

# 知识精炼（二）



主讲人：邓哲也



## HDU 6149 Valley Number II

给定一张  $N$  个点  $M$  条边的无向图，其中有  $K$  个点被标记为高点，剩下的  $(N-K)$  个点是低点。

图中的山谷定义为三元组  $\langle X, Y, Z \rangle$ ，满足  $X$  和  $Y$  之间有边， $Y$  与  $Z$  之间也有边，同时  $X$  和  $Z$  是高点， $Y$  是低点。

问这个图中最多有几个山谷（一个点只能出现在一个山谷中）

$N \leq 30, K \leq \min(N, 15)$

# HDU 6149 Valley Number II

样例1:

3

3 2 2

1 2

1 3

2 3

答案: 1

样例2:

3 2 2

1 2

1 3

1 2

答案: 0

样例3:

7 6 5

1 2

1 3

1 4

2 3

2 6

2 7

3 4 5 6 7

答案: 2

## HDU 6149 Valley Number II

高点最多只有 15 个。

可以考虑用状态压缩， $s$  表示高点的使用状态。

低点排成一个长度为  $n-k$  的序列。

$f[i][s]$  表示前  $i$  个低点，使用过的高点的状态为  $s$  的情况下，组成的山谷的最大可能值。

## HDU 6149 Valley Number II

转移  $f[i][s]$  的时候, 取出第  $i+1$  个低点。

枚举不在  $s$  中的两个高点  $p$  和  $q$ 。

检查  $p$  和  $q$  和第  $i+1$  个低点能否配对。

如果可以, 那么就可以用  $f[i][s] + 1$  去更新

$f[i + 1][s \mid (1 \ll p) \mid (1 \ll q)]$

答案就是  $\max\{ f[n-k][i] \mid 0 \leq i < 2^k \}$

## HDU 6149 Valley Number II

```
scanf("%d%d%d", &n, &m, &k);
for (int i = 1; i <= m; i++) {
    scanf("%d%d", &x, &y);
    e[x][y] = e[y][x] = 1;
}
for (int i = 1; i <= n; i++) {
    scanf("%d", &high[i]);
    ishigh[high[i]] = 1;
}
int tot = 0;
for (int i = 1; i <= n; i++) {
    if (!ishigh[i]) {
        low[++tot] = i;
    }
}
```

## HDU 6149 Valley Number II

```
vector < pair<int,int> > trans[35];

for (int i = 1;i <= tot;i ++){
    trans[i].clear();
    for (int p = 1;p <= k;p ++){
        if (e[low[i]][high[p]])
            for (int q = p + 1; q <= k; q ++){
                if (e[low[i]][high[q]])
                    trans[i].push_back(make_pair(p, q));
            }
    }
}
```

## HDU 6149 Valley Number II

```
for (int i = 0; i < tot; i++) {
    for (int s = 0; s < (1 << k); s++) {
        f[i + 1][s] = max(f[i + 1][s], f[i][s]);
        for (int j = 0; j < trans[i].size(); j++) {
            int p = trans[i][j].first;
            int q = trans[i][j].second;
            if (s >> (p - 1) & 1) continue;
            if (s >> (q - 1) & 1) continue;
            f[i + 1][s | (1 << (p - 1)) | (1 << (q - 1))] =
max(f[i + 1][s | (1 << (p - 1)) | (1 << (q - 1))], f[i][s] + 1);
        }
    }
}
```



## HDU 6149 Valley Number II

```
int ans = 0;
for(int i = 0; i < (1 << k); i++)
    ans = max(ans, f[tot][i]);
printf("%d\n", ans);
```

下节课再见