

# 第二十五届全国青少年信息学奥林匹克联赛初赛

CSP-J C++语言模拟试题

竞赛时间:2019 年 10 月

## 选手注意:

- 试题纸共有 10 页, 答题纸共有 2 页, 满分 100 分。请在答题纸上作答, 写在试题纸上的一律无效。
- 不得使用任何电子设备 (如计算器、手机、电子词典等) 或查阅任何书籍资料。

## 一、单项选择题 (共 20 题, 无特殊说明每题 1.5 分, 共计 30 分; 每题有且仅有一个正确选项)

1. 下列著名人物中, 不是计算机相关领域专家的是 ()

- A. 王选
- B. 屠呦呦
- C. 冯诺依曼
- D. 图灵

2. 操作系统的作用是 ()

- A. 把源程序译成目标程序
- B. 控制和管理系统资源
- C. 实现硬件之间的连接
- D. 便于进行数据管理

3. 每个不同的二进制数可以表示一位学生, 现要用二进制数来表示 1200 位学生, 至少需要二进制数的位数是 ()

- A. 10
- B. 11
- C. 12
- D. 13

4. 丁丁一回到家，第一件事情就是对电脑喊一声“开机”，该台电脑就乖乖地开机了。这种现象说明了该台电脑使用的技术是

- A. 语音识别技术
- B. 字符识别技术
- C. 网络技术
- D. 自动关机技术

5. IPv4 协议使用 32 位地址，随着其不断被分配，地址资源日趋枯竭，因此，它正逐渐被 IPv6 协议所取代，IPv6 使用 128 位地址，则 1 个 IPv6 所占的字节是

- A. 8
- B. 12
- C. 20
- D. 16

6. 字符“D”的 ASCII 码为 68，则字符“Q”的 ASCII 码是

- A. 视具体计算机而定
- B. 83
- C. 82
- D. 81

7. 十进制算术表达式： $5 \times 512 + 7 \times 64 + 4 \times 8 + 7$ ，运算结果用二进制表示是

- A. 111110100101
- B. 111111011011
- C. 111111100101
- D. 101111100111

8. 将一组数 1, 3, 4, 7, 9 依次存放入数组 a 的下标为 1, 2, 3, 4, 5 单元格中, 则  $a[a[a[2]]+1]$  表达式的值是

- A. 9
- B. 7
- C. 5
- D. 3

9. 某计算机硬盘容量是 1T, 我们将 1T 用字节来表示, 以下选项中正确的是

- A.  $1024 \times 1024 \times 1024 \times 1024$
- B.  $1000 \times 1024 \times 1024 \times 1024$
- C.  $1024 \times 1024 \times 1024$
- D.  $1000 \times 1000 \times 1000 \times 1000$

10. 下列无符号数中, 最小的数是

- A.  $(4C)_{16}$
- B.  $(114)_8$
- C.  $(75)_{10}$
- D.  $(1001100)_2$

11. 在一次校趣味比赛中, 某组 6 位选手的跳绳次数分别是 “169、187、156、201、199、166”, 通过冒泡升序处理, 数据进行两两交换的总次数为 ( ) 次。

- A. 8
- B. 6
- C. 7
- D. 5

12. 有 20 只猴子顺时针围成一圈, 编号分别为 1 至 20, 从 1 号猴子开始顺时针报数, 数字从 1 开始数下去, 1, 2, 3, 4, ……, 20, 21,

22……，一圈又一圈，当数到数字  $x$  时停止，报数字  $x$  猴子的编号是：

- A.  $1+(x-1)\%20$
- B.  $(x+1)\%20-1$
- C.  $(x-1)\%20$
- D.  $(x+1)\%20$

13. 假设  $A=\text{TRUE}$ ,  $B=\text{FALSE}$ ,  $C=\text{TRUE}$ ,  $D=\text{TRUE}$ , 逻辑运算表达式  $A\wedge B\vee C\wedge D$  的值是

- A.  $\text{TRUE}$
- B.  $\text{FALSE}$
- C. 0
- D. 1

14. 已知每个 `integer` 类型的变量需要用 2 个字节的存储空间存放，则数组 `integer a[10][2]`；需要占用的内存空间字节数是

- A. 200
- B. 80
- C. 40
- D. 100

15. 设  $a[1]=1, a[2]=2, a[3]=3, a[4]=4, a[5]=5, a[6]=6$ ，且  $i=1, j=2, k=3, m=4$ ，以下选项中变量值等于 3 的是

- A.  $a[a[k-i]+3]$
- B.  $a[m/j]$
- C.  $a[a[j+k-2]]$
- D.  $a[i*j]$

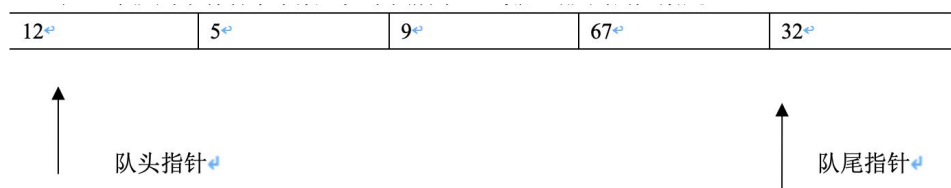
16. 进行在线游戏时，①用户登录；②进入在线游戏网；③选择游戏伙伴、开始游戏；④选择游戏室。以上 4 个环节它们正确的次序是

- A. 2 1 4 3
- B. 1 2 4 3
- C. 1 2 3 4
- D. 2 1 3 4

17. 依次将 1, 2, 3, 4, 5 放入一个栈中, 若第一个出栈元素是 3, 第二个出栈元素不可能的是

- A. 5
- B. 4
- C. 1
- D. 2

18. 已知一个队列中有若干个数 (如下图所示), 最后出队列的元素是



- A. 9
- B. 32
- C. 12
- D. 5

19. 对于序列 “7、5、1、9、3、6、8、4”, 在不改变顺序的情况下, 要使逆序对的个数减少 4, 应去掉序列中的一个数是

- A. 6

- B.7
- C.5
- D.3

20. 则城市 1 到城市 6 的最短距离是

| <div><div></div><div></div></div> | 城市 1 | 城市 2 | 城市 3 | 城市 4 | 城市 5 | 城市 6 |
|-----------------------------------|------|------|------|------|------|------|
| 城市 1                              | 0    | 2    | 3    | 1    | 12   | 15   |
| 城市 2                              | 2    | 0    | 2    | 5    | 3    | 12   |
| 城市 3                              | 3    | 2    | 0    | 3    | 6    | 5    |
| 城市 4                              | 1    | 5    | 3    | 0    | 7    | 9    |
| 城市 5                              | 12   | 3    | 6    | 7    | 0    | 2    |
| 城市 6                              | 15   | 12   | 5    | 9    | 2    | 0    |

- A.7
- B.6
- C.5
- D.8

二、阅读程序 (共 18 题，判断对错 2 分，选择题 3 分，共计 40 分；每题有且仅有一个正确选项)

```
21.
#include <bits/stdc++.h>
using namespace std;
int n, i, t, s;
int main()
{
```

```

cin >> n;
t=0;
s=0;
for (i = -n; i <= n; i++)
{
    t++;
    s = s + t;
}
cout << "t=" << t << "s=" << s;
return 0;
}

```

若输入 4，则输出 t=9s=45

- A.正确
- B.错误

22.

```

#include <bits/stdc++.h>
using namespace std;
int n, i, t, s;
int main()
{
    cin >> n;
    t=0;
    s=0;
    for (i = -n; i <= n; i++)
    {

```

```

        t++;
        s = s + t;
    }
    cout << "t=" << t << "s=" << s;
    return 0;
}

```

若输入 0，则输出 t=0s=0

A.正确

B.错误

23.

```

#include <bits/stdc++.h>
using namespace std;
int n, i, t, s;
int main()
{
    cin >> n;
    t=0;
    s=0;
    for (i = -n; i <= n; i++)
    {
        t++;
        s = s + t;
    }
    cout << "t=" << t << "s=" << s;
    return 0;
}

```



```
}
```

该程序中 for (i=-n;i<=n;i++) 可以改成 for (i=0;i<=2\*n;i++)

A.错误

B.正确

24.

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
int n, i, t, s;
```

```
int main()
```

```
{
```

```
    cin >> n;
```

```
    t=0;
```

```
    s=0;
```

```
    for (i = -n; i <= n; i++)
```

```
    {
```

```
        t++;
```

```
        s = s + t;
```

```
    }
```

```
    cout << "t=" << t << "s=" << s;
```

```
    return 0;
```

```
}
```

若输入 100000, 则输出 t=200001s=20000100000

A.正确

B.错误

25.

```
#include <bits/stdc++.h>
using namespace std;
int n, i, t, s;
int main()
{
    cin >> n;
    t=0;
    s=0;
    for (i = -n; i <= n; i++)
    {
        t++;
        s = s + t;
    }
    cout << "t=" << t << "s=" << s;
    return 0;
}
```

若把 `t++;` 放到 `s=s+t;` 之后, 则若输入 1, 输出 ( )

- A. `t=3s=6`
- B. `t=1s=1`
- C. `t=3s=3`
- D. `t=2s=3`

26.

```
#include <bits/stdc++.h>
```

```

using namespace std;
int n, i, t, s;
int main()
{
    cin >> n;
    t=0;
    s=0;
    for (i = -n; i <= n; i++)
    {
        t++;
        s = s + t;
    }
    cout << "t=" << t << "s=" << s;
    return 0;
}

```

若把 for (i = -n; i <= n; i++) 改成 for (t = -n; t <= n; t++), 则若输入 1, 输出 ( )

- A.t=1s=0
- B.t=3s=6
- C.t=1s=1
- D.t=3s=2

27.

```

#include <bits/stdc++.h>
using namespace std;
int n = 7;
int i, k, m, j, a[15];

```

```

int main()
{
    cin >> m;
    do
    {
        cin >> k;
        j = k % n;
        a[j] = 1;
        m--;
    } while (m > 0);
    k = 0;
    for (i = 1; i <= 10; i++)
        if (a[i] == 0)
            k++;
    cout << k;
    return 0;
}

```

上述代码中，把 `a[j] = 1;` 改为 `a[j]++;` 输出结果一定不变

- A. 正确
- B. 错误

28.

```

#include <bits/stdc++.h>
using namespace std;
int n = 7;
int i, k, m, j, a[15];

```

```

int main()
{
    cin >> m;
    do
    {
        cin >> k;
        j = k % n;
        a[j] = 1;
        m--;
    } while (m > 0);
    k = 0;
    for (i = 1; i <= 10; i++)
        if (a[i] == 0)
            k++;
    cout << k;
    return 0;
}

```

上述代码中，删掉 `k=0;` 输出结果一定不变

- A.正确
- B.错误

29.

```

#include <bits/stdc++.h>
using namespace std;
int n = 7;
int i, k, m, j, a[15];

```

```

int main()
{
    cin >> m;
    do
    {
        cin >> k;
        j = k % n;
        a[j] = 1;
        m--;
    } while (m > 0);
    k = 0;
    for (i = 1; i <= 10; i++)
        if (a[i] == 0)
            k++;
    cout << k;
    return 0;
}

```

若输入的 m 为 0，则 do...while... 循环不会执行

A. 错误

B. 正确

30.

```

#include <bits/stdc++.h>
using namespace std;

```

```

int n = 7;
int i, k, m, j, a[15];
int main()
{
    cin >> m;
    do
    {
        cin >> k;
        j = k % n;
        a[j] = 1;
        m--;
    } while (m > 0);
    k = 0;
    for (i = 1; i <= 10; i++)
        if (a[i] == 0)
            k++;
    cout << k;
    return 0;
}

```

若输入数据中  $k$  为负数，则程序一定可以正常运行

- A. 正确
- B. 错误

31.

```
#include <bits/stdc++.h>
```

```
using namespace std;
int n = 7;
int i, k, m, j, a[15];
int main()
{
    cin >> m;
    do
    {
        cin >> k;
        j = k % n;
        a[j] = 1;
        m--;
    } while (m > 0);
    k = 0;
    for (i = 1; i <= 10; i++)
        if (a[i] == 0)
            k++;
    cout << k;
    return 0;
}
```

若输入数据为

10

2006 2007 2008 2009 2010 2011 2012 2013 2014 2015

输出为 ( )



A.7

B.0

C.4

D.10

32.

```
#include <bits/stdc++.h>
using namespace std;
int n = 7;
int i, k, m, j, a[15];
int main()
{
    cin >> m;
    do
    {
        cin >> k;
        j = k % n;
        a[j] = 1;
        m--;
    } while (m > 0);
    k = 0;
    for (i = 1; i <= 10; i++)
        if (a[i] == 0)
            k++;
    cout << k;
    return 0;
}
```

上述代码中，在只改变声明时 a 数组大小的情况下，a 数组大小最小是（ ）

A.11

B.6

C.10

D.7

33.

```
#include <bits/stdc++.h>
using namespace std;
int n, m, k, max, s, a[101];
int main()
{
    cin >> n >> m >> k;
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= m; j++) cin >> a[j];
        if (i == k)
        {
            for (int j = 1; j <= m + 1; j++)
            {
                if (a[j] != 0) s++;
                else
                {
                    if (s > max) max = s;
                    s = 0;
                }
            }
        }
    }
}
```

```
    cout << max;
    return 0;
}
```

上述代码中是在统计每一行中连续非 0 数的数量 ( )

- A.正确
- B.错误

34.

```
#include <bits/stdc++.h>
using namespace std;
int n, m, k, max, s, a[101];
int main()
{
    cin >> n >> m >> k;
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= m; j++) cin >> a[j];
        if (i == k)
        {
            for (int j = 1; j <= m + 1; j++)
            {
                if (a[j] != 0) s++;
                else
                {
                    if (s > max) max = s;
                }
            }
        }
    }
}
```

```

        s = 0;
    }
}
}
cout << max;
return 0;
}

```

上述代码中若将 for (int j = 1; j <= m + 1; j++) 改为 for (int j = 1; j <= m; j++) 输出结果一定不变 ( )

- A. 错误
- B. 正确

35.

```

#include <bits/stdc++.h>
using namespace std;
int n, m, k, max, s, a[101];
int main()
{
    cin >> n >> m >> k;
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= m; j++) cin >> a[j];
        if (i == k)
        {
            for (int j = 1; j <= m + 1; j++)

```

```

        {
            if (a[j] != 0) s++;
            else
            {
                if (s > max) max = s;
                s = 0;
            }
        }
    }
}
cout << max;
return 0;
}

```

为保证上述代码正确运行，m 不能超过 99

- A.正确
- B.错误

36.

```

#include <bits/stdc++.h>
using namespace std;
int n, m, k, max, s, a[101];
int main()
{
    cin >> n >> m >> k;
    for (int i = 1; i <= n; i++)
    {

```

```

    for (int j = 1; j <= m; j++) cin >> a[j];
    if (i == k)
    {
        for (int j = 1; j <= m + 1; j++)
        {
            if (a[j] != 0) s++;
            else
            {
                if (s > max) max = s;
                s = 0;
            }
        }
    }
}
cout << max;
return 0;
}

```

在保证  $1 \leq k \leq n$  的情况下，可以将上述代码中的 `for (int i = 1; i <= n; i++)` 改为 `for (int i = 1; i <= k; i++)` 输出结果一定不变

- A.正确
- B.错误

37.

```

#include <bits/stdc++.h>
using namespace std;
int n, m, k, max, s, a[101];

```

```

int main()
{
    cin >> n >> m >> k;
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= m; j++) cin >> a[j];
        if (i == k)
        {
            for (int j = 1; j <= m + 1; j++)
            {
                if (a[j] != 0) s++;
                else
                {
                    if (s > max) max = s;
                    s = 0;
                }
            }
        }
    }
    cout << max;
    return 0;
}

```

若输入数据为

```

4 12 3
9 2 0 0 3 4 2 5 0 1 3 0
4 0 2 3 0 3 2 3 1 2 5 7

```



3 8 1 2 0 4 7 5 1 3 0 9

1 1 1 1 1 0 2 3 1 3 4 1

max = s; 执行了 ( ) 次

A.3

B.0

C.1

D.2

38.

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
int n, m, k, max, s, a[101];
```

```
int main()
```

```
{
```

```
    cin >> n >> m >> k;
```

```
    for (int i = 1; i <= n; i++)
```

```
    {
```

```
        for (int j = 1; j <= m; j++) cin >> a[j];
```

```
        if (i == k)
```

```
        {
```

```
            for (int j = 1; j <= m + 1; j++)
```

```
            {
```

```
                if (a[j] != 0) s++;
```

```
                else
```

```
                {
```

```
                    if (s > max) max = s;
```

```
        s = 0;
    }
}
}
}
cout << max;
return 0;
}
```

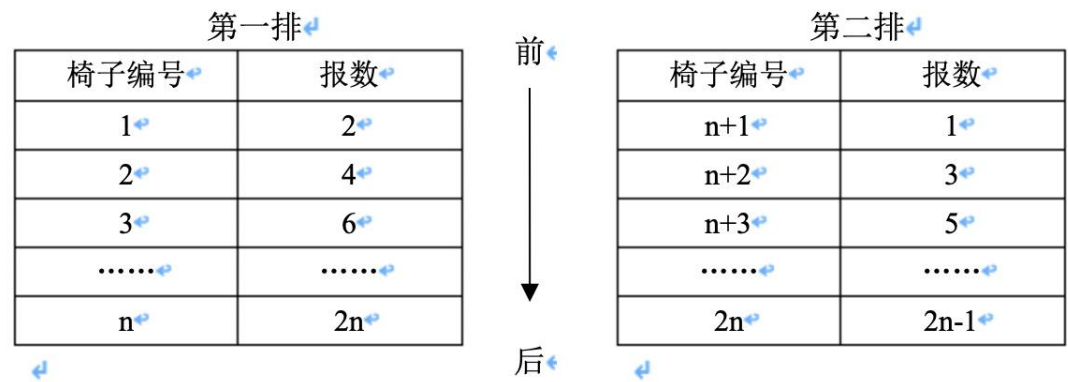
上述代码的时间复杂度为（ ）

- A.  $O(m)$
- B.  $O(n)$
- C.  $O(n+m)$
- D.  $O(nm)$

三、阅读程序(共 10 题，共计 30 分；每题有且仅有一个正确选项)

在奇思妙想班里，每天都有一些奇思妙想的东西产生，他们的想法总是让人匪夷所思，总能根据一些课堂上获得的知识，联想出一个新的问题。一天编程课上，老师与奇思妙想班的同学们讨论了“猴子选大王”问题，老师要求课后，同学们能想一个与书本上不一样的猴子选大王问题，这下可不得了了，奇思妙想班的同学们开始创想，并展开了大讨论。多多创想的“大王的学号”问题，被大家公认为不错的问题，并进行了演示游戏活动。

学号为 1 至  $2n$  的同学带好他们的椅子来到操场上，排成两排，第一排椅子从前向后标上  $1, 2, 3, \dots, n$  编号，第二排椅子从前向后标上  $n+1, n+2, n+3, \dots, 2n$  编号。游戏开始前让同学们按学号就座，这是最初的顺序。第一排学生按  $2, 4, 6, 8, \dots, 2n$  报数，第二排学生按  $1, 3, 5, 7, \dots, 2n-1$  报数，如下图表所示。然后每位同学按所报的数字坐到对应编号的椅子上。经过  $m$  次后，每位同学都会回到最初的顺序，游戏结束。最后  $(m \bmod (2 \times n)) + 1$  学号的同学就是要选的“大王”。



输入说明，一个数  $n$

输出说明，一个数，“大王”的学号

【样例输入】

1

8

**【样例输出】**

1

9

**【程序清单】**

```
#include <bits/stdc++.h>
using namespace std;
int n, m, a[101], b[101];
bool ok;
int main()
{
    ①;
    for (int i = 1; i <= 2 * n; i++) ②;
    do
    {
        for (int i = 1; i <= 2 * n; i++)
            if (i % 2 == 0) b[i] = a[i / 2];
        else ③;
        for (int i = 1; i <= 2 * n; i++) a[i] = b[i];
        m++;
        ok = true;
        for (int i = 1; i <= 2 * n; i++)
            if (a[i] != i)
            {
```

```

        ④;
        break;
    }
} while ( ⑤ );
cout << m % (2 * n) + 1 << " ";
return 0;
}

```

39. 请选择 ① 应该填写的代码。

- A. scanf("%d", n)
- B. cin>>m
- C. cin>>n
- D. cout<<n

40. 请选择 ② 应该填写的代码。

- A. a[i]=i/2
- B. a[i]=i
- C. a[i]=i%n
- D. cin>>a[i]

41. 请选择 ③ 应该填写的代码。

- A. b[i]=a[i/2+1]
- B. b[i]=a[i%n+1]
- C. b[i]=a[(i+1)/2+n]
- D. b[i]=a[i]

42. 请选择 ④ 应该填写的代码。

- A.ok=true
- B.m++
- C.m--
- D.ok=false

43. 请选择 ⑤ 应该填写的代码。

- A.!ok
- B.m>=0
- C.m>0
- D.ok

多多对数字特别感兴趣，在一节数学课中老师介绍了“完全数”，多多被“完全数”吸引住了，他认为“完全数”是最美的数字，所谓完全数是指约数之和等于它本身的数。例如 28 的约数是 1,2,4,7,14，并且  $1+2+4+7+14=28$ ，所以 28 是完全数。多多想找出一段连续数据里面的所有完全数，如求  $x\sim y$  中的完全数。但是他又感觉十进制的输出没有十六进制好看，所以他想把找到的完全数用十六进制输出，如果这段数据里面没有完全数就输出 no。

输入说明，一行，两个数  $x,y$ ，中间用空格分开。

输出说明，一行， $x\sim y$  之间用十六进制表示的完全数，数之前用空格分开。

**【样例输入】**

1  
2 29

**【样例输出】**

1

6 1C

【程序清单】

```
#include <bits/stdc++.h>
using namespace std;
int x, y, e;
bool wqs(int a)
{
    int ks = 1;
    for (int i = 2; i <= trunc(sqrt(a)); i++)
        if ( ⑥ )
        {
            if (i * i == a) ks += i;
            else ⑦ ;
        }
    if (ks == a) return true;
    else return false;
}
string change(int a)
{
    string s1 = "0123456789ABCDEF";
    string s = "";
    while (a > 0)
    {
        ⑧ ;
        a /= 16;
    }
}
```

```

    }
    return s;
}
int main()
{
    cin >> x >> y;
    e = 0;
    for (int i = x; i <= y; i++)
        if( ⑨ )
        {
            cout << change(i) << " ";
            e++;
        }
    ⑩ ;
    return 0;
}

```

44.请选择 ⑥ 应该填写的代码

- A.  $a \% i == 0$
- B.  $a / i > 0$
- C.  $a \% i != 0$
- D.  $a / i == 0$

45.请选择 ⑦ 应该填写的代码。

- A.  $ks += a / i$
- B.  $ks += i + a / i$



- C.ks+=a%i
- D.ks+=i+a%i

46. 请选择 ⑧ 应该填写的代码。

- A.s+=s1[a]
- B.s+=s1[a%16]
- C.s+=s1[a/16]
- D.s+=s1[a%16+1]

47. 请选择 ⑨ 应该填写的代码。

- A.!wqs(i)
- B.wqs(change(i))
- C.wqs(i)
- D.!wqs(change(i))

48. 请选择 ⑩ 应该填写的代码。

- A.cout<<e
- B.if(e) cout<<"no"
- C.if(e==0) cout<<"no"
- D.cout<<"no"

## 第二十五届全国青少年信息学奥林匹克联赛初赛

### CSP-J模拟卷参考答案

#### 一、单项选择题（共20题，每题1.5分，共计30分）

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  |
| B  | B  | B  | A  | D  | D  | D  | A  |
| 9  | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| A  | C  | C  | A  | A  | C  | C  | A  |
| 17 | 18 | 19 | 20 |    |    |    |    |
| C  | B  | C  | A  |    |    |    |    |

#### 二、阅读程序写结果（共4题，每题10分，共计40分）

|     |    |    |    |    |    |    |
|-----|----|----|----|----|----|----|
| 第一题 | 21 | 22 | 23 | 24 | 25 | 26 |
|     | A  | B  | B  | B  | C  | D  |
| 第二题 | 27 | 28 | 29 | 30 | 31 | 32 |
|     | A  | B  | A  | B  | C  | A  |
| 第三题 | 33 | 34 | 35 | 36 | 37 | 38 |
|     | B  | A  | A  | A  | D  | D  |

三、完善程序（每空3 分，共计30 分）

|     |    |    |    |    |    |
|-----|----|----|----|----|----|
| 第一题 | 39 | 40 | 41 | 42 | 43 |
|     | C  | B  | C  | D  | A  |
| 第二题 | 44 | 45 | 46 | 47 | 48 |
|     | A  | B  | B  | C  | C  |