

## 2024 LUOGU 非专业级别收容能力认证第一轮

### (SCP-J1) 入门级 C++语言试题

认证时间：2024 年 8 月 11 日 09:30~11:30

#### 考生注意事项：

- 试题纸共有 10 页，满分 100 分。请在洛谷作答，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。
- 试题由洛谷网校学术组命制，欢迎报名洛谷网校第一轮课程。课程内容包含专题讲解、真题讲评与本试题讲评。<https://class.luogu.com.cn/course/yugu24acs>

#### 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. C++是一种面向对象的程序设计语言。在 C++中，下面哪个关键字用于声明一个类，其缺省继承方式为 private 继承？（ ）

- A. union
- B. struct
- C. class
- D. enum

2. 下述代码实现的数据结构是（ ）。

```
int data[100], f = 1, r;  
void insert(int value) {  
    data[++r] = value;  
}  
void pop() {  
    f++;  
}
```

- A. 链表
- B. 栈
- C. 队列
- D. 平衡树

3. C++语言中，以 0b 开头的数为（ ）进制数。

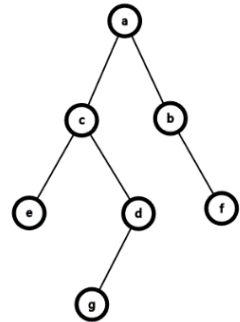
- A. 二进制
- B. 八进制
- C. 十进制
- D. 十六进制

4. 根结点的高度为 1，高度为 5 的完全二叉树至少有 ( ) 个结点。

- A. 15
- B. 16
- C. 31
- D. 32

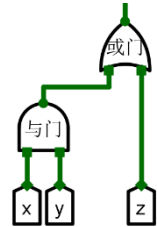
5. 右图所示的二叉树，其后序遍历的结果是什么? ( )

- A. acedgbf
- B. fbacdge
- C. edgcabf
- D. egdcfba



6. 考虑右图所示的数字电路，有关逻辑门的含义已在图中标出。高电平表示 true，低电平表示 false。当 x,y,z 的输入依次为低电平、高电平、高电平时，输出为 ( )。

- A. 高电平
- B. 低电平
- C. 电路故障
- D. 高阻



7. 十进制数 10.375 转换为八进制数的结果为 ( )。

- A. 10.5
- B. 10.3
- C. 12.5
- D. 12.3

8. 假设有一组字符 {g,h,i,j,k,l}，它们对应的频率分别为 8%,14%,17%,20%,23%,18%。请问以下哪个选项是字符 g,h,i,j,k,l 分别对应的一组哈夫曼编码? ( )

- A. g: 1100, h: 1101, i: 111, l: 10, k: 00, j: 01
- B. g: 0000, h: 001, i: 010, l: 011, k: 10, j: 11
- C. g: 111, h: 110, i: 101, l: 100, k: 01, j: 00
- D. g: 110, h: 111, i: 101, l: 100, k: 0, j: 01

9. 中缀表达式  $((6 - 3) * 2 + 7) / (5 ^ (3 * 4 + 2))$  对应的后缀表达式为 ( )。

- A. / + \* - 6 3 2 7 ^ 5 + \* 3 4 2
- B. 6 3 2 - \* 7 + 5 3 4 \* 2 + ^ /
- C. 6 3 - 2 \* 7 + 5 3 4 \* 2 + ^ /
- D. 6 3 - 2 \* 7 + 3 4 \* 2 + 5 ^ /

10. 将 3 个相同的红球和 3 个相同的黑球装入三个不同的袋中，每袋均装 2 个球，则不同的装法总数为 ( )。

- A. 7
- B. 8

C. 9

D. 10

11. 从 2 至 8 的 7 个整数中随机取 2 个不同的数，这两个数互质的概率为 ( )。

A. 1/6

B. 1/3

C. 1/2

D. 2/3

12. 以下哪一种算法典型地使用了分治法的思想来解决问题? ( )

A. 线性搜索

B. 快速排序

C. 冒泡排序

D. 插入排序

13. 奇偶校验编码是常见的校验编码方式。对于二进制编码  $A_n A_{n-1} \dots A_2 A_1$ ，奇偶校验编码在编码的最后增加一位校验位  $G$ ，并将原编码与校验位作为整体发送。校验位分为奇校验位与偶校验位，奇校验位保证  $A_n \text{ xor } A_{n-1} \text{ xor } \dots \text{ xor } A_2 \text{ xor } A_1 \text{ xor } G = 1$ ，偶校验位保证  $A_n \text{ xor } A_{n-1} \text{ xor } \dots \text{ xor } A_2 \text{ xor } A_1 \text{ xor } G = 0$ 。下列编码与校验位对应正确的是 ( )。

A. 编码 11100111 奇校验位 0

B. 编码 01100010 偶校验位 0

C. 编码 00010010 奇校验位 1

D. 编码 11100010 偶校验位 1

14. 下列关于 NOI 系列活动的有关说法，错误的是 ( )。

A. NOI 考试对 C++ 语言的使用没有限制。

B. 选手不可以携带草稿纸、手机、U 盘等进入考场。

C. 主办单位 CCF 的全称为中国计算机学会。

D. 在 CSP 第一轮考试中舞弊，可能会被给予取消考试资格、禁赛等处罚。

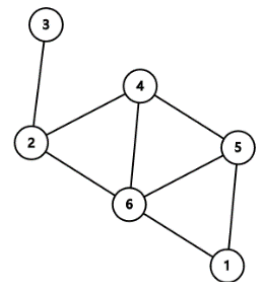
15. 考虑右图所示的无向图，度最大的结点为 ( ) 号结点。

A. 3

B. 4

C. 5

D. 6



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 T，错误填 F；除特殊说明外，判断题 1.5 分，选择题 3 分，共计 40 分）

(1)

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

```

2 using namespace std;
3 int x, y;
4 unsigned int n;
5 int main() {
6     cin >> n >> x >> y;
7     unsigned int mask = 0xff;
8     int x8 = x << 3;
9     int y8 = y << 3;
10    unsigned int nx = (n >> x8) & mask, ny = (n >> y8) & mask;
11    n &= (~(mask << x8));
12    n &= (~(mask << y8));
13    n |= (nx << y8);
14    n |= (ny << x8);
15    cout << "0x";
16    cout << std::hex << n << endl;
17    return 0;
18 }

```

假设输入的  $n$  是 32 位无符号整数范围内的整数， $x, y$  是不超过 3 的自然数，完成下面的判断题和单选题。

• 判断题

16. 代码中 `mask` 变量的值转化为二进制的低 16 位结果是 `0000 0000 1111 1111`。( )
17. 当输入 `x=0` 的时候，`nx` 表示 `n` 中最低八位对应的字节的数据。( )
18. 去掉程序第 11 行至第 12 行中 `(~(mask << x8))` 和 `(~(mask << y8))` 两处中的最内层括号不会改变程序的结果。( )

• 单选题

19. 当输入为“15078 0 1”时，变量 `nx, ny` 的值分别为多少？( )  
 (提示：十进制数 15078 与十六进制数 3AE6 相同)
- A. 0xE6, 0x3A    B. 0x6, 0xE0    C. 0x6, 0xE    D. 0x6, 0xA
20. 当输入为“23270 0 1”时，输出为( )。  
 (提示：十进制数 23270 与十六进制数 5AE6 相同)
- A. 0x5A6E    B. 0x5E6A    C. 0xA56E    D. 0xE65A
21. 以下哪一个变量的类型修改可能影响程序的输出？( )
- A. 将 `x, y` 修改为 `unsigned int` 类型。
- B. 将 `x8, y8` 修改为 `short` 类型。
- C. 将 `mask` 修改为 `int` 类型。
- D. 将 `nx, ny` 修改为 `unsigned long long` 类型。

(2)

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int n, k;
5
6 int func(vector <int> &nums) {
7     int ret = 0;
8     for(int i = n; i > k; i--) {
9         if(nums[i] > nums[i - k]) {
10             swap(nums[i], nums[i - k]);
11             ret++;
12         }
13     }
14     return ret;
15 }
16
17 int main() {
18     cin >> n >> k;
19     vector <int> a(n + 1, 0);
20     for(int i = 1; i <= n; i++)
21         cin >> a[i];
22     int counter = 0, previous = -1;
23     while(counter != previous){
24         previous = counter;
25         counter += func(a);
26     }
27     for(int i = 1; i <= n; i++)
28         cout << a[i] << ", ";
29     cout << endl << counter << endl;
30     return 0;
31 }
```

假设输入的  $n, k$  是不超过  $100000$  的正整数，输入的  $a[i]$  是不超过  $10^9$  的整数， $k$  小于等于  $n$ ，完成下面的判断题和单选题：

• 判断题

22. 当输入的  $k$  为  $1$ ，程序将  $a$  从小到大排序。( )
23. 在题目限制的输入规模下， $counter$  可能会溢出。( )

24. (1分) 当输入为“8 1 1 9 2 3 4 6 8 7”，输出共有 18 个可见字符。( )

• 单选题

25. 当输入的  $k$  为 1，该程序的排序方法最接近 ( )。

A. 冒泡排序      B. 选择排序      C. 计数排序      D. 插入排序

26. 该程序的时间复杂度为 ( )。

A.  $O(n + k^2)$       B.  $O(n^2)$       C.  $O(nk)$       D.  $O\left(\frac{n^2}{k}\right)$

27. 当输入为“8 3 1 5 2 6 3 7 4 8”，输出的第一行第三个数字为 ( )。

A. 2      B. 6      C. 7      D. 8

(3)

```
1 #include <iostream>
2 #include <vector>
3 #include <queue>
4 using namespace std;
5 const int MAXN = 200001;
6 int main() {
7     int n, m, l, r, w;
8     cin >> n >> m;
9     vector <int> dist(MAXN, -1);
10    vector <bool> vis(MAXN, false);
11    vector <vector <pair<int, int> > > go(MAXN);
12    for(int i = 1; i <= m; i++) {
13        cin >> l >> r >> w;
14        go[l].push_back(make_pair(r + 1, w));
15        go[r + 1].push_back(make_pair(l, -w));
16    }
17    queue <int> q;
18    dist[1] = 0, vis[1] = true;
19    q.push(1);
20    while(!q.empty()) {
21        int x = q.front(); q.pop();
22        for(auto i : go[x]) {
23            if(!vis[i.first]) {
24                vis[i.first] = true;
25                dist[i.first] = dist[x] + i.second;
26                q.push(i.first);
27            }
28        }
29    }
```

```

28     }
29 }
30 if(dist[n + 1] == -1) cout << "sorry" << endl;
31 else cout << dist[n + 1] << endl;
32 return 0;
33 }

```

假设输入的  $n, m$  是不超过 200000 的正整数，程序第 13 行每次输入的  $l, r$  保证  $l \leq r$ ，完成下面的判断题和单选题：

• 判断题

28. 交换程序的第 13 行与第 14 行，不影响程序运行的结果。( )
29. 输入的  $r$  的最大值为  $n$  时，程序可以正常运行。( )
30. 在程序的第 16 行至第 28 行，相同的数可能重复进入队列。( )

• 选择题

31. 当输入的  $l$  最小值为  $x$ ，输入的  $r$  的最大值为  $y$ ，最多有 ( ) 个元素进入过队列。  
A. 1                      B.  $y - x$                       C.  $y - x + 1$                       D.  $y - x + 2$
32. 当输入的  $n$  为偶数，且  $r=l+1$  时， $m$  至少为 ( ) 时输出不为 sorry。  
A.  $n / 2$                       B.  $n / 2 + 1$                       C.  $n / 2 - 1$                       D.  $n$
33. 当输入为 “5 3 1 3 4 3 4 2 4 5 3” 时，输出为 ( )。  
A. 4                      B. 5                      C. 6                      D. 7

三、完善程序（单选题，每小题 3 分，共计 30 分）

(1) (优美的进制) 问题：给出整数  $n$ ； $k$  进制是优美的，当且仅当  $n$  在  $k$  进制下至少有两位，且每一位的数值都不同。求对于给定的  $n$ ，有哪些进制是优美的，不存在则输出 -1。

试补全程序。

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int MAXN = 100000;
4 int n;
5 int vis[MAXN], a[MAXN];
6 vector<int> ans;
7 int check(int k) {
8     int x = n, top = 0;
9     for (int i = 0; i <= k; i++) vis[i] = 0;
10    while (①) {
11        a[++top] = ②;
12        x = ③;
13    }

```

```

14     if (top < 2)
15         return 0;
16     for (int i = 1; i <= top; i++) {
17         if (④)
18             return 0;
19         vis[a[i]] = 1;
20     }
21     return 1;
22 }
23 int main() {
24     cin >> n;
25     for (int i = ⑤; i <= n; i++) {
26         if (check(i))
27             ans.push_back(i);
28     }
29     if (ans.empty()) {
30         cout << -1;
31     }
32     for (int i = 0; i < ans.size(); i++)
33         cout << ans[i] << " ";
34     return 0;
35 }

```

34. ①处应填 ( )

- A.  $x > 0$                       B.  $x > 1$                       C.  $x / k > 0$                       D.  $x / k > 1$

35. ②处应填 ( )

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

36. ③处应填 ( )

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

37. ④处应填 ( )。

- A.  $vis[i] == 1$                               B.  $vis[a[i]] == 0$   
C.  $vis[i] == 0$                               D.  $vis[a[i]] == 1$

38. ⑤处应填 ( )。

- A. 1                                      B.  $n - 1$                               C. 2                                      D. 0



(2) (好运的日期) 一个日期可以用  $x$  年  $y$  月  $z$  日来表示。我们称一个日期是好运的, 当且仅当  $xy(w-z+1)$  为质数, 其中  $w$  为  $x$  年  $y$  月的总天数。输入  $x, y, z$ , 判断其对应的日期是否好运。

保证  $x$  是不超过 2024 的正整数,  $y$  是不超过 12 的正整数,  $x, y, z$  可以构成一个合法的日期。

试补全线性筛法算法, 空间限制 512MiB。

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 const int MAXW = ①;
5 const int days[13] = {0, 31, 0, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
6 int prime[MAXW], cnt;
7 bool not_prime[MAXW];
8
9 void linear_prime(int n) {
10     --n;
11     not_prime[0] = not_prime[1] = true;
12     for(int i = 2; i <= n; i++) {
13         if(not_prime[i] == false)
14             prime[++cnt]=i;
15         for(int j = 1; ②; j++) {
16             not_prime[i * prime[j]] = 1;
17             if(i % prime[j] == 0)
18                 ③;
19         }
20     }
21 }
22 bool check(int n) {
23     return ④;
24 }
25
26 int main() {
27     linear_prime(MAXW);
28     int x, y, z, w;
29     cin >> x >>y >> z;
30     if(y == ⑤)
31         w = check(x) ? 29 : 28;
```

```
32     else
33         w = days[y];
34     if(not_prime[x * y * (w - z + 1)])
35         cout << "unlucky" << endl;
36     else
37         cout << "lucky" << endl;
38     return 0;
39 }
```

39. ①处可以填 ( )

- A. 753005
- B. 10000000000
- C. 725041
- D. 2024

40. ②处应填 ( )

- A.  $j \leq cnt$
- B.  $i * prime[j] \leq n$
- C.  $(j \leq cnt) \ \&\& \ (i * prime[j] \leq n)$
- D.  $(i \leq cnt) \ \&\& \ (prime[i] * prime[j] \leq n)$

41. ③处应填 ( )

- A. `not_prime[i] = true`
- B. `return`
- C. `continue`
- D. `break`

42. ④处应填 ( )。

- A.  $n \% 4 == 0$
- B.  $(n \% 400 == 0 \ || \ (n \% 4 == 0 \ \&\& \ n \% 100 != 0))$
- C.  $(n \% 4 == 0 \ \&\& \ n \% 100 != 0)$
- D.  $(n \% 100 == 0 \ || \ (n \% 4 == 0 \ \&\& \ n \% 100 != 0))$

43. ⑤处应填 ( )。

- A. 1
- B. 7
- C. 8
- D. 2