

太戈编程
etiger.vip

信奥算法

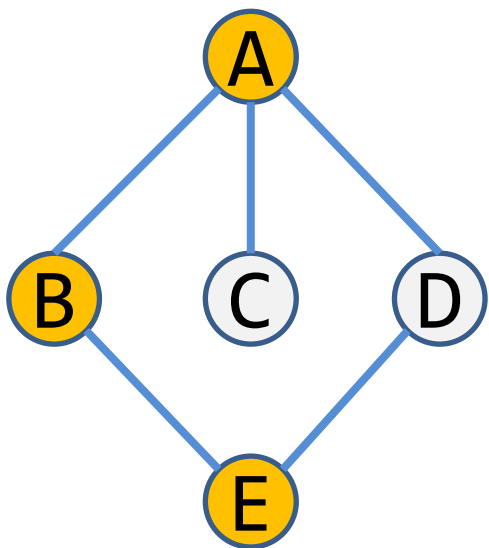
状压DP

状态压缩

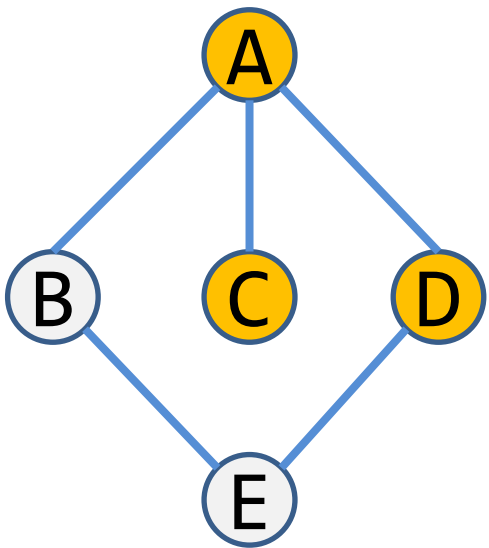
动态规划

状态压缩

用二进制数表示状态



ABCDE	
$(11001)_2$	$= (25)_{10}$



ABCDE	
$(10110)_2$	$= (22)_{10}$

状态压缩

用二进制数表示状态

选中	选中	不选	选中	不选
----	----	----	----	----

$(11010)_2$

选中	不选	选中	不选	选中
----	----	----	----	----

$(10101)_2$

不选	选中	不选	选中	选中
----	----	----	----	----

$(01011)_2$

状态压缩

用二进制数表示状态

空地	禁区	空地	禁区	空地
----	----	----	----	----

禁区	禁区	空地	空地	空地
----	----	----	----	----

禁区	空地	空地	禁区	禁区
----	----	----	----	----

$(01010)_2$	禁区
$(10101)_2$	空地

$(11000)_2$	禁区
$(00111)_2$	空地

$(10011)_2$	禁区
$(01100)_2$	空地

状态压缩

用二进制数表示状态

选中	不选	不选	不选	选中
----	----	----	----	----

空地	禁区	空地	禁区	空地
----	----	----	----	----

$$p = (10001)_2$$

$q = (01010)_2$	禁区
-----------------	----

$r = (10101)_2$	空地
-----------------	----

限制	选中的格子 必须是空地 不可以是禁区
----	--------------------------

$$(p \& q) == 0$$

$$(r | p) == r$$

状态压缩

用二进制数表示状态

选中	选中	不选	不选	选中
----	----	----	----	----

p	$(11001)_2$
---	-------------

不选	选中	选中	不选	不选
----	----	----	----	----

$p \gg 1$	$(01100)_2$
-----------	-------------

限制	选中的格子不可以相邻
----	------------

$$(p \& (p \gg 1)) == 0$$

现场挑战

1769

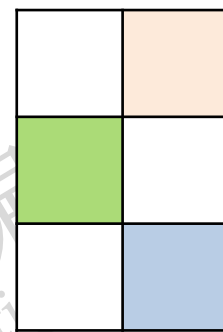
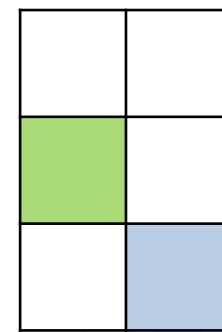
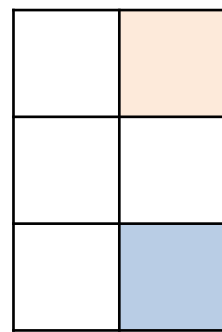
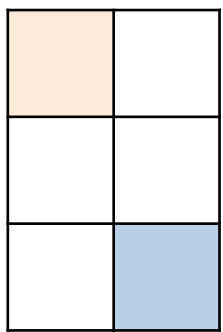
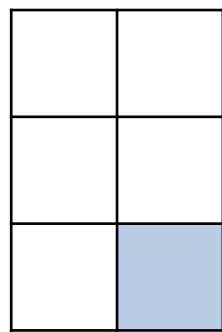
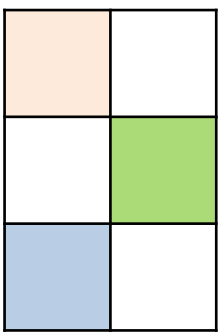
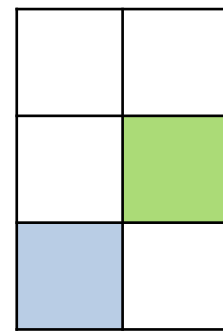
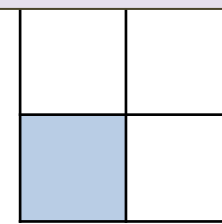
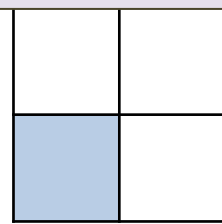
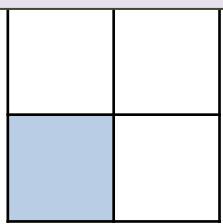
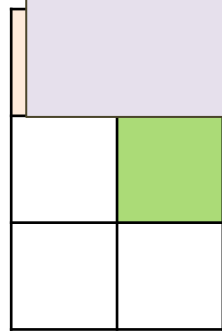
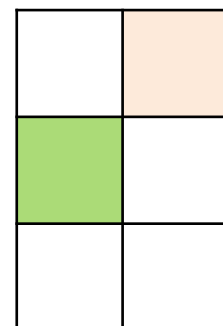
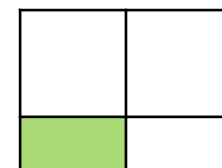
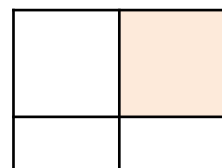
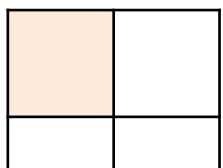
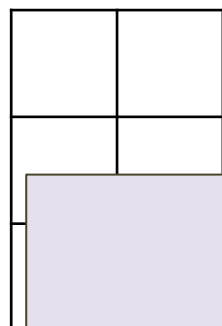
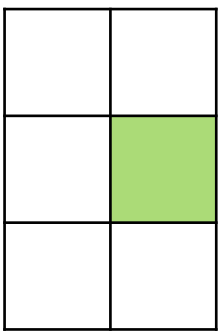
输入

3 2

输出

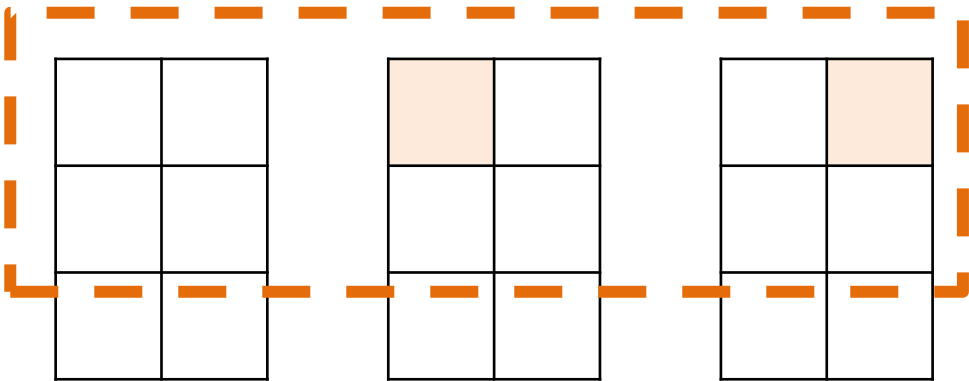
17

重大发现 重叠子问题

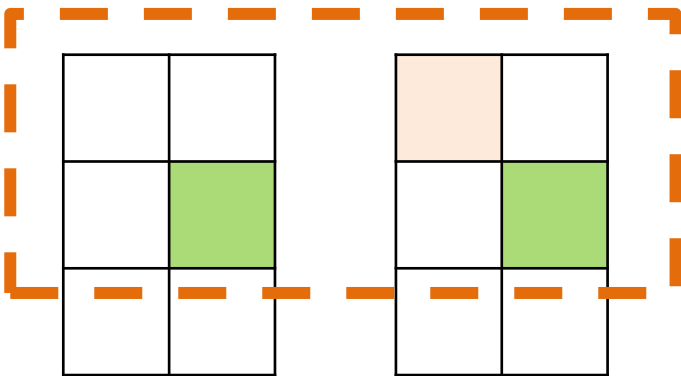


定义
状态

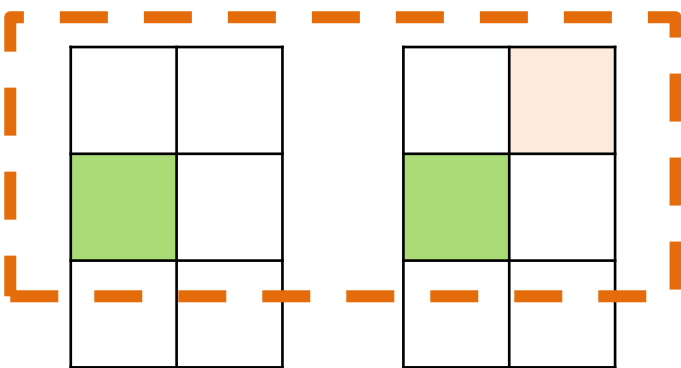
$f[i][p]$ 第 i 行选中模式为 p 时前 i 行的方案数



$$f[2][0] = f[2][(00)_2] = 3$$



$$f[2][1] = f[2][(01)_2] = 2$$



$$f[2][2] = f[2][(10)_2] = 2$$

定义
状态


$f[i][p]$ 第 i 行选中模式为 p 时前 i 行的方案数

每行的可能模式数量与列数 m 有关


$nPtn = (1 \ll m)$	2 的 m 次方种可能模式
pattern 简写 Ptn	$0, 1, 2, \dots, 2^m - 1$

已知输入 $n \leq 12, m \leq 12$
 $f[][]$ 数组定义多大?


```
int f[12][(1<<12)];
```



```
int f[13][(1<<12)];
```




```
int f[30][(1<<30)];
```



$f[i][p]$ 第 i 行选中模式为 p 时前 i 行的方案数

$nPtn = (1 \ll m)$

```
int ans=0;
for(int p=0; p<nPtn; p++)
    ans=(ans+f[n][p])%MOD;
cout<<ans<<endl;
```



枚举第 n 行所有可能模式
 $p=0, 1, 2, \dots, 2^m-1$

初始
情况

$f[i][p]$ 第*i*行选中模式为p时前*i*行的方案数

当模式p合法 没有相邻格被选中情况	当模式p非法 有相邻格被选中情况
$f[1][p]=1$	$f[1][p]=0$

二进制p	000	001	010	011	100	101	110	111
十进制p	0	1	2	3	4	5	6	7
$f[1][p]$	1	1	1	0	1	1	0	0

初始
情况

$f[i][p]$ 第 i 行选中模式为 p 时前 i 行的方案数

当模式 p 合法 没有相邻格被选中情况	当模式 p 非法 有相邻格被选中情况
$f[1][p]=1$	$f[1][p]=0$

```
18 int OK(int p){
19     return (p & (p >> 1)) == 0;
20 }

24 cin >> n >> m;
25 int nPtn = (1 << m);
26 for(int p = 0; p < nPtn; p++)
27     f[1][p] = OK(p);
```

手算表格

输入
3 2

输出
17

请在程序开头完成
状态定义+手算表格
完成后给老师检查

```
1 /*
2 f[i][p]: 第i行选中模式为p时前i行的方案数
3 输入n=3,m=2, 输出17
4 二进制p=000,001,010,011
5 十进制p= 0 , 1 , 2 , 3
6     i=1    1    1    1    0
7     i=2    3    2    2    0
8     i=3    7    5    5    0
9 */
```

$f[i][p]$ 第*i*行选中模式为p时前*i*行的方案数

二进制p	000	001	010	011
十进制p	0	1	2	3
$f[0][p]$	-	-	-	-
$f[1][p]$	1	1	1	0
$f[2][p]$	3	2	2	0
$f[3][p]$	7	5	5	0

$f[i][p]$ 依赖哪些格?

$f[i][p]$ 依赖某些 $f[i-1][q]$

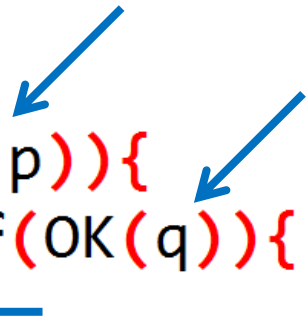
限制

p,q 选中的格子
不可以出现相邻格都选

$f[i][p]$ 第 i 行选中模式为 p 时前 i 行的方案数

$$f[i][p] = \text{sum_q} \{ f[i][q] \mid p, q \text{ 合法} \}$$

```
25 int nPtn=(1<<m);
26 for(int p=0;p<nPtn;p++)
27     f[1][p]=OK(p);
28 for(int i=2;i<=n;i++)
29     for(int p=0;p<nPtn;p++)if(OK(p)){
30         for(int q=0;q<nPtn;q++)if(OK(q)){
31             if(p&q)continue;
32             f[i][p]=(f[i][p]+f[i-1][q])%MOD;
33         }
34     }
```



错在哪里？

```

30 cin>>n>>m;
31 int nPtn=(1<<m);
32 for(int p=0;p<nPtn;p++)
33     f[1][p]=OK(p);
34 for(int i=2;i<=n;i++)
35     for(int p=0;p<nPtn;p++){
36         for(int q=0;q<nPtn;q++){
37             f[i][p]=(f[i][p]+f[i-1][q])%MOD;
38         }
39     }

```

无效状态可以尽早剔除
预处理只保留有效状态

```
25 int nPtn=(1<<m);
26 int nOK=0;
27 for(int p=0;p<nPtn;p++)
28     if(OK(p))
29         ptn[nOK++]=p;
30 for(int u=0;u<nOK;u++){
31     int p=ptn[u];
32     f[1][p]=1;
33 }
```

nOK代表当前已找到
合法的选中模式数量

ptn[u]代表
u号合法模式

已知输入
 $n \leq 12, m \leq 12$

ptn[]数组
定义多大?

无效状态可以尽早剔除
预处理只保留有效状态

```
34 for(int i=2;i<=n;i++)
35     for(int u=0;u<nOK;u++){
36         int p=ptn[u];
37         for(int v=0;v<nOK;v++){
38             int q=
39             f[i][p]=(f[i][p]+f[i-1][q])%MOD;
40         }
41     }
42 }
```

无效状态可以尽早剔除
预处理只保留有效状态

```
43 int ans=0;
44 for(int u=0; [ ]; u++){
45     [ ]
46     ans=(ans+f[n][p])%MOD;
47 }
48 cout<<ans<<endl;
```

现场挑战

1759

请为动态规划定义状态

$f[i][p]$ 第 i 行选中模式为 p 时
前 i 行能取得的最大总分

已知输入 $n \leq 16$
 $f[][]$ 数组定义多大? 请写出定义语句

```
int f[17][(1<<16)];
```

$s(i,p)$ 函数返回值为
第*i*行选中模式为*p*时第*i*行的总得分

第1行	1	2	3	4
第2行	5	6	7	8
第3行	8	7	6	5
第4行	4	3	2	1

$s(1,3)=s(1,(0011)_2)$	为7
$s(1,7)=s(1,(0111)_2)$	为9
$s(1,15)=s(1,(1111)_2)$	为10

$s(2,3)=s(2,(0011)_2)$	为15
$s(2,7)=s(2,(0111)_2)$	为21
$s(3,4)=s(3,(0100)_2)$	为7
$s(4,10)=s(4,(1010)_2)$	为6

$s(i, p)$ 函数返回值为
第 i 行选中模式为 p 时第 i 行的总得分

```

10 int s(int i, int p){
11     int sum=0;
12     for(int j=0; j<n; j++)
13         if( p & (1<<j) )
14             sum+=x[i][n-j];
15     return sum;
16 }

```

提取出
 p 的 j 号位

数组编号 12345678

$n=8$

$p = 10111010$

二进制位编号 76543210

```
26 int nPtn=(1<<n);
27 for(int i=1;i<=n;i++)
28     for(int p=0;p<nPtn;p++) {
29         int bst=-INF;
30         for(int q=0;q<nPtn;q++) if(OK(q)){
31             bst=max(bst,f[i-1][q]);
32         }
33         f[i][p]=bst+s(i,p);
34     }
35 }
```

现场挑战

1758

如何处理有障碍的格子？

$mp[i]$ 代表第 i 行地图的空地模式

第1行	0	0	1	1
第2行	0	1	0	1
第3行	1	0	1	0
第4行	1	1	1	0

$mp[1]$	为3
$mp[2]$	为5
$mp[3]$	为10
$mp[4]$	为14

如何处理有障碍的格子?

`mp[i]`代表第*i*行地图的空地模式

```
16 cin>>n>>m;  
17 for(int i=1;i<=n;i++)  
18     for(int j=1;j<=m;j++){  
19         int x; cin>>x;  
20         mp[i]<<=1;  
21         mp[i]+=x;  
22     }
```

请为动态规划定义状态

$f[i][p]$ 第 i 行选中模式为 p 时
前 i 行的方案数

已知输入 $n, m \leq 12$
 $f[][]$ 数组定义多大? 请写出定义语句

```
int f[13][(1<<12)];
```

OK(i,p)判断在第i行
选中模式p是否合法

```
10 int OK(int i,int p){
11     return (mp[i]|p)==mp[i]&&(p&(p>>1))==0;
12 }

23 int nPtn=(1<<m);
24 for(int p=0;p<nPtn;p++)
25     f[1][p]=OK(1,p);
26 for(int i=2;i<=n;i++)
27     for(int p=0;p<nPtn;p++)if(OK(i,p)){
28         for(int q=0;q<nPtn;q++) {
29             if(p&q)continue;
30             f[i][p]=
31         }
32     }
```

作业要求

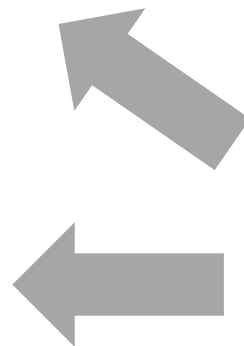
写程序前请写明：

1. 二维数组每一格的含义
2. 手算样例对应表格

查错方法：

1. 打印二维数组
2. 和手算表格对比找不同

```
1  /*
2  f[i][j]代表
3      j=0,1,2,3,4,5
4  i=0
5  i=1
6  i=2
7  i=3
8  i=4
9  i=5
10 */
```



模版格式
供参考

查错方法

```
for(int i=1;i<=n;i++,cout<<endl)
    for(int p=0;p<nPtn;p++)cout<<f[i][p]<<" ";
```

打印
表格



对照
手算
结果



定位
错误

太戈编程

1769

1759

1758

拓展题

1101,1102,1103