

第四章 阅读程序

第1节 模拟策略

1. 【NOIP2015】

```
01 #include<iostream>
02 using namespace std;
03 int main() {
04     int a,b,c;
05     a=1;b=2;c=3;
06     if (a>b) {
07         if (a>c)
08             cout<<a<<' ';
09         else
10             cout<<b<<' ';
11     }
12     cout<<c<<endl;
13     return 0;
14 }
```

●判断题

- (1)把第1行 `iostream` 改为 `cstdio`,程序仍然可以正常运行。
- (2)如果把 `a` 的初值改为 `-1`,结果不会改变。
- (3)程序结果为 3。
- (4)如果把 `c` 的初值改为 5,结果不会改变。

(✓)
(✓)
(✓)
(✗)

●选择题

- (5)把 `b` 的初值改为 4,结果为(C)。
A. 1 B. 2 C. 3 D. 4
- (6)该程序的最坏时间复杂度为(A)。
A. $O(1)$ B. $O(n)$ C. $O(n * m)$ D. $O(2^n)$

2. 【NOIP2013】

```
01 #include<iostream>
02 using namespace std;
03 int main() {
04     int a,b;
05     cin>>a>>b;
06     cout<<a<<"+"<<b<<"="<<a+b<<endl;
07 }
```

●判断题

- (1) 当 a, b 为负数且 $(a+b)$ 在 `int` 范围内, 输出的等式在数学上也成立。
- (2) 当输入为“0 0”时, 输出为“0 + 0 = 0”。
- (3) 将 03 行的 `int` 改为 `signed` 程序不会出错。
- (4) 当输出为“-1+1=0”时候, 输入可以为“2 - 2”。

●选择题

- (5) 输入为“114514 1919810”时, 输出为(**D**)。

A. $114514 + 1919810 = 2034324$

B. $114514 + 1919810 = 2034314$

C. $114514 + 1919810 = 2024324$

D. $114514 + 1919810 = 1034324$

- (6) 时间复杂度为(**B**)。

A. $O((a+b)\ln(a-b))$

B. $O(1)$

C. $O\left(\frac{a^b}{w}\right)$

D. $O(\sqrt{\log^b a})$

3. 【NOIP2014】

```
01 #include<iostream>
02 using namespace std;
03 int main() {
04     int a,b,c,d,ans;
05     cin>>a>>b>>c;
06     d=a-b;
07     a=d+c;
08     ans=a*b;
09     cout<<"Ans="<<ans<<endl;
10     return 0;
11 }
```

●判断题

- (1) `ans` 一定为 `c` 的倍数。
- (2) 输入为“8 8 8”, 输出为“64”。
- (3) 将 04 行移动到 02 03 行之间, 程序不会发生错误。
- (4) 将 09 行的 `ans` 改为 `a * b`, 输出结果不会发生改变。

●选择题

- (5) 输入为“2 3 4”时, 答案为(**C**)。

A. `Ans=9`

B. `Ans=9`

C. `Ans=9`

D. 9

- (6) 将 06 的“-”改为“+”, (5) 的输出为(**A**)。

A. `Ans=27`

B. `Ans=36`

C. `Ans=54`

D. `Ans=81`

4. 【NOIP2012】

```
01 #include<iostream>
02 using namespace std;
03 int a,b,c,d,e,ans;
04 int main() {
05     cin>>a>>b>>c;
06     d=a+b;
07     e=b+c;
08     ans=d+e;
09     cout<<ans<<endl;
10 }
```

●判断题

- (1)当输入为“11 45 14”时,输出为“70”。
- (2)输入的数都为负数时,输出可能为正数。
- (3)输入的数都为正数时,输出可能为负数。
- (4)将 ans 的类型改为 char,输出结果不会改变。

(X)
(X)
(X)
(X)

●选择题

- (5)输入为“19 19 810”时,输出为(A)。

A. 867

B. 857

C. 967

D. 767

- (6)输出为“28”时,输入可以为(C)。

A. 5 8 7

B. 1 5 7

C. 1 1 25

D. 1 1 4

5. 【NOIP2011】

```
01 #include<iostream>
02 using namespace std;
03 int main() {
04     int i,n,m,ans;
05     cin>>n>>m;
06     i=n;
07     ans=0;
08     while (i<=m) {
09         ans+=i;
10         i++;
11     }
12     cout<<ans<<endl;
13     return 0;
14 }
```

●判断题

- (1)删去第 7 行,运行结果不变。
- (2)将第 8 行的 \leq 改为 $<$,输出减小 n 。
- (3)可以实现一个复杂度为 $O(1)$ 的代码,效果与上述代码等价。
- (4)当 $m < n$ 时,程序不会运行错误。

(X)
(X)
(X)
(X)

● 选择题

(5) 输入 10 20, 输出(B)。
A. 5 B. 165

(6) 时间复杂度为(A)。
A. $O(\max\{m-n, 0\})$
C. $O(n^2)$

C. 20 D. 10

B. $O(n)$
D. $O(m \log n)$

6. 【NOIP2015】

```
01 #include<iostream>
02 using namespace std;
03 struct point {
04     int x;
05     int y;
06 };
07 int main() {
08     struct EX {
09         int a;
10         int b;
11         point c;
12     };
13     e.a=1;
14     e.b=2;
15     e.c.x=e.a+e.b;
16     e.c.y=e.a*e.b;
17     cout<<e.c.x<<','<<e.c.y<<endl;
18     return 0;
19 }
```

● 判断题

- (1) 输出结果为“3 2”。 (X)
- (2) 将 08~12 行移到 06~07 行之间, 程序运行结果不会发生改变。 (✓)
- (3) 将 01 行的 `iostream` 改为 `cstdio` 程序运行结果不会发生改变。 (✓)
- (4) 程序的时间复杂度为 $O(1)$ 。 (✓)

● 选择题

(5) 该程序会输出(B)个数。
A. 1 B. 2

C. 3

D. 0

(6) 将 13 行改为 `e.a=4`; 输出将为(A)。
A. 6, 8 B. 6, 7

C. 7, 8

D. 8, 6

7. 【NOIP2009】

```
01 #include<iostream>
02 using namespace std;
03 int main()
04 {
05     int a[3],b[3];
06     int i,j,tmp;
07     for (i=0;i<3;i++)
08         cin>>b[i];
09     for (i=0;i<3;i++)
10     {
11         a[i]=0;
12         for (j=0;j<=i;j++)
13         {
14             a[i]+=b[j];
15             b[a[i]%3]+=a[j];
16         }
17     }
18     tmp=1;
19     for (i=0;i<3;i++)
20     {
21         a[i]%=10;
22         b[i]%=10;
23         tmp*=a[i]+b[i];
24     }
25     cout<<tmp<<endl;
26     return 0;
27 }
```

●判断题

- (1)输入的3个数越大,则输出的结果也越大。
(2)执行完第17行后,b[i]比a[i]相对应的值要大。
(3)将第18行的tmp=1改为tmp=0,不论输入什么数据,输出结果都是0。
(4)若输入数据中包含字母,如3 3 a则程序会出错。

(☒)
(☒)
(☒)
(☒)

●选择题

- (5)输入2 3 5,输出的结果是()。

A. 414

B. 415

C. 416

D. 417

- (6)输入1 1 1,输出的结果是()。

A. 36

B. 48

C. 72

D. 108

8.【NOIP2009】

```
01 #include<iostream>
02 using namespace std;
03 const int c=2009;
04 int main()
05 {
06     int n,p,s,i,j,t;
07     cin>>n>>p;
08     s=0;t=1;
09     for (i=1;i<=n;i++)
10     {
11         t=t*p%c;
12         for (j=1;j<=i;j++)
13             s=(s+t)%c;
14     }
15     cout<<s<<endl;
16     return 0;
17 }
```

●判断题

(1)将 12 13 行改为 $s=(s+111*t*i)\%c$; ,程序输出不会改变。 ()

(2)将 15 行改为 `printf("%d\n",s)`; 程序输出不会改变。 (✓)

(3)将 08 行的 `s=0`; 去掉,程序输出不会改变。 (✗)

(4)将 03 行 `const` 去掉,程序输出不会发生变化。 (✓)

●选择题

(5)输入为“11 2”,输出为(A)。

A. 782

B. 762

C. 802

D. 114

(6)该算法的时间复杂度为(C)。

A. $O(1)$

B. $O(n)$

C. $O(n^2)$

D. $O(n\log n)$

9. 【NOIP2008】

```
01 #include<iostream>
02 using namespace std;
03 int main()
04 {
05     int i, a, b, c, d, f[4];
06     for (i=0; i<4; i++) cin>>f[i];
07     a=f[0]+f[1]+f[2]+f[3];
08     a=a/f[0];
09     b=f[0]+f[2]+f[3];
10     b=b/a;
11     c=(b*f[1]+a)/f[2];
12     d=f[(b/c)%4];
13     if (f[(a+b+c+d)%4]>f[2])
14         cout<<a+b<<endl;
15     else
16         cout<<c+d<<endl;
17     return 0;
```

9 19 29 39

18 }

●判断题

- (1)将 05 行移到 02 03 之间,程序不会出错。
- (2)将 13 行的“>”改为“>=”,输出不会发生改变。
- (3)将 10 行改为 $b/=a$;输出不会发生改变。
- (4)输出只会有一行。

(✓)
(✗)
(✓)
(✗)

●选择题

- (5)当输入为“9 19 29 39”,输出为(C)。

A. 21

B. 22

C. 23

D. 24

- (6)时间复杂度为(A)。

A. $O(1)$

B. $O(a)$

C. $O(\log a)$

D. $O(a^{\log a})$

10. 【NOIP2014】

```

01 #include<iostream>
02 using namespace std;
03 const int SIZE=100;
04 int alive[SIZE];
05 int n;
06 int next (int num) {
07     do {
08         num++;
09         if (num>n) num=1;
10     } while (alive[num]==0);
11     return num;
12 }
13 int main() {
14     int m,i,j,num;
15     cin>>n>>m;
16     for (i=1;i<=n;i++)
17         alive[i]=1;
18     num=1;
19     for (i=1;i<=n;i++) {
20         for (j=1;j<m;j++)
21             num=next (num);
22         cout<<num<<" ";
23         alive[num]=0;
24         if (i<n) num=next (num);
25     }
26     cout<<endl;
27     return 0;
28 }

```

● 判断题

- (1) 若输入 100 0, 该程序会运行错误。 ()
- (2) 若把 17 行去掉, 程序结果不会发生改变。 (X)
- (3) 如果把 24 行的“<”改为“<=”, 则程序结果发生改变。 ()
- (4) 若输入 11 3, 则输出 3 6 9 1 5 10 4 11 8 2 7。 ()

● 选择题

- (5) 该程序的最坏时间复杂度为 (C)。
- A. $O(1)$ B. $O(n)$ C. $(n * m)$ D. (2^n)
- (6) 若输入为 10 4, 则第 4 个数为 ()。
- A. 5 B. 6 C. 7 D. 8

【答案】

11. 【NOIP2017】

```
01 #include<iostream>
02 using namespace std;
03 int main() {
04     int n,i,j,x,y,nx,ny;
05     int a[40][40];
06     for (i=0;i<40;i++)
07         for (j=0;j<40;j++)
08             a[i][j]=0;
09     cin>>n;
10     y=0;x=n-1;
11     n=2*n-1;
12     for (i=1;i<=n*n;i++) {
13         a[y][x]=i;
14         ny=(y-1+n)%n;
15         nx=(x+1)%n;
16         if ((y==0&&x==n-1)||a[ny][nx]!=0)
17             y=y+1;
18         else {y=ny;x=nx;}
19     }
20     for (j=0;j<n;j++)
21         cout<<a[0][j]<<" ";
22     cout<<endl;
23     return 0;
24 }
```

●判断题

(1) 11行改为 $n=(n<<1)-1$; 程序运行结果不变。

(2) 输出的数字有 n (n 是最初输入的数值) 个。

(3) 将 12 行的 $i++$ 改为 $++i$ 程序运行结果不变。

(4) 将 06~08 行去掉, 程序运行结果不变。

●选择题

(5) 输入为 3 时, 输出为()。

A. 17 24 1 8 15

B. 17 21 3 5 2

C. 1 5 36 7 5

D. 17 2 8 3 1

(6) 输入为 4 时, 输出为()。

A. 30 39 48 1 10 19 28

B. 24 39 48 1 10 19 28

C. 30 39 48 1 5 19 14

D. 30 39 24 1 10 19 28

