

2023 年 12 月认证 C++ 六级真题解析

CCF 编程能力等级认证，英文名 Grade Examination of Software Programming (以下简称 GESP)，由中国计算机学会发起并主办，是为青少年计算机和编程学习者提供学业能力验证的平台。GESP 覆盖中小学全学段，符合条件的青少年均可参加认证。GESP 旨在提升青少年计算机和编程教育水平，推广和普及青少年计算机和编程教育。

GESP 考察语言为图形化 (Scratch) 编程、Python 编程及 C++ 编程，主要考察学生掌握相关编程知识和操作能力，熟悉编程各项基础知识和理论框架，通过设定不同等级的考试目标，让学生具备编程从简单的程序到复杂程序设计的编程能力，为后期专业化编程学习打下良好基础。

本次为大家带来的是 2023 年 12 月份 C++ 六级认证真题解析。

一、单选题（每题 2 分，共 30 分）

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	D	C	C	D	B	B	B	D	B	D	A	A	B	C	B

1、关于 C++ 类和对象的说法，错误的是()。

- A. 在 C++ 中，一切皆对象，即便是字面量如整数 5 等也是对象
- B. 在 C++ 中，可以自定义新的类，并实例化为新的对象
- C. 在 C++ 中，内置函数和自定义函数，都是类或者对象
- D. 在 C++ 中，可以在自定义函数中嵌套定义新的函数

【答案】D

【考纲知识点】语法知识

【解析】自定义函数中不能嵌套定义新函数。

2、有关下面C++代码的说法，错误的是()。

```
3 class Rectangle
4 {
5     private:
6         class Point
7         {
8             public:
9                 double x;
10                double y;
11        };
12        Point a, b, c, d;
13        double length;
14        double width;
15    public:
```

- A. C++中类内部可以嵌套定义类
- B. 在类中定义的类被称为内部类，定义类的类被称为外部类
- C. 内部类可以随便访问，不需要通过外部类来访问
- D. 代码中 Point 被称为内部类，可以通过外部类 Rectangle 来访问

Rectangle::Point

【答案】C

【考纲知识点】语法知识点

【解析】内部类是需要通过外部类访问的。

3、有关下面 C++代码的说法，正确的是()。

```
2 using namespace std;
3 class newClass
4 {
5     public:
6         static int objCounter;
7
8 };
9 int newClass::objCounter=2;
10 int main()
11 {
12     newClass classA;
13     newClass classB;
14     cout<<newClass::objCounter<<endl;
15     cout<<classB.objCounter<<endl;
16 }
```

- A. 第 14行代码错误，第 15行正确

- B. 第 15 行代码错误，第 14 行代码正确
- C. 第 14、15 两行代码都正确
- D. 第 6 行代码可修改为 `objCounter += 1`

【答案】 C

【考纲知识点】 语法知识点

【解析】 objCounter 是静态成员，2 种方式都可以访问。

4、有关下面C++代码的说法，错误的是()。

```
3 struct BiNode {
4     char data;
5     BiNode* lchid,*rchid;
6 };
7 class BiTree {
8     private:
9         BiNode* Creat();
10        void Release(BiNode* bt);
11        BiNode* root;
12    public:
13        BiTree() {
14            root = Creat();
15        }
16        ~BiTree() {
17            Release(root);
```

- A. 上列 C++ 代码适用于构造各种二叉树
- B. 代码 `struct BiNode` 用于构造二叉树的节点
- C. 代码 `BiTree(){root=Creat();}` 用于构造二叉树
- D. 析构函数不可以省略

【答案】 D

【考纲知识点】 语法知识点

【解析】 析构函数可以省略。系统可以默认一个析构函数。

5、基于第 4 题的定义，有关下面C++代码的说法正确的是()。

```
2 void Order(BiNode* bt) {  
3     if (bt == nullptr)  
4         return;  
5     else {  
6         cout << bt->data;  
7         Order(bt->lchid);  
8         Order(bt->rchid);  
9     }  
10 }
```

- A. 代码中 Order()函数是中序遍历二叉树的方法
- B. 代码中 Order()先访问根节点，然后对左子树进行前序遍历，再对右子树前序遍历
- C. 代码中 Order()先访问中序遍历左子树，然后访问根节点，最后则是中序遍历右子树
- D. 代码中 Order()先后序遍历左子树，然后后序遍历右子树，最后访问根节点

【答案】B

【考纲知识点】数据结构知识点

【解析】对二叉树的先序遍历，根左右。

6、有关下面C++代码的说法正确的是（ ）。

```
1 typedef struct LinkList {  
2  
3     int data;  
4  
5     LinkList* next;  
6  
7     LinkList* prev;  
8  
9 }LinkList,LinkNode;  
10 bool ListInit(LinkList* &L) {  
11  
12     L = new LinkNode;  
13     if (!L)return false;  
14  
15     L->next = NULL;  
16     L->prev = NULL;  
17     L->data = -1;  
18  
19     return true;  
20 }
```

- A. 上述代码构成单向链表
- B. 上述代码构成双向链表
- C. 上述代码构成循环链表
- D. 上述代码构成指针链表

【答案】 B

【考纲知识点】 数据结构知识点

【解析】 实现一个链表，指向前面和后面的元素。

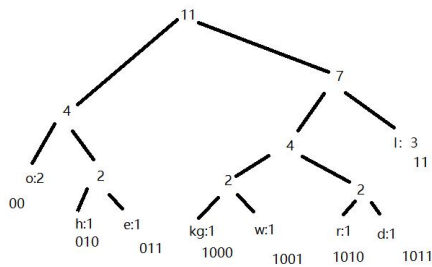
7、对 hello world 使用霍夫曼编码(Huffman Coding) ,最少 bit(比特)为()。

- A. 4
- B. 32
- C. 64
- D. 88

【答案】 B

【考纲知识点】 数据结构知识点

【解析】 赫夫曼编码。按照出现次数进行赫夫曼编码，如下图。共 32bit。



$$2*2+3+3+4+4+4+4+3*2=32$$

8、下面的 fiboA() 和 fiboB() 两个函数分别实现斐波那契数列，该数列第 1、第 2 项值为 1，其余各项分别为前两项之和。下面有关说法错误的是 ()。

```
3 int fiboA(int n)
4 {
5     if(n==0)
6         return 1;
7     if(n==1)
8         return 1;
9     else
10    {
11        return fiboA(n-1)+fiboA(n-2);
12    }
13 }
14 int fiboB(int n)
15 {
16     if( (n==0) || (n==1) ){
17         fiboB[n]=n;
18         return n;
19     }
20     else{
21         if(fiboB[n] == 0){
22             fiboB[n]=FiboB(n-1)+FiboB(n-2);
23         }
24         return fiboB[n];
25     }
26 }
```

- A. fiboA() 采用递归方式实现斐波那契数列
- B. fiboB() 采用动态规划算法实现斐波那契数列
- C. 当 N 值较大时， fiboA() 存在大量重复计算
- D. 由于 fiboA() 代码较短，其执行效率较高

【答案】D

【考纲知识点】算法知识点

【解析】fiboA 函数会存在重复计算，执行效率不高。

9、有关下面C++代码不正确的说法是（ ）。

```
int Depth(BiTree T)
{
    if (T == NULL)
    {
        return 0;
    }
    else
    {
        int m = Depth(T->lchild);
        int n = Depth(T->rchild);
        if (m > n)
        {
            return m + 1;
        }
        else
        {
            return n + 1;
        }
    }
}
```

- A. 该代码可用于求解二叉树的深度

- B. 代码中函数 `Depth()` 的参数 `T` 表示根节点，非根节点不可以作为参数
- C. 代码中函数 `Depth()` 采用了递归方法
- D. 代码中函数 `Depth()` 可用于求解各种形式的二叉树深度，要求该二叉树节点至少有 `left` 和 `right` 属性

【答案】 B

【考纲知识点】 数据结构知识点

【解析】 左右孩子也可以作为参数，不一定只有根节点。

10、下面有关树的存储，错误的是（ ）。

- A. 完全二叉树可以用 `list` 存储
- B. 一般二叉树都可以用 `list` 存储，空子树位置可以用 `None` 表示
- C. 满二叉树可以用 `list` 存储
- D. 树数据结构，都可以用 `list` 存储

【答案】 D

【考纲知识点】 数据结构知识点

【解析】 完全二叉树和满二叉树的节点下标是连续的，其他类型的节点数不确定，下标不一定连续，就不宜用 `list` 存储。

11、构造二叉树 `[1,2,3,null,4]`（ ）。

- A. `1(2()(4))(3)`
- B. `1(2(3)()(4))`
- C. `(1,2(3),(4))`
- D. `(1,(2)(3),(4))`

【答案】 A

【考纲知识点】 数据结构知识点

【解析】 构造二叉树，用括号表示法。1 是根结点，其左右孩子为 2，3，写成 `1(2)(3)`。用同样的思路构建左子树和右子树，选 A。

12、下面有关布尔类型的函数的说法，正确的是（ ）。

- A. bool 类型函数只能返回 0 或者 1 两种值
- B. bool 类型函数可以返回任何整数值
- C. bool 类型函数必须有参数传递
- D. bool 类型函数没有返回值

【答案】A

【考纲知识点】语言函数知识点

【解析】bool 类型函数只能返回布尔类型，布尔值只有 0 或者 1。

13、通讯卫星在通信网络系统中主要起到（ ）的作用。

- A. 信息过滤
- B. 信号中继
- C. 避免攻击
- D. 数据加密

【答案】B

【考纲知识点】计算机基础知识

【解析】本题考察计算机基础知识。通信卫星可以转发无线电信号，实现通信地球站间或地球站与航天器间的无线电通信，因此具有信号中继作用。选 B。

14、小杨想编写一个判断任意输入的整数 N 是否为素数的程序，下面哪个方法不合适？（ ）

- A. 埃氏筛法
- B. 线性筛法
- C. 二分答案
- D. 枚举法

【答案】C

【考纲知识点】数学知识

【解析】本题考察数学知识。线筛和埃筛都可以判断素数，枚举也可以，二分规模减半，不能合理判断。

15、内排序有不同的类别，下面哪种排序算法和冒泡排序是同一类？（ ）

- A. 希尔排序
- B. 快速排序
- C. 堆排序
- D. 插入排序

【答案】B

【考纲知识点】排序知识

【解析】冒泡排序和快速排序，都属于交换排序。

交换排序：冒泡排序、快速排序

选择排序：简单选择排序、堆排序

插入排序：直接插入排序、希尔排序

归并排序

基数排序

二、判断题（每题 2 分，共 20 分）

题号	1	2	3	4	5	6	7	8	9	10
答案	√	√	√	√	×	×	√	√	√	√

1、在面向对象中，方法在 C++ 的 class 中表现为 class 内定义的函数。（ ）

【答案】正确

【考纲知识点】c++语言知识

【解析】方法在 c++ 中也就是类内的函数。

2、C++ 类的定义中，可以没有构造函数，会给出默认的构造函数（ ）

【答案】正确

【考纲知识点】c++语言知识

【解析】构造函数没有，系统会给一个默认的构造函数。

3、如果某个 C++ 对象（ object）支持下标运算符（方括号运算符），则该对象在所对应 class 中以成员函数的形式 进行了重载。（ ）

【答案】正确

【考纲知识点】c++语言知识

【解析】标准库类型 `vector` 和 `string` 都重载了下标运算符。

4、深度优先搜索（DFS，Depth First Search 的简写）属于图算法，其过程是对每一个可能的分支路径深入到不能再深入为止，而且每个节点只能访问一次。（ ）

【答案】正确

【考纲知识点】算法知识

【解析】深搜就是“能深则深、不能深则退”。

5、哈夫曼编码（Huffman Coding）具有唯一性，因此有确定的压缩率。（ ）

【答案】错误

【考纲知识点】算法知识

【解析】哈夫曼编码确实是唯一的。但是，这并不意味着它总是有确定的压缩率。压缩率通常定义为压缩后数据大小与原始数据大小的比值。对于哈夫曼编码来说，压缩率会受到输入数据的特性（如字符频率分布）的影响。不同的输入数据，即使使用相同的哈夫曼编码算法，也可能得到不同的压缩率。

6、在下面C++代码中，由于删除了变量 ptr，因此 ptr 所对应的数据也随之删除，故第 8行代码被执行时，将报错。（ ）

```
5   int * ptr=new int(10);
6   cout<<*ptr<<endl;
7   delete ptr;
8   cout<<*ptr<<endl;
```

【答案】错误

【考纲知识点】语法知识

【解析】一个指针的指向对象已被删除，那么就成了悬空指针。指针本身仍然存在，但它的指向变得不确定，因为原来的内存已经被释放并可能被重新分配给其他用途。可能会导致程序崩溃，但是可能正常执行。

7、二叉搜索树查找的平均时间复杂度为。（ ）

【答案】正确

【考纲知识点】数据结构知识

【解析】若它的左子树不空，则左子树上所有结点的值均小于它的根结点的值；若它的右子树不空，则右子树上所有结点的值均大于它的根结点的值；它的左、右子树也分别为二叉排序树。平均情况下是一分为二。

8、二叉搜索树可以是空树（没有任何节点）或者单节点树（只有一个节点），或者多节点。如果是多节点，则左节点的值小于父节点的值，右节点的值大于父节点的值，由此推理，右节点树的值都大于根节点的值，左节点树的值都小于根节点的值。（ ）

【答案】正确

【考纲知识点】数据结构知识

【解析】若它的左子树不空，则左子树上所有结点的值均小于它的根结点的值；若它的右子树不空，则右子树上所有结点的值均大于它的根结点的值；它的左、右子树也分别为二叉排序树。平均情况下是一分为二。

9、小杨想写一个程序来算出正整数 N 有多少个因数，经过思考他写出了重复没有超过 $N/2$ 次的循环就能够算出来了。（ ）

【答案】正确

【考纲知识点】算法知识

【解析】可以的。 N 除了自身，只能有 1 个大于等于 $n/2$ 的因子。

10、同样的整数序列分别保存在单链表和双向链中，这两种链表上的简单冒泡排序的复杂度相同。（ ）

【答案】正确

【考纲知识点】算法知识

【解析】都是相邻的数字交换。

三、编程题（每题 25 分，共 50 分）

题号	1	2
答案		

1、闯关游戏

问题描述

你来到了一个闯关游戏。

这个游戏总共有 N 关，每关都有 M 个通道，你需要选择一个通道并通往后续关卡。其中，第 i 个通道可以让你前进 a_i 关，也就是说，如果你现在在第 x 关，那么选择第 i 个通道后，你将直接来到第 $x+a_i$ 关（特别地，如果 $x+a_i \geq N$ ，那么你就通关了）。此外，当你顺利离开第 s 关时，你还将获得 b_s 分。

游戏开始时，你在第 0 关。请问，你通关时最多能获得多少总分？

输入描述

第一行两个整数 N, M ，分别表示关卡数量和每关的通道数量。

接下来一行 M 个用单个空格隔开的整数 a_0, a_1, \dots, a_{M-1} 。保证 $1 \leq a_i \leq N$ 。

接下来一行 N 个用单个空格隔开的整数 b_0, b_1, \dots, b_{N-1} 。保证 $|b_i| \leq 10^5$ 。

输出描述

一行一个整数，表示你通关时最多能够获得的分数。

特别提醒

在常规程序中，输入、输出时提供提示是好习惯。但在本场考试中，由于系统限定，请不要在输入、输出中附带任何提示信息。

样例输入 1

```
1 6 2
2 2 3
3 1 0 30 100 30 30
```

样例输出 1

```
1 131
```

样例解释 1

你可以在第 0 关选择第 1 个通道，获得 1 分并来到第 3 关;随后再选择第 0 个通道，获得 100 分并来到第 5 关;最后任选一个通道，都可以获得 30 分并通关。

如此，总得分为 $1 + 100 + 30 = 131$ 。

样例输入 2

```
1 6 2
2 2 3
3 1 0 30 100 30 -1
```

样例输出 2

```
1 101
```

样例解释 2

请注意，一些关卡的得分可能是负数。

数据规模

对于 20%的测试点，保证 $M = 1$ 。

对于 40%的测试点，保证 $N \leq 20$ ；保证 $M \leq 2$ 。

对于所有测试点，保证 $N \leq 10^4$ ；保证 $M \leq 100$ 。

【题目大意】从 0 关出发，每次都有 m 种选择，选的第 i 个关卡，到达的下一个关卡是 $0 + m_i$ 。当关卡的总和大于等于 N 的时候，便停止游戏。注意，离开第 i 个关卡，可以获得关卡的分数。

【考纲知识点】循环知识，动态规划

【解题思路】可以考虑动态规划。每关都是一个阶段，每关增加的都是一个正数，关数增加是单向的。保存好某个阶段的状态，因为是离开才能获得分数，因此不能加上当前关卡的分数。可求出所有阶段的最值，因为可能存在某关是负值，要求出所有状态的最大值。

【参考程序】

```
1 #include <cstdio>
2 #include <cstdlib>
3 #include <cstring>
4 #include <algorithm>
5 #include <string>
6 #include <map>
7 #include <iostream>
8 #include <cmath>
9 using namespace std;
10 const int N = 10005;
11 const int M = 105;
12 const int inf = 0x3f3f3f3f;
13 int a[M], b[N], f[N];
14 int main() {
15     // freopen("data/1.in", "r", stdin);
16     int n, m;
17     scanf("%d%d", &n, &m);
18     for (int i = 1; i <= m; i++)
19         scanf("%d", &a[i]);
20     for (int i = 0; i < n; i++)
21         scanf("%d", &b[i]);
22
23     memset(f, -0x3f, sizeof(f));
24     f[0] = 0;
25     for (int i = 1; i < n; i++)
26         for (int j = 1; j <= m; j++)
27             if (i - a[j] >= 0)
28                 f[i] = max(f[i], f[i - a[j]] + b[i - a[j]]);
29
30     int ans = -inf;
31     for (int i = 0; i < n; i++)
32         for (int j = 1; j <= m; j++)
33             if (i + a[j] >= n) {
34                 ans = max(ans, f[i] + b[i]);
35                 break ;
36             }
37
38     cout << ans << endl;
39     return 0;
40 }
```

2、工作沟通

问题描述

某公司有 N 名员工，编号从 0 至 $N-1$ 。其中，除了 0 号员工是老板，其余每名员工都有一个直接领导。我们假设编号为 i 的员工的直接领导是 f_i 。

该公司有严格的管理制度，每位员工只能受到本人或本人直接领导或间接领导的管理。具体来说，规定员工 x 可以管理员工 y ，当且仅当 $x=y$ ，或 $x = f_y$ ，或 x 可以管理 f_y 。特别地， 0 号员工老板只能自我管理，无法由其他任何员工管理。

现在，有一些同事要开展合作，他们希望找到一位同事来主持这场合作，这位同

事必须能够管理参与合作的所有同事。如果有多名满足这一条件的员工，他们希望找到编号最大的员工。你能帮帮他们吗？

输入描述

第一行一个整数 N ，表示员工的数量。

第二行 $N-1$ 个用空格隔开的正整数，依次为 f_1, f_2, \dots, f_{N-1} 。

第三行一个整数 Q ，表示共有 Q 场合作需要安排。

接下来 Q 行，每行描述一场合作：开头是一个整数 m ($2 \leq m \leq N$)，表示参与本次合作的员工数量；接着是 m 个整数，依次表示参与本次合作的员工编号（保证编号合法且不重复）。

保证公司结构合法，即不存在任意一名员工，其本人是自己的直接或间接领导。

输出描述

输出 Q 行，每行一个整数，依次为每场合作的主持人选。

特别提醒

在常规程序中，输入、输出时提供提示是好习惯。但在本场考试中，由于系统限定，请不要在输入、输出中附带任何提示信息。

样例输入 1

```
1 5
2 0 0 2 2
3 3
4 2 3 4
5 3 2 3 4
6 2 1 4
```

样例输出 1

```
1 2
2 2
3 0
```

样例解释

对于第一场合作，员工 3, 4 有共同领导 2，可以主持合作。

对于第二场合作，员工 2 本人即可以管理所有参与者。

对于第三场合作，只有 0 号老板才能管理所有员工。

样例输入 2

```
1 | 7
2 | 0 1 0 2 1 2
3 | 5
4 | 2 4 6
5 | 2 4 5
6 | 3 4 5 6
7 | 4 2 4 5 6
8 | 2 3 4
```

样例输出 2

```
1 | 2
2 | 1
3 | 1
4 | 1
5 | 0
```

数据规模

对于 50%的测试点，保证 $N \leq 50$ 。

对于所有测试点，保证 $3 \leq N \leq 300$; $Q \leq 100$ 。

【题目大意】0号是老板，一个人可以管理自己，管理直接下属，管理下属的下属。可以根据关系建立起树结构。现在要求有 m 个人自己组织会议，要找一个人主持会议。如果 m 个人中能主持就从 m 个人中选，不能就找上级领导，有多个选择，就求出最大的编号。注意，参会和主持的角色不一样。

【考纲知识点】循环知识，树结构，深搜，有根树

【解题思路】首先要建立树结构，根据父子关系，求出节点的结构，并求出对应的层数。针对每次查询，先看一下 m 个人中，层数最低的能不能主持会议，不能就从上级中选取，结果最小就是 0 号来主持。注意，要编号最大的，因此在筛选的时候从大到小去筛选。根据层中的第 i 个节点去判断能否访问到 m 中的每个孩子，如果是，那就可以主持。假设 3 和 1 都符合，也在同一层，但是倒序查找，找到 3 就退出了。符合最终答案。

【参考程序】



GESP

```
1 #include <cstdio>
2 #include <cstdlib>
3 #include <cstring>
4 #include <algorithm>
5 #include <string>
6 #include <map>
7 #include <iostream>
8 #include <cmath>
9 #include <vector>
10 using namespace std;
11 const int N = 305;
12 int fa[N], dep[N];
13 bool vis[N];
14 vector<int> ch[N];
15 int getdep(int x) {
16     return x == 0 ? 0 : getdep(fa[x]) + 1;
17 }
18 void dfs(int x) {
19     vis[x] = 1;
20     for (int y : ch[x])
21         dfs(y);
22 }
23 bool check(int x, int n, const vector<int> &vec) {
24     for(int i = 0; i <= n; i++)
25         vis[i] = 0;
26     dfs(x);
27     for (int y : vec)
28         if(! vis[y])
29             return 0;
30     return 1;
31 }
32 int main() {
33     // freopen("data/1.in", "r", stdin);
34     // freopen("tmp.txt", "w", stdout);
35     int n;
36     scanf("%d", &n);
37     for(int i = 1; i <= n; i++) {
38         scanf("%d", &fa[i]);
39         ch[fa[i]].push_back(i);
40     }
41     for(int i = 1; i <= n; i++)
42         dep[i] = getdep(i);
43
44     int q;
45     scanf("%d", &q);
46     while(q --) {
47         int m, mnd = n + 1;
48         scanf("%d", &m);
49         vector<int> vec(m);
50         for(int i = 0; i < m; i++) {
51             scanf("%d", &vec[i]);
52             mnd = min(mnd, dep[vec[i]]);
53         }
54         for (int i = n - 1; i >= 0; i --)
55             if (dep[i] <= mnd && check(i, n, vec)) {
56                 printf("%d\n", i);
57                 break ;
58             }
59     }
60     return 0;
61 }
```



GESP

CCF-GESP编程能力等级认证

Grade Examination of Software Programming

GESP 2024年3月认证

认证语言: C++/Python/图形化编程

报名时间: 2024年1月18日至3月5日24点截止

缴费时间: 2024年1月18日至3月5日24点截止

认证时间: 1-4级 2024年3月16日 上午 09:30-11:30

5-8级 2024年3月16日 下午 13:30-16:30

认证方式: 全国统一线下机考



扫码即刻报名

GESP面向全国征集授权服务中心和考点 (考点仅限公立校)

Scratch 欢迎申请, 扫码至官网了解更多

【联系我们】

1. GESP 微信：关注“CCF GESP”公众号，将问题以文字方式留言即可得到回复。

2. GESP 邮箱：gesp@ccf.org.cn

注：请在邮件中详细描述咨询的问题并留下考生的联系方式及姓名、身份证号，以便及时有效处理。

3. GESP 电话：0512-67656856

咨询时间：周一至周五(法定节假日除外)：上午 8:30-12:00；下午 13:00-17:30

GESP 第五期认证报名已启动，扫描下方二维码，关注 GESP 公众号即可报名

