

算法设计与分析 循环赛日程表

班级：2015211312

学号：2015211484

姓名：刘佳鑫

● 问题描述：

- 设有 n 个运动员要进行网球循环赛。设计一个满足下列条件的比赛日程表：
 - 每个选手必须与其他 $n-1$ 个选手各赛一次；
 - 每个选手一天只能赛一次；
 - 当 n 是偶数时，循环赛进行 $n-1$ 天。
 - 当 n 是奇数时，循环赛进行 n 天

● 算法描述及思考过程：

根据分治法，思想借鉴课上讲的复制表格方法，偶数选手按课上算法复制表格。奇数参考偶数的算法，开始想的是加一个人，之后从参考了下网上的算法，选择了加一个虚拟对手，然后处理时在把和虚拟对手 pk 的置零，则选手在那天可休息一天。

第一版算法是奇数偶数复制表格时都用同样的方法，只是奇数要有置零操作。但是运行后发现在 $n/2$ 为奇数时，如 $n=6$ 时会出现错误，有比不完的现象。后来看了一下网上他人的算法，发现奇数复制表格时不能和偶数相同，要在置零后判断如果两个都置零，则安排他们比赛。而且在置零后也不能直接同偶数一样根据此位置数字安排后面的比赛。所以又新加了一个专门复制奇数选手表格的函数。

● 源代码：

```
#include<iostream>

using namespace std;

void deal(int m,int A[][100])//第 m 号为虚拟选手 虚拟选手置零函数
{
    int i,j;
    for(i=0;i<m-1;i++)
    {
        for(j=0;j<=m;j++)
        {
            if(A[i][j]==m)
            {
                A[i][j]=0;//把和虚拟选手比赛置为 0
            }
        }
    }
}
```

```

void evenTable(int m,int A[][100])//偶数填充表格
{
    int i,j;
    for(j=0;j<m;j++)
        {
            for(i=0;i<m;i++)
                {
                    A[i+m][j]=A[i][j]+m;//第二对的安排（分配左下角）
                }
        }
    for(j=m;j<2*m;j++)
        {
            for(i=0;i<m;i++)
                {
                    A[i][j]=A[i+m][j-m]);//左下角放到右上角
                }
            for(i=m;i<2*m;i++)
                {
                    A[i][j]=A[i-m][j-m]);//左上角放到右下角
                }
        }
}

```

```

void oddTable(int m,int A[][100])//奇数填充表格
{
    int i,j;
    for(j=0;j<=m;j++)
        {
            for(i=0;i<=m;i++)
                {
                    if(A[i][j]!=0)//不是和虚拟对手 pk 时
                        {
                            A[i+m][j]=A[i][j]+m;
                        }
                    else//都是和虚拟对手 pk 时，相互补上
                        {
                            A[i+m][j]=i+1;
                            A[i][j]=i+m+1;
                        }
                }
        }
    i=0;
    for(j=m+1;j<2*m;j++)
        {
            A[i][j]=j+1;
            A[(A[i][j]-1)][j]=i+1;
        }
    for(i=1;i<m;i++)//复制
        {
            for(j=m+1;j<2*m;j++)
                {
                    if((A[i-1][j]+1)%m==0)
                        A[i][j]=A[i-1][j]+1;
                    else
                        A[i][j]=m+(A[i-1][j]+1)%m;//规律：循环递增
                }
        }
}

```

```

        A[(A[i][j]-1)][j]=i+1;
    }
}
}

```

```

void play(int n,int A[][100])
{
    if(n==1)
        {A[0][0]=1;

        }
    else{

        if(n%2==1)
            {
                play(n+1,A);

                deal(n+1,A);

            }
        else
            {
                play(n/2,A);
                if((n/2)%2==0)
                    evenTable(n/2,A);
                else
                    oddTable(n/2,A);//复制表格

            }
    }
}
}

```

```

void print(int A[][100],int row,int col)//打印
{
    for(int i=0;i<row;i++)
    {
        for(int j=0;j<col;j++)
            {
                cout<<A[i][j]<<" ";
            }
        cout<<endl;
    }
}
}

```

```

main()
{   int num;
    cout<<"请输入比赛人数 (小于 100) "<<endl;

    cin>>num;
    if(num>=100||cin.fail())
    {   cout<<"您的输入有误，请重新输入"<<endl;
        cout<<"请输入比赛人数 (小于 100) "<<endl;
        cin.clear();
        cin.ignore();
        cin>>num;
    }
    int game[100][100];

    for(int i=0;i<num;i++)
    {   game[i][0]=i+1;//初始化第一列
    }

    play(num,game);

    if(num%2==0)
    {   print(game,num,num);
    }
    if(num%2==1)
    {   print(game,num,num+1);//奇数是 n 天，所以多出一列
    }

    system("pause");
    return 0;
}

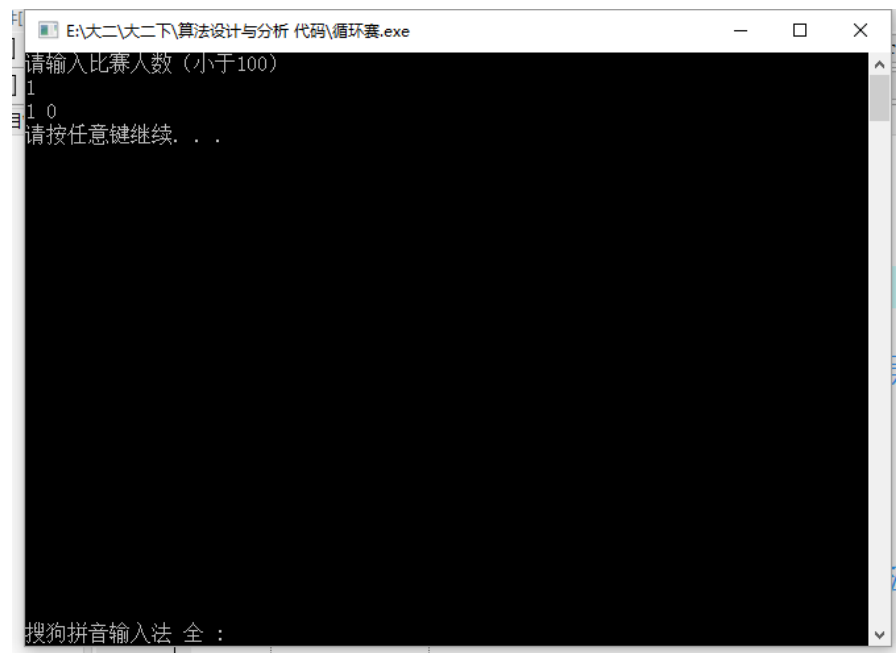
```

● 实验结果说明：

第一列为比赛的一方，同行的第 j 列为第 j-1 天和此人 pk 的对手。若 pk 对手处显示为 0，则此人当天休息。

● 运行结果：

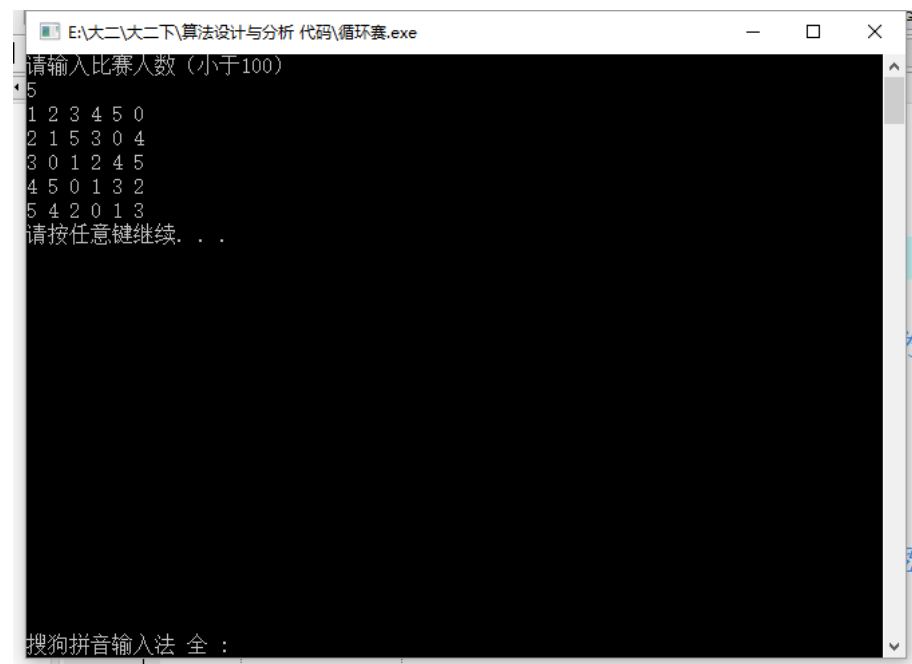
1. $n=1$ 时，即只有一个人没有对手时



The screenshot shows a console window titled "E:\大二\大二下\算法设计与分析 代码\循环赛.exe". The prompt "请输入比赛人数 (小于100)" is displayed. The user has entered "1", and the program has output "1 0" and "请按任意键继续. . .".

```
E:\大二\大二下\算法设计与分析 代码\循环赛.exe
请输入比赛人数 (小于100)
1
1 0
请按任意键继续. . .
```

2. $n=5$ 时，主要测验 n 为奇数时的结果

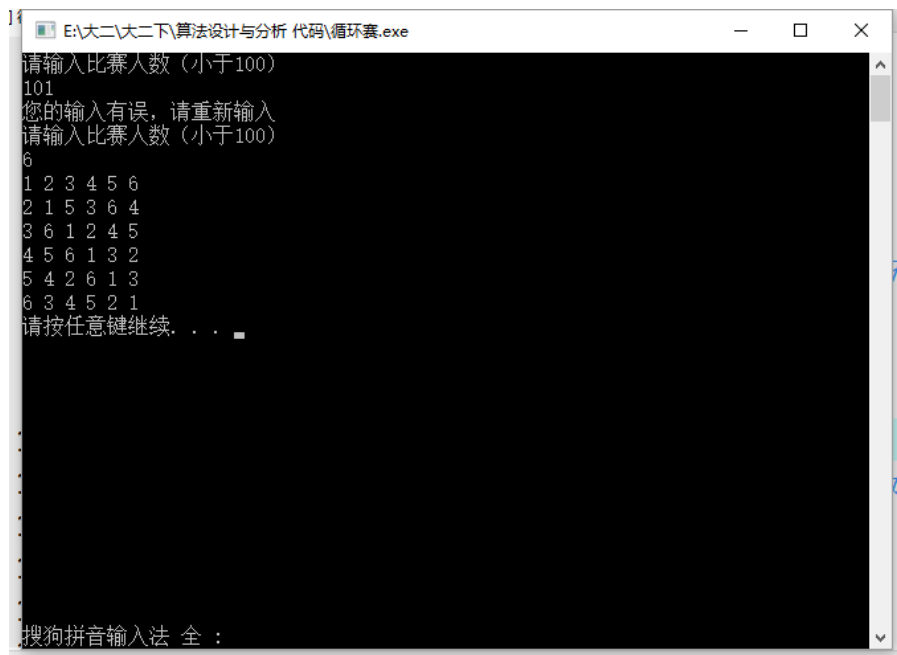


The screenshot shows a console window titled "E:\大二\大二下\算法设计与分析 代码\循环赛.exe". The prompt "请输入比赛人数 (小于100)" is displayed. The user has entered "5", and the program has output a 5x5 matrix of numbers and "请按任意键继续. . .".

```
E:\大二\大二下\算法设计与分析 代码\循环赛.exe
请输入比赛人数 (小于100)
5
1 2 3 4 5 0
2 1 5 3 0 4
3 0 1 2 4 5
4 5 0 1 3 2
5 4 2 0 1 3
请按任意键继续. . .
```

3. 第一次输入 $n=101$ ，超出系统给定的范围，所以提示输入有误，第二次输入 $n=6$ 时，

为了检测 n 为偶数时的结果。



```
E:\大二\大二下\算法设计与分析 代码\循环赛.exe
请输入比赛人数 (小于100)
101
您的输入有误, 请重新输入
请输入比赛人数 (小于100)
6
1 2 3 4 5 6
2 1 5 3 6 4
3 6 1 2 4 5
4 5 6 1 3 2
5 4 2 6 1 3
6 3 4 5 2 1
请按任意键继续. . .
```

4. 第一次输入 n 为字母, 会提示输入错误, 重新输入。第二次输入 $n=3$, 输出正确结果。



```
E:\大二\大二下\算法设计与分析 代码\循环赛.exe
请输入比赛人数 (小于100)
a
您的输入有误, 请重新输入
请输入比赛人数 (小于100)
3
1 2 3 0
2 1 0 3
3 0 1 2
请按任意键继续. . .
```

● 实验总结：

循环赛的关键问题在于巧妙的填充表格, 来安排比赛时间。填充表格用了分治法。和课上讲的区别是有 n 为奇数的情况。分治法的思想也可用于奇数中, 只需要添加一个虚拟对手。表格填充需要技巧, 偶数的比较容易理解, 奇数的主要参考网上代码, 我也花了较长时间去理解这部分代码, 对这部分的了解还是有些欠缺。需要仔细考虑多种情况, 想全奇偶之前的区别, 而不是单考虑少数情况。