# 石子合并问题

## 无条件的合并问题

有N堆石子，每堆有A[I]个，每次可以合并M堆 (M<N)成新的一堆，合并一次花费是M堆石子价值总和，问最小花费。

贪心的问题，哈夫曼编码变形，只要用一个优先队列，每次取最小的M个加完了再放回去，求最终合并只有一堆了的结果

**变形：**有N堆石子，每堆有A[I]个，每次可以合并最多M堆 (M<N)成新的一堆，合并一次花费是M堆石子价值总和，问最小花费。

和上面一样，且贪心的每次尽可能多的选取石子，直到不够M堆，才选小于M堆

## 每次的合并相邻得

设A[i]为第i堆石子价值，sum[i]为前i堆之和，sum[i][j]代表[I,j]范围石子A[i]之和

### 1 每次合并2堆

设dp[i][j]代表从i到j合并花费的最小值，

显然只有1堆的时候:dp[i][i]=0;

有2堆的时候:dp[i][i+1]两堆花费是A[i]+A[i+1] =sum[i,i+1]

有3堆的时候:dp[i][i+2]=

min(dp[i][i+1]+dp[i+2][i+2]+sum[i][i+2],dp[i+1][i+2]+dp[i][i]+sum[i][i+2])

有4堆的时候:dp[i][i+3]=min(

dp[i][i+2]+dp[i+3][i+3]+sum[i][i+3],

dp[i][i+1]+dp[i+2][i+3]+sum[i][i+3],

dp[i][i]+dp[i+1][i+3]+sum[i][i+3],)

那么dp[i][j]中设j=i+h;

dp[i][j]=dp[i][i+h]=min(dp[i][i+0]+dp[i+0+1][i+h],sum[i][i+h], dp[i][i+1]+dp[i+1+1][i+h],sum[i][i+h],dp[i][i+2]+dp[i+2+1][i+h],sum[i][i+h]….., dp[i][i+h-1]+dp[i+h][i+h],sum[i][i+h]);

dp[i][j]=min(k=0;i+k<j;k++ | dp[i][i+k]+dp[i+k+1][j])+sum[i][j];

从这些取个最小的，那么时间复杂度是n^3,太大而且可以优化

**优化时间：**

**四边形优化：**

对于循环对于 f[i][j]= min{f[i][k]+f[k+1][j]+w[i][j]} k∈[i,j]来说

w[i][j]代表区间[i,j]的价值，在石子合并问题里代表从i到j的价值，也就是sum[i][j]

设s[i][j]代表f[i][j]= min{f[i][k]+f[k+1][j]+w[i][j]}里最优的k

凸四边形不等式：w[a][c]+w[b][d]<=w[b][c]+w[a][d]（a<b<c<d）

区间包含关系单调: w[b][c]<=w[a][d]（a<b<c<d）

定理1：  如果w同时满足四边形不等式和决策单调性 ,则f[i][j]也满足四边形不等式

定理2：若f满足四边形不等式和决策单调性，则s单调递增

即：s[i][j-1]<=s[i][j]<=s[i+1][j]

定理3： w为凸当且仅当w[i][j]+w[i+1][j+1]<=w[i+1][j]+w[i][j+1]