

# 一、单项选择题（共15题，每题2分，共计30分；每题有且仅有一个正确选项）

---

1. 一个 64 位浮点型变量占用（ ）个字节。
  - A. 4
  - B. 8
  - C. 16
  - D. 128
2. 设有 1000 个已排好序的数据元素，采用折半查找时，最大比较次数为（ ）。
  - A. 6
  - B. 8
  - C. 10
  - D. 12
3. 现有两张分辨率为  $2048 * 2048$  像素的 128 位真彩色图像。要存储这两张图像，需要多大的存储空间？（ ）。
  - A. 32MB
  - B. 64MB
  - C. 128MB
  - D. 256MB
4. 二进制数 1011.11 转成十进制数是（ ）。
  - A. 11.55
  - B. 11.75
  - C. 13.55
  - D. 13.75
5. 设简单无向图  $G$  有 20 条边且每个顶点的度数都是 4，则图  $G$  有（ ）个顶点。
  - A. 5
  - B. 10
  - C. 20
  - D. 40
6. 周末徐老师和爸爸妈妈三个人一起想动手做三道菜。徐老师负责洗菜、爸爸负责切菜、妈妈负责炒菜。假设做每道菜的顺序都是：先洗菜 15 分钟，然后切菜 10 分钟，最后炒菜 20 分钟。那么做一道菜需要 45 分钟。

注意：两道不同的菜的相同步骤不可以同时进行。例如第一道菜和第二道的菜不能同时洗，也不能同时切。

那么做完三道菜的最短时间需要 ( ) 分钟。

- A. 85
- B. 90
- C. 120
- D. 135

7. 表达式  $(a + b) * (c + d) + e$  的后缀形式是 ( ) 。

- A.  $a + b c + d * + e$
- B.  $a b c d + + * + e$
- C.  $a b + c d + * e +$
- D.  $a b + c d + e + *$

8. 字符串 daniu 的子串个数是 ( ) 。

- A. 10
- B. 15
- C. 16
- D. 20

9. 对于入栈顺序为  $a, b, c, d, e, f, g$  的序列, 下列 ( ) 可能是其合法的出栈序列。

- A.  $a, b, c, g, f, d, e$
- B.  $c, b, a, f, d, e, g$
- C.  $c, b, f, g, e, d, a$
- D.  $g, f, e, d, a, b, c$

10. 设  $x = true, y = false, z = true$ , 以下逻辑表达式值为真的是 ( ) 。

- A.  $(x \wedge y) \vee (y \wedge z)$
- B.  $x \wedge y \wedge (y \vee z)$
- C.  $(x \vee y) \wedge (y \wedge z)$
- D.  $(x \vee y) \wedge (y \vee z)$

11. 一棵具有 8 层的满二叉树中结点数为 ( ) 。

- A. 128
- B. 255
- C. 256
- D. 511

12. 一个字长为 8 位的整数的补码是 10101010, 则它的原码是 ( ) 。

- A. 11011010
- B. 11010101
- C. 11010110
- D. 10110011

13. 一棵二叉树的先序遍历序列是 ABCDEFG, 中序遍历序列是 DCBEAFG, 则这个二叉树的后序遍历序列为 ( ) 。

- A. CDFEGBA
- B. CFDBEAG

C. DCEBGFA

D. CFBDEGA

14. 设哈希表的地址空间为 0 到 10,散列函数为  $hash(n) = n \bmod 11$ ,用线性探查法解决碰撞。现从空的哈希表开始,依次插入关键码值 84, 25, 38, 57, 71,则最后一个关键码 71 的地址为 ( )。

A. 7

B. 6

C. 5

D. 4

15. 3 名医生和 6 名护士被分配到 3 所学校为学生体检,每校分配 1 名医生和 2 名护士,不同的分配方法共有 ( ) 种。

A. 360

B. 540

C. 720

D. 960

---

## 二、阅读程序 (程序输入不超过数组或字符串定义的范围; 判断题正确填√, 错误填×; 除特殊说明外, 判断题 1.5 分, 选择题 3 分, 共计 40 分)

---

### 第一小题

---

```
1     #include <stdio>
2     #include <string>
3     using namespace std;
4     char s[101];
5     int n, cnt[26];
6     int main() {
7         scanf("%s", s);
8         n = strlen(s);
9         for (int i = 0; i < n; i++) {
10            if (s[i] >= 'A' && s[i] <= 'Z')
```

```

11         cnt[s[i]-'A']++;
12         if (s[i] >= 'a' && s[i] <= 'z')
13             cnt[s[i]-'a']++;
14         if (s[i] >= '0' && s[i] <= '9')
15             cnt[s[i]-'0']++;
16     }
17     int p = 0;
18     for (int i = 1; i < 26; i++)
19         if (cnt[i] > cnt[p])
20             p = i;
21     printf("%d\n", p);
22     return 0;
23 }

```

假设输入的字符串长度不超过 100，完成下面的判断题和单选题：

16. 输入的字符串只能由小写字母或大写字母组成。 ( )
17. 将第 9 行的  $i < n$  改成  $i \leq n$ ，程序运行时可能会发生错误。 ( )
18. 将第 12 行的  $s[i] \geq 'a' \ \&\& \ s[i] \leq 'z'$  改成  $s[i] \geq 'a'$ ，程序运行时可能会发生错误。 ( )
19. 若输入的字符串全部由数字字符组成，则输出的整数必然小于 10。 ( )
20. 若输入为 ABCDcbaAcDbC，输出为 ( )。
- A. 0
- B. 1
- C. 2
- D. 3
21. 若输入为 a2B3233CCDC，输出为 ( )。
- A. 0
- B. 1
- C. 2
- D. 3

## 第二小题

```

1     #include <iostream>
2     using namespace std;
3     int f(int n, int m) {
4         if (n == 1) return m;
5         if (m == 1) return n;

```

```

6         int res = 0;
7         for (int i = 1; i < n; i++)
8             for (int j = 1; j < m; j++)
9                 res += f(i, j);
10        return res;
11    }
12    int n, m;
13    int main() {
14        cin >> n >> m;
15        cout << f(n, m) << endl;
16        return 0;
17    }

```

22. 计算  $f(n, m)$  的时间复杂度为  $O(nm)$ 。 ( )
23. 去掉第 4 行的代码, 程序将会不停地递归调用而不会返回结果。 ( )
24. 同时去掉第 4 行和第 5 行的代码, 程序将会不停地递归调用而不会返回结果。 ( )
25. 当输入为 5 6 时, 输出为 ( )。
- A. 92
- B. 143
- C. 166
- D. 175
26. 当输入为 100 2 时, 输出为 ( )。
- A. 3762
- B. 4950
- C. 5050
- D. 6689
27. 当去掉程序中的第 4 行, 且输入为 3 6 时, 输出为 ( )。
- A. 0
- B. 7
- C. 25
- D. 43

### 第三小题

```

1     #include <cstring>
2     using namespace std;
3     char s[1000];
4     int f[1000][8], n, x, y;

```

```

5     int Log2(int x) {
6         int a = 0, b = 1;
7         while (b*2 <= x) {
8             a ++;
9             b *= 2;
10        }
11        return a;
12    }
13    int main() {
14        cin >> s >> x >> y;
15        n = strlen(s);
16        for (int i = 0; i < n; i++)
17            f[i][0] = s[i];
18        for (int i = 1; (1<<i) <= n; i++)
19            for (int j = 0; j+(1<<i)-1 < n; j++)
20                f[j][i] = max(f[j][i-1], f[j+(1<<i-1)][i-1]);
21        int z = Log2(y-x+1);
22        cout << (char)max(f[x][z], f[y-(1<<z)+1][z]) << endl;
23        return 0;
24    }

```

28. 当输入为 CGFCDBAE 2 6 时, 输出为 G。 ( )
29. 当输入的第一个字符串的长度为 500 时, 会发生数组越界。 ( )
30. 交换第 19 行和第 20 行的代码并不会影响程序的输出结果。 ( )
31.  $\text{Log}_2(15)$  的返回值为 ( )。
- A. 2
  - B. 3
  - C. 4
  - D. 5
32. 当输入为 HETAOACCEPT 0 9 时, 输出结果为 ( )。
- A. *H*
  - B. *E*
  - C. *P*
  - D. *T*
33. (4分) 当输入为 CodeInHETAO 4 10 时, 输出结果为 ( )。
- A. *I*
  - B. *n*
  - C. *T*
  - D. *O*
-

## 三、完善程序（单选题，每小题3分，共计30分）

### 第一小题

（排序问题）大牛老师发明了一个排序算法，用于将  $n$  个整数从小到大排序后输出。下面是他编写的排序代码，但是有一部分丢失了。

试补全下方的排序代码，使其能够将  $n$  个整数从小到大排序后输出。

```
1     #include <iostream>
2     using namespace std;
3     int main() {
4         int n, a[101], i, t;
5         cin >> n;
6         for (i = 0; i < n; i++)
7             cin >> a[i];
8         ___ (1) ___ = 1;
9         while ( ___ (2) ___ ) {
10            if (i==0 || ___ (3) ___ ) {
11                i++;
12            }
13            else {
14                t = a[i];
15                a[i] = ___ (4) ___ ;
16                a[i-1] = t;
17                ___ (5) ___ ;
18            }
19        }
20        for (i = 0; i < n; i++)
21            cout << a[i] << endl;
22        return 0;
23    }
```

34. \_\_\_ (1) \_\_\_ 处应填 ( )

- A.  $i$
- B.  $n$
- C.  $t$
- D.  $a[0]$

35. \_\_\_ (2) \_\_\_ 处应填 ( )
- A.  $i > 0$
  - B.  $i > 1$
  - C.  $i < n$
  - D.  $i \leq n$
36. \_\_\_ (3) \_\_\_ 处应填 ( )
- A.  $i == 1$
  - B.  $i == n$
  - C.  $a[i - 1] \leq a[i]$
  - D.  $a[i - 1] \geq a[i]$
37. \_\_\_ (4) \_\_\_ 处应填 ( )
- A.  $a[0]$
  - B.  $a[i - 1]$
  - C.  $a[i + 1]$
  - D.  $a[n - 1]$
38. \_\_\_ (5) \_\_\_ 处填写 ( ) 可以使程序运行地最快。
- A.  $i = 0$
  - B.  $i = 1$
  - C.  $i - -$
  - D.  $i + +$

---

## 第二小题

---

(上一个排列问题) 给定一个由  $1 \sim n$  构成的排列  $a$ , 求  $a$  的上一个排列。  
比如, 当  $n$  等于 3 时, 由  $1 \sim 3$  构成的全排列按照从小到大排如下所示:

1, 2, 3  
1, 3, 2  
2, 1, 3  
2, 3, 1  
3, 1, 2  
3, 2, 1

其中, 排列 3, 2, 1 的上一个排列为 3, 1, 2; 排列 3, 1, 2 的上一个排列为 2, 3, 1; 排列 2, 3, 1 的上一个排列为 2, 1, 3; 排列 2, 1, 3 的上一个排列为 1, 3, 2; 排列 1, 3, 2 的上一个排列为 1, 2, 3; 排列 1, 2, 3 没有上一个排列。

你的任务是根据输入的排列计算并输出其上一个排列;

特别地，若输入的排列没有上一个排列，输出 -1。  
试补全下面的模拟上一个排列的程序。

```
1     #include <iostream>
2     #include <algorithm>
3     using namespace std;
4     int n, a[1000];
5     bool cmp(int a, int b) {
6         return ___(1)___ ;
7     }
8     bool pre_permutation(int a[], int n) {
9         int i = n-1;
10        while (i > 0 && a[i-1] < a[i])
11            i--;
12        if ( ___(2)___ )
13            return false;
14        for (int j = i; j < n; j++) {
15            if (j == n-1 || ___(3)___ ) {
16                swap( ___(4)___ , a[j]);
17                break;
18            }
19        }
20        sort( ___(5)___ , a+n, cmp);
21        return true;
22    }
23    int main() {
24        cin >> n;
25        for (int i = 0; i < n; i++)
26            cin >> a[i];
27        if (pre_permutation(a, n)) {
28            for (int i = 0; i < n; i++)
29                cout << a[i] << " ";
30        }
31        else {
32            cout << -1 << endl;
33        }
34        return 0;
35    }
```

39. \_\_\_ (1) \_\_\_ 处应填 ( )

- A.  $a == b$
- B.  $a > b$
- C.  $a < b$
- D.  $a + b$

40. \_\_\_ (2) \_\_\_ 处应填 ( )

A.  $!i$

B.  $i = 1$

C.  $i == 1$

D.  $i > 1$

41. \_\_\_ (3) \_\_\_ 处应填 ( )

A.  $a[j] == a[i]$

B.  $a[j] < a[j + 1]$

C.  $a[j + 1] > a[i - 1]$

D.  $a[j + 1] > a[i]$

42. \_\_\_ (4) \_\_\_ 处应填 ( )

A.  $a[0]$

B.  $a[i - 1]$

C.  $a[i]$

D.  $a[j + 1]$

43. \_\_\_ (5) \_\_\_ 处应填 ( )

A.  $a[i]$

B.  $a + i - 1$

C.  $a + i$

D.  $a + j$