线性模型

主讲人：邓哲也

线性模型

这里的线性指的是状态的排布是呈线性的。

线性模型是动态规划中最常用的模型。

最长单调上升子序列就是经典的线性模型。

线性模型

最长上升子序列问题。

f[i] 表示以 a[i] 结尾的最长上升子序列问题。

计算 f[i] 的时候，只需要去考虑 f[1], f[2], ..., f[i-1] 看是否能够转移即可。

$f[i] = \max\{ f[j] + 1 \mid 1 \le j < i, a[j] < a[i] \}$

POJ 3486 Computers

你想保证 n 年中你都有一台电脑，一开始你有一台。

如果你在第 y (1≤y≤n) 年购买了一台电脑，那么你需要花费 c 的代价。

如果你这台电脑一直用到了第 z 年，在第 z 年又买了一台新的，你需要支付 m(y, z) 的维修费用。

给定 n，c，数组 m。求最小花费。

线性模型

样例输入

3 3

5 7 50

6 8

10

样例输出

19

第一年买一台电脑，花费3，

之前的维修费用为m(1,1)=5

第三年买一台电脑，花费3，

之前的维修费用为m(2,3)=8

最小花费是3+5+3+8=19

线性模型

首先划分阶段。

每一年可以划分为一个阶段。

f[i] 表示直到第 i 年f[0], f[1],...,f[i-1]你手里都有一台电脑的最小花费。

f[i] 需要从 转移过来。

如何转移？

枚举上一次买电脑是哪一年！

线性模型

假设上一次买电脑是第 j 年。

那么1~j-1年就是一个子问题，我们已经算出了 f[j-1] 是满足这个子问题的最优解，后面我们就不用考虑前 j-1年的情况，且它也不会影响我们后面的决策。

第 j 年到第 i 年的维修费用是 m(j, i)，花费是 c

因此可以用 f[j-1]+m(j,i)+c 来更新 f[i]

$f[i] = \min\{ f[j-1]+m(j,i) + c \mid 1 \le j \le i \}$

线性模型

边界条件：

$f[0] = 0$

$f[1], f[2], ..., f[n]$ 一开始都应该初始化为 $+\infty$

代码实现

```
memset(f, 0x3f, sizeof(f));
f[0] = 0;
for (int i = 1;i <= n;i ++)
	for (int j = 1;j <= i;j ++)
		f[i] = min(f[i], f[j-1]+m[j][i]+c);
printf(“%d\n”, f[n]);
```

时间复杂度 $O(n^2)$

空间复杂度 $O(n)$

下节课再见