归

典型的题目：大数取余，上楼梯。

3：简单，常用的STL C++的一些好用的地方。 这个不熟，这些地方需要学习

Vecter不定长度数组。

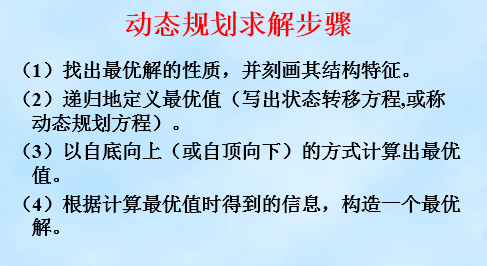
Deque 双端队列

运算符的重载

Bool类型的，一种判断的方法

Set集合。

4：动态规划 这个很重要



一种是状态表示，一种是状态转移

Dp【】

相当于用一个数组来存这些状态，再用递归的思想一个一个的推下去。

题目:背包问题

0 1背包的模版

for(int i=1;i<=n;++i){//第i个物体我可以选择要和不要，取最大值即可

for(int j=m;j>=v[i];--j){

dp[j] = max(dp[j],dp[j-v[i]]+h[i]);

}

}

部分背包问题

for(int i=1;i<=n;++i){

for(int j=1;j<=num[i];++j){

for(int k=v;k>=0;--k){

if(k>=j\*c[i]) dp[k] = max(dp[k],dp[k-j\*c[i]]+k\*w[i])

}

}

}

完全背包

for(int i=1;i<=n;++i){

for(int j=c[i];j<=v;++j){

dp[j] = max(dp[j],dp[j-c[i]]+w[i]);

}

}

数位Dp

相当于找一个区间里满足一些条件的数

Dp【数位的长度】【最高的数位的大小】

用递归来写

5：一些思想：

写一些题目，思想要放开点，很多题目都是有一定的规律，可以从前面的一些具体数据得到一些规律，不要思想局限与强解

6：矩阵快速幂

关键问题，一个矩阵的乘法，一个是矩阵的降幂的算法，主要就是从这两个方面来解决。