# 状态DP

04计算机-肖锋

# 动态规划(Dynamic Programming)

- 动态规划是解决多阶段决策问题的一种思想方法
  - 阶段性：原问题的解决过程可以划分成一系列子问题，通过逐个解决子问题来得到原问题的解
  - 最优子结构：原问题是最优当且仅当子问题最优
  - 无后效性：原问题的解只与子问题的解有关，而与得到子问题的过程无关

# 动态规划(Dynamic Programming)

- 阶段性
  - 原问题的解决过程可以划分成一系列子问题，通过逐个解决子问题来得到原问题的解

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 5 | 2 | 2 |
| 5 | 1 | 3 | 5 | 4 |
| 3 | 2 | 2 | 5 | 2 |
| 2 | 5 | 1 | 1 | 2 |
| 2 | 6 | 3 | 4 | 3 |

# 动态规划(Dynamic Programming)

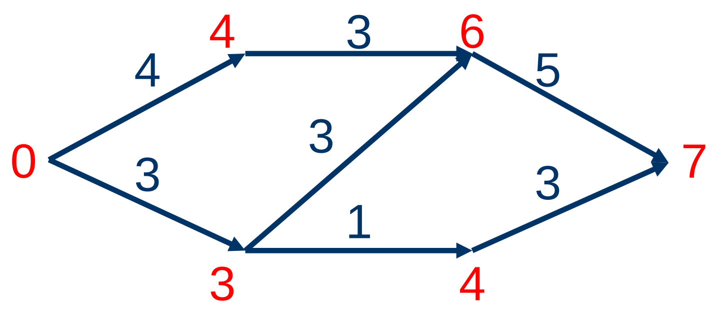
- 阶段性

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 5 | 2 | 2 |
| 5 | 1 | 3 | 5 | 4 |
| 3 | 2 | 2 | 5 | 2 |
| 2 | 5 | 1 | 1 | 2 |
| 2 | 6 | 3 | 4 | 3 |

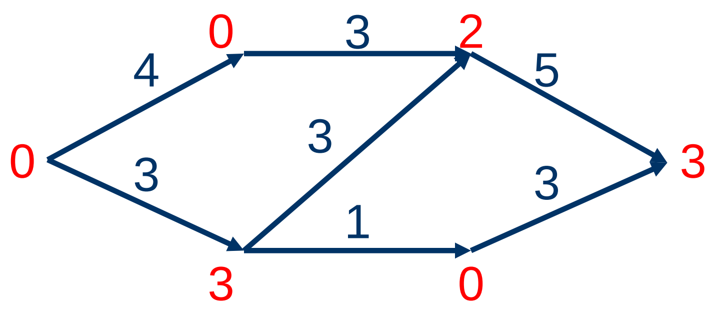
| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 1 | 3 | 8 | … | |
| 2 | 6 | 4 | … | | |
| 3 | 9 | … | | | |
| 4 | … | | | | |
| 5 | | | | | ? |

# 动态规划(Dynamic Programming)

- 最优子结构
  - 原问题是最优当且仅当子问题最优

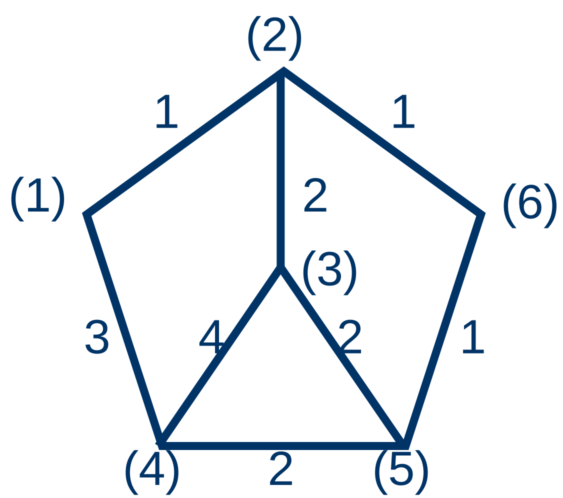



路径长度模4最小?



# 动态规划(Dynamic Programming)

- 无后效性
  - 原问题的解只与子问题的解有关，而与得到子问题的过程无关




| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 0 | 0 | inf | inf | inf | inf | Inf |
| 1 | inf | 1 | inf | 3 | inf | Inf |
| 2 | … | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |

不允许经过重复结点？

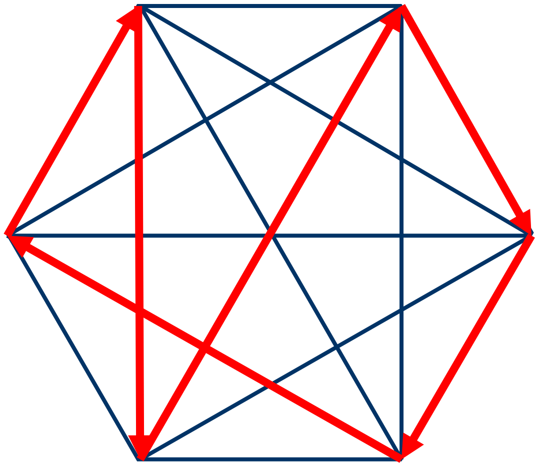
# 动态规划(Dynamic Programming)

- 状态表示和状态转移
  - 状态表示方案要满足最优子结构和无后效性
- 递推与记忆化搜索
  - 递推：可通过滚动数组等手段优化
  - 搜索：简单、自顶向下剪枝、避免无用状态

# 状态DP

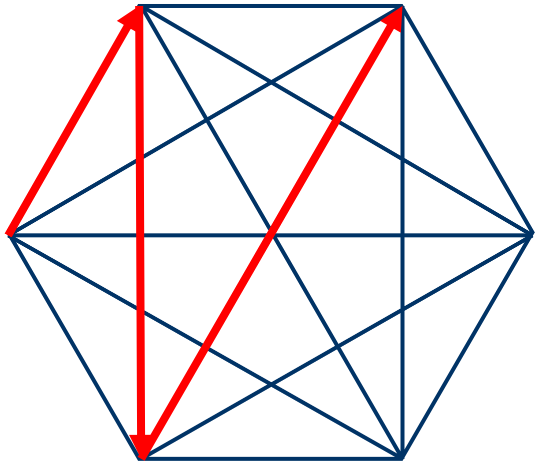
- 基于状态压缩的动态规划
- 引例：哈密顿回路问题

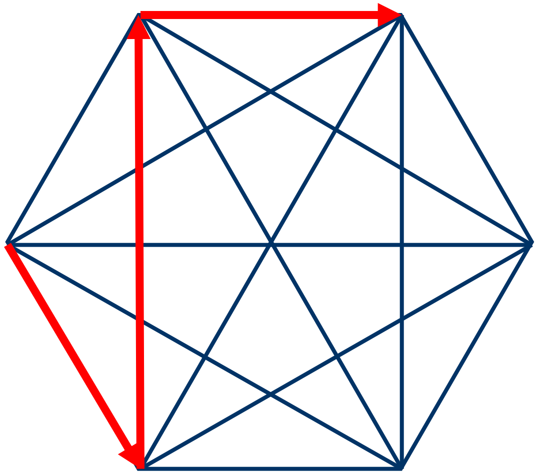
# 引例：哈密顿回路问题



# 引例：哈密顿回路问题

- 搜索+剪枝





# 引例：哈密顿回路问题

- 当前状态：起点、终点、经过的点的集合
  - 用位压缩的办法表示经过的点的集合，则每种状态都可以用三个整数表示，不妨记为<a, b, st>，其中st表示经过的点的集合

$$f(<a,b,st>)=\min\{\ f(<a,k,st-2^{b}>)+w(k,b)\},k\in \text{集合}st$$

# 引例：哈密顿回路问题

- 状态DP
  - 最优子结构
  - 无后效性
- 复杂度
  - 状态数O(n^2*2^n)，状态转移O(n)
  - 时间复杂度O(n^3*2^n)，空间复杂度 O(n^2*2^n)

# 状态DP

- 基于状态压缩的动态规划
  - 位压缩的状态表示方法
  - 状态数目是指数级别的，但相对于搜索而言仍不失为一种高效算法
  - 状态转移过程需要仔细考虑

# 例题

- POJ1691 Painting A Board
- POJ2836 Rectangular Covering
- POJ2285 The Floor Bricks

# POJ1691 Painting A Board

- 状态表示
  - 已经刷好的矩形和当前刷子的颜色
- 状态转移
  - 刷一面颜色和刷子颜色一样的矩形
  - 换刷子

# POJ2836 Rectangular Covering

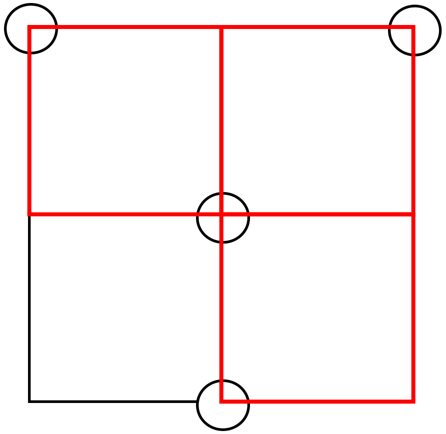
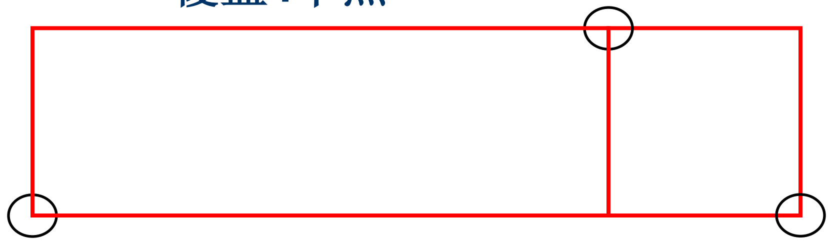


图5

一个大矩形划分为更小的矩形的例子：
划分后得到的小矩形的面积和必小于等于原始大矩形的面积，并且小矩形覆盖的点都在小矩形的顶点上。

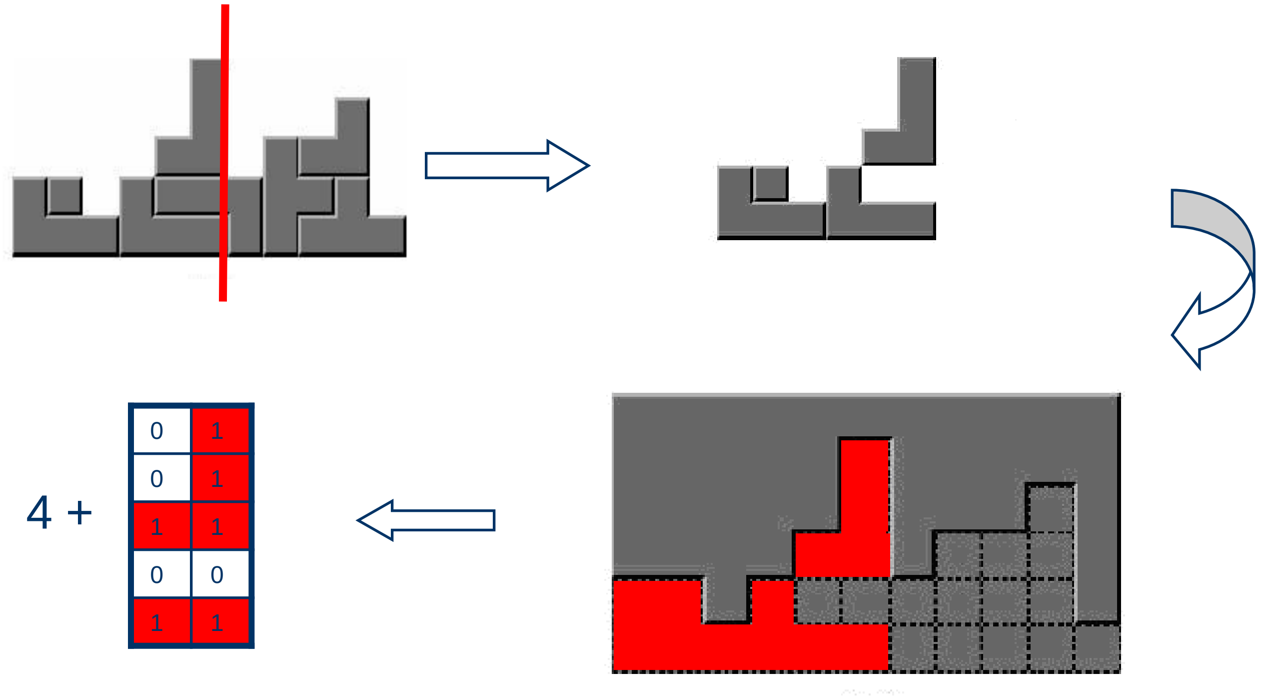
# POJ2836 Rectangular Covering

覆盖4个点



覆盖3个点

# POJ2285 The Floor Bricks



# 状态DP

- 总结
  - 基于位压缩的动态规划
  - 解决的问题规模小
  - 容易设计出状态表示方案，状态转移的过程需要仔细考虑
  - 编写代码的“复杂度”较高，需要大量练习