专题4

目录

1001 Machine Schedule

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[510][510];/*连接关系*/
int vis[510];/*右边点使用与否*/
int link[510];/*右边点连接的左边点*/
int n, m, k;
int dfs(int x)
{
    for (int y = 0; y < m; y++) {
        if (!vis[y] && con[x][y]) {
            vis[y] = 1;
            if (dfs(link[y])||link[y]==-1) {
                link[y] = x;
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
```

```
    memset(link, -1, sizeof(link));
    for (int x = 0; x < n; x++) {
        memset(vis, 0, sizeof(vis));
        if (dfs(x))res++;
    }
    return res;
}
int main()
{
    while (scanf("%d", &n), n) {
        scanf("%d%d", &m, &k);
        int id, x, y;
        memset(con, 0, sizeof(con));
        for (int i = 0; i < k; i++) {
            scanf("%d%d%d", &id, &x, &y);
            if (x && y)con[x][y] = 1;
        }
        printf("%d\n", hungary());
    }
    return 0;
}
```

1002 Air Raid

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[150][150];/*连接关系*/
int vis[150];/*右边点使用与否*/
int link[150];/*右边点连接的左边点*/
int n, m, k;/*如题意*/
int dfs(int x)/*用以判断找不找得到结尾为之前未连的边的情况*/
{
    for (int y = 1; y <= m; y++) {
        if (!vis[y] && con[x][y]) /*代表右边这个点在此轮没有使用过且可有连线*/
        {
            vis[y] = 1;/*标注已使用*/
            if (dfs(link[y]) || link[y] == -1) /*代表要么这个点并未连过或者它的连线的点能找到另一个有未连的边的点……*/
            {
                link[y] = x;/*找到就重新连接*/
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));/*注意连线是需要存储的，不然下一步就没有方向了*/
    for (int x = 1; x <=n; x++) {
        memset(vis, 0, sizeof(vis));/*注意这是每轮清空，表示需要抢之前已经匹配的*/
```

```
        if (dfs(x))res++;/*成功一次多一个匹配数*/
    }
    return n-res;
}
int t;
int main()
{
    scanf("%d", &t);
    while (t--) {
        scanf("%d%d", &n, &k);
        m = n;
        int x, y;
        memset(con, 0, sizeof(con));
        for (int i = 1; i <=k; i++) {
            scanf("%d%d", &x, &y);
            con[x][y] = 1;
        }
        printf("%d\n", hungary());
    }
    return 0;
}
```

1003 棋盘游戏

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[110][110];
int vis[110];
int link[110];
int n, m;
int find(int x)
{
    for (int i = 1; i <= m; i++)
    {
        if (vis[i] == 0 && con[x][i] == 1)
        {
            vis[i] = 1;
            if (link[i] == -1 || find(link[i]))
            {
                link[i] = x;
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int x = 1; x <= n; x++) {
        memset(vis, 0, sizeof(vis));
```

```
        if (find(x))res++;
    }
    return res;
}
int x[10010], y[10010];/*注意这里数组大小是k的范围*/
int main()
{
    int k;
    int count = 1;
    while (scanf("%d%d%d", &n, &m, &k) == 3) {
        memset(con, 0, sizeof(con));
        for (int i = 1; i <= k; i++) {
            scanf("%d%d", &x[i], &y[i]);
            con[x[i]][y[i]] = 1;
        }
        int sum = hungary();
        int cnt = 0;
        for (int i = 1; i <= k; i++) {
            con[x[i]][y[i]] = 0;
            if (hungary() < sum)cnt++;
            con[x[i]][y[i]] = 1;
        }
        printf("Board %d have %d important blanks for %d chessmen.\n",
count++, cnt, sum);
    }
    return 0;
}
```

1004 50 years,50 colors

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[110][110];
int vis[110];
int link[110];
int n;
int find(int x)
{
    for (int i = 1; i <= n; i++)
    {
        if (vis[i] == 0 && con[x][i] == 1)
        {
            vis[i] = 1;
            if (link[i] == -1 || find(link[i]))
            {
                link[i] = x;
                return 1;
            }
        }
    }
    return 0;
}
```

```
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int x = 1; x <= n; x++) {
        memset(vis, 0, sizeof(vis));
        if (find(x))res++;
    }
    return res;
}
int Map[110][110];
int visc[55];
int main()
{
    int k;
    int count = 1;
    while (scanf("%d%d", &n,&k),n,k) {
        memset(con, 0, sizeof(con));
        memset(visc, 0, sizeof(visc));
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= n; j++) {
                scanf("%d", &Map[i][j]);
                visc[Map[i][j]] = 1;
            }
        }
        int flag = 1;
        for (int q = 1; q <= 50; q++) {
            if (!visc[q])continue;
            memset(con, 0, sizeof(con));
            for(int i=1;i<=n;i++)
                for (int j = 1; j <= n; j++) {
                    if (Map[i][j] == q)con[i][j] = 1;
                }
            if (hungary() > k) {
                if (flag)printf("%d", q);
                else printf(" %d", q);
                flag = 0;
            }
        }
        if (flag)printf("-1");
        printf("\n");
    }
    return 0;
}
```

1005 Card Game Cheater

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[110][110];
int vis[110];
int link[110];
```

```cpp
int n;
int t;
int find(int x)
{
    for (int i = 1; i <= n; i++){
        if (vis[i] == 0 && con[x][i] == 1){
            vis[i] = 1;
            if (link[i] == -1 || find(link[i])){
                link[i] = x;
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int x = 1; x <= n; x++) {
        memset(vis, 0, sizeof(vis));
        if (find(x))res++;
    }
    return res;
}

int getScore(string str) {
    int ans = 0;
    if (str[0] >= '2' && str[0] <= '9')
        ans += (str[0] - '0') * 10;
    else if (str[0] == 'T')
        ans += 100;
    else if (str[0] == 'J')
        ans += 110;
    else if (str[0] == 'Q')
        ans += 120;
    else if (str[0] == 'K')
        ans += 130;
    else if (str[0] == 'A')
        ans += 140;
    if (str[1] == 'C')
        ans += 1;
    else if (str[1] == 'D')
        ans += 2;
    else if (str[1] == 'S')
        ans += 3;
    else if (str[1] == 'H')
        ans += 4;
    return ans;
}

int main()
{
    scanf("%d", &t);
    while (t--) {
        memset(con, 0, sizeof(con));
        scanf("%d", &n);
        string str;
```

```
        int scoreAdam[2000], scoreEve[2000];
        for (int i = 1; i <= n; i++) {
            cin >> str;
            scoreAdam[i] = getScore(str);
        }
        for (int i = 1; i <= n; i++) {
            cin >> str;
            scoreEve[i] = getScore(str);
        }
        for (int i = 1; i <= n; i++)
            for (int j = 1; j <= n; j++)
                if (scoreEve[i] > scoreAdam[j])
                    con[i][j] = 1;
        printf("%d\n", hungary());
    }
    return 0;
}
```

1006 Uncle Tom's Inherited Land*

未AC

1007 Girls and Boys

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[510][510];
int vis[510];
int link[510];
int n, m;
int find(int x)
{
    for (int i = 0; i < n; i++)
    {
        if (vis[i] == 0 && con[x][i] == 1)
        {
            vis[i] = 1;
            if (link[i] == -1 || find(link[i]))
            {
                link[i] = x;
                return 1;
            }
        }
    }
    return 0;
}

int main()
{
    while (scanf("%d", &n)!=EOF) {
```

```
        m = n;
        memset(con, 0, sizeof(con));
        memset(link, -1, sizeof(link));
        int x,p,y;
        for (int i = 0; i < n; i++) {
            scanf("%d: (%d)", &x, &p);
            while (p--) {
                scanf("%d", &y);
                con[x][y] = 1;
            }
        }
        int res = 0;
        for (int x = 0; x < n; x++) {
            memset(vis, 0, sizeof(vis));
            if (find(x))res++;
        }
        printf("%d\n", n - res / 2);
    }
    return 0;
}
```

1008 Fire Net

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[10][10];
int vis[10];
int link[10];
int n;
char Map[10][10];
int find(int x)
{
    for (int i = 0; i < n*2; i++){
        if (vis[i] == 0 && con[x][i] == 1){
            vis[i] = 1;
            if (link[i] == -1 || find(link[i])){
                link[i] = x;
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int x = 0; x < n*2; x++) {
        memset(vis, 0, sizeof(vis));
        if (find(x))res++;
    }
    return res;
```

```
}


int main()
{
    while (~scanf("%d", &n) && n) {
        memset(con, 0, sizeof(con));
        for (int i = 0; i < n; ++i) {
            scanf("%s", Map[i]);
        }
        for (int i = 0; i < n; ++i) {
            int flag = 0;
            int flag1 = 0;
            for (int j = 0; j < n; ++j) {
                if (Map[i][j] == '.' && flag == 1) {
                    flag = 0;
                    con[n + i][j] = 1;
                }
                else if (Map[i][j] == '.' && flag == 0) {
                    con[i][j] = 1;
                }
                else if (Map[i][j] == 'X') {
                    flag = 1;
                }
                if (Map[j][i] == '.' && flag1 == 1) {
                    flag1 = 0;
                    con[j][i + n] = 1;
                }
                else if (Map[j][i] == '.' && flag1 == 0) {
                    con[j][i] = 1;
                }
                else if (Map[j][i] == 'X') {
                    flag1 = 1;
                }
            }
        }
        printf("%d\n", hungary());
    }
    return 0;
}
```

1009 Courses

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[305][305];
int vis[305];
int link[305];
int n, m;

int find(int x)
{
```

```
    for (int i = 1; i <=m; i++){
        if (vis[i] == 0 && con[x][i] == 1){
            vis[i] = 1;
            if (link[i] == -1 || find(link[i])){
                link[i] = x;
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int x = 1; x <= n; x++) {
        memset(vis, 0, sizeof(vis));
        if (find(x))res++;
    }
    return res;
}

int t;
int k;
int main()
{
    scanf("%d", &t);
    while (t--) {
        memset(con, 0, sizeof(con));
        scanf("%d%d", &n, &m);
        int g;
        for (int i = 1; i <= n; i++) {
            scanf("%d", &g);
            int y;
            while (g--) {
                scanf("%d", &y);
                con[i][y] = 1;
            }
        }
        if (hungary() == n)printf("YES\n");
        else printf("NO\n");
    }
    return 0;
}
```

1010 过山车

AC代码

```
#include<bits/stdc++.h>
using namespace std;
int con[510][510];
int vis[510];
int link[510];
```

```cpp
int n, m, k;
int dfs(int x)
{
    for (int y = 1; y <= m; y++) {
        if (!vis[y] && con[x][y])
        {
            vis[y] = 1;
            if (dfs(link[y]) || link[y] == -1)
            {
                link[y] = x;
                return 1;
            }
        }
    }
    return 0;
}
int hungary()
{
    int res = 0;
    memset(link, -1, sizeof(link));
    for (int x = 1; x <=n; x++) {
        memset(vis, 0, sizeof(vis));
        if (dfs(x))res++;
    }
    return res;
}
//int t;
int main()
{
    while (scanf("%d",&k),k) {
        scanf("%d%d", &n, &m);
        int x, y;
        memset(con, 0, sizeof(con));
        for (int i = 1; i <=k; i++) {
            scanf("%d%d", &x, &y);
            con[x][y] = 1;
        }
        printf("%d\n", hungary());
    }
    return 0;
}
```