

GESP C++ 四级模拟试题 1

一、 选择题

1. (2023年6月) 高级语言编写的程序需要经过以下()操作, 可以生成在计算机上运行的可执行代码。
 - A. 编辑
 - B. 保存
 - C. 调试
 - D. 编译
2. (2023年12月) 某公司新出了一款无人驾驶的小汽车, 通过声控智能驾驶系统, 乘客只要告诉汽车目的地, 车子就能自动选择一条优化路线, 告诉乘客后驶达那里。请问下面哪项**不是**驾驶系统完成选路所必须的()。
 - A. 麦克风
 - B. 扬声器
 - C. 油量表
 - D. 传感器
3. (2023年12月) 下列C++语句执行以后结果是true的是()。
 - A. 3&&false
 - B. 5&&2
 - C. 101&&000
 - D. 4&&>true
4. (2023年6月) 下列关于 C++语言中指针的叙述, **不正确**的是()。
 - A. 指针变量中存储的是内存地址。
 - B. 定义指针变量时必须指定其指向的类型。
 - C. 指针变量只能指向基本类型变量, 不能指向指针变量。
 - D. 指针变量指向的内存地址不一定能够合法访问。
5. (2023年6月) 下列关于 C++语言中数组的叙述, **不正确**的是()。
 - A. 一维数组在内存中一定是连续存放的。
 - B. 二维数组是一维数组的一维数组。
 - C. 二维数组中的每个一维数组在内存中都是连续存放的。
 - D. 二维数组在内存中可以不是连续存放的。
6. (2023年9月) 下列关于C++语言中函数的叙述, 正确的是()。
 - A. 函数调用前必须定义。
 - B. 函数调用时必须提供足够的实际参数。
 - C. 函数定义前必须声明。
 - D. 函数声明只能写在函数调用前。
7. (2023年6月) 下列关于 C++语言中变量的叙述, 正确的是()。

- A. 变量定义后可以一直使用。
- B. 两个变量的变量名不能是相同的。
- C. 两个变量的变量名可以相同，但它们的类型必须是不同的。
- D. 两个变量的变量名可以相同，但它们的作用域必须是不同的。
8. (2023年9月) 如果n为int类型的变量，一个指针变量定义为int *p=&n; ,则下列说法正确的是()。
- A. 指针变量p的值与变量n是相同。
- B. 指针变量p的值与变量n的地址是相同的。
- C. 指针变量p指向的值为 'n'。
- D. 指针变量p指向的值与变量n的地址是相同的。
9. (2023年6月) 一个二维数组定义为 int array[5][3];, 则 array[1][2]和 array[2][1]在内存中的位置相差多少字节? ()
- A. 2字节
- B. 4字节
- C. 8字节
- D. 无法确定
10. (2023年9月) 如果 a 为 int 类型的变量，且 a 的值为6，则执行a = ~a; 之后，a 的值会是()。
- A. -6
- B. 6
- C. -7
- D. 7
11. (2023年6月) 一个数组定义为 int a[5] = {1, 2, 3, 4, 5};, 一个指针定义为 int *p = &a[2];, 则 执行 a[1] = *p;后，数组 a 中的值会变为()。
- A. {1, 3, 3, 4, 5}
- B. {2, 2, 3, 4, 5}
- C. {1, 2, 2, 4, 5}
- D. {1, 2, 3, 4, 5}
12. (2023年9月) 下列关于C++语言中异常处理的叙述，正确的是()。
- A. 一个try子句可以有多个catch子句与之对应。
- B. 如果try子句在执行时发生异常，就一定会进入某一个catch子句执行。
- C. 如果try子句中没有可能发生异常的语句，会产生编译错误。
- D. catch子句处理异常后，会重新执行与之对应的try子句。
13. (2023年6月) 在下列代码的横线处填写()，可以使得输出是 “20 10” 。

```
#include <iostream>
using namespace std;
void xchg(_____) { //在此处填入代码
    int t=*x;
```

```

    *x=*y;
    *y=t;
}
int main(){
    int a=10, b=20;
    xchg(&a,&b);
    cout<<a<<" "<<b<<endl;
    return 0;
}

```

- A. int x, int y
- B. int * x, int * y
- C. int a, int b
- D. int & a, int & b

14. (2023年12月) 下列C++代码输入1, 2, 3, 4 , 执行后, 将输出的是()。

```

string str="";
cin>>str;
int strlen=str.length();
for(int i=0; i<strlen; i++){
    if(str[i]<='9'&&str[i]>='0'){
        cout<<str[i];
    }else{
        cout<<"#";
    }
}
}

```

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

15. (2023年6月) 在下列代码的横线处填写(), 完成对有 n 个 int 类型元素的数组 array **由小到大**排序。

```

void SelectionSort(int array[], int n){
    int i,j,min,temp;
    for(int i=0; i<n-1; i++){
        min=i;
        for(int j=i+1; j<n; j++){
            if(_____)//在此处填写代码
                min=j;
        }
        temp=array[min];
    }
}

```

```

        array[min]=array[i];
        array[i]=temp;
    }
}

```

- A. array[min] > array[j]
- B. array[min] > array[i]
- C. min > array[j]
- D. min > array[i]

二、 判断题

1. (2023年12月) C++的内置函数 sort() 支持数组的局部排序。例如 int a={10,9,8,7,6,5,4,3,2,1} , 可以用 sort(a,a+5) ,排序成 {6,7,8,9,10,5,4,3,2,1} 。
2. (2023年6月) 数列 1, 1, 2, 3, 5, 8 ... 是以意大利数学家列昂纳多·斐波那契命名的数列, 从第三个数开始, 每个数是前面两项之和。如果计算该数列的第 n 项 (其中 $n > 3$) fib(n), 我们采用如下方法: ① 令 fib(1)=fib(2)=1 ②用循环 for i=3 to n 分别计算 f(i) ③输出 fib(n)。这体现了递推的编程思想。
3. (2023年9月) 在C++语言中, 每个变量都有其作用域。
4. (2023年6月) 在 C++语言中, 函数的参数默认以引用传递方式进行传递。
5. (2023年9月) 在C++语言中, 可以通过定义结构体, 定义一个新的数据类型。
6. (2023年6月) 如果希望记录 10 个最长为 99 字节的字符串, 可以将字符串数组定义为 char s[100][10];。
7. (2023年12月) 小杨最近在准备考GESP, 他用的Dev C++来练习和运行程序, 所以Dev C++也是一个小型操作系统。
8. (2023年6月) 字符常量'0'和'\0'是等价的。
9. (2023年12月) 执行C++代码 cout<<(5||2)的结果为1。
10. (2023年6月) 由于文件重定向操作, 程序员在使用 C++语言编写程序时无法确定通过 cout 输出的内容是否会被输出到屏幕上。

三、 编程题

1. 进制转换 (2023年9月)

【问题描述】

进制数指的是逢 N 进一的计数制。例如, 人们日常生活中大多使用十进制计数, 而计算机底层则一般使用二进制。除此之外, 八进制和十六进制在一些场合也是常用的计数制 (十六进制中, 一般使用字母 A 至 F 表示十至十五; 本题中, 十一进制到十五进制也是类似的)。

在本题中, 我们将给出 N 个不同进制的数。你需要分别把它们转换成十进制数。

【提示】

对于任意一个 L 位 K 进制数, 假设其最右边的数位为第 0 位, 最左边的数位为第 L-1 位, 我们只需要将其第 i 位的数码乘以权值 K^i , 再将每位的结果相加, 即可得到原 K 进制数对应的十进制数。下面是两个例子:

1. 八进制数 1362 对应的十进制数为 $1 \times 8^3 + 3 \times 8^2 + 6 \times 8^1 + 2 \times 8^0 = 754$

2. 十六进制数 3F0 对应的十进制为 $3 \times 16^2 + 15 \times 16^1 + 0 \times 16^0 = 1008$

【输入描述】

输入的第一行为一个十进制表示的整数 N。接下来 N 行，每行一个整数 K，随后是一个空格，紧接着是一个 K 进制数，表示需要转换的数。保证所有 K 进制数均由数字和大写字母组成，且不以 0 开头。保证 K 进制数合法。

保证 $N \leq 1000$ ，保证 $2 \leq K \leq 16$ 。

保证所有 K 进制数的位数不超过 9。

【输出描述】

输出 N 行，每一个十进制数，表示对应 K 进制数的十进制数值。

【样例输入 1】

2
8 1362
16 3F0

【样例输出 1】

754
1008

【样例输入 2】

2
2 11011
10 123456789

【样例输出 2】

27
123456789

2. 图像压缩 (2023年6月)

【问题描述】

图像是由很多的像素点组成的。如果用 0 表示黑，255 表示白，0 和 255 之间的值代表不同程度的灰色，则可以用一个字节表达一个像素（取值范围为十进制 0-255、十六进制 00-FF）。这样的像素组成的图像，称为 **256 级灰阶** 的灰度图像。

0

255



现在希望将 256 级灰阶的灰度图像压缩为 16 级灰阶，即每个像素的取值范围为十进制 0-15、十六进制 0-F。压缩规则为：

- ① 统计出每种灰阶的数量，取数量**最多的前 16 种灰阶**（如某种灰阶的数量与另外一种灰阶的数量相同，则以灰阶值**从小到大**为序），分别编号 **0-F**（最多的编号为 0，以此类推）。
- ② 其他灰阶转换到**最近的** 16 种灰阶之一，将某个点灰阶数与 16 种灰阶中的一种相

减，绝对值**最小**即为最近，如果绝对值相等，则编号较小的灰阶更近。

【输入描述】

输入第 1 行为一个正整数 N，表示接下来有 N 行数据组成一副 256 级灰阶的灰度图像。约定 $10 \leq N \leq 20$ 。

第 2 行开始的行，每行为长度相等且为**偶数的字符串**，每**两个字符**用十六进制表示一个像素。约定输入的灰度图像**至少**有 16 种灰阶。约定每行**最多** 20 个像素。

【输出描述】

第一行输出压缩选定的 16 种灰阶的十六进制编码，共计 32 个字符。

第二行开始的行，输出压缩后的图像，每个像素一位十六进制数表示压缩后的灰阶值。

【样例输入】

```
10
00FFCFAB00FFAC09071B5CCFAB76
00AFCBAB11FFAB09981D34CFAF56
01BFCEAB00FFAC0907F25FCFBA65
10FBCBAB11FFAB09981DF4CFCA67
00FFCBFB00FFAC0907A25CCFFC76
00FFCBAB1CFFCB09FC1AC4CFCF67
01FCCBAB00FFAC0F071A54CFBA65
10EFCBAB11FFAB09981B34CFCF67
01FFCBAB00FFAC0F071054CFAC76
1000CBAB11FFAB0A981B84CFCF66
```

【样例输出】

```
ABCFFF00CB09AC07101198011B6776FC
321032657CD10E
36409205ACC16D
B41032657FD16D
8F409205ACF14D
324F326570D1FE
3240C245FC411D
BF4032687CD16D
8F409205ACC11D
B240326878D16E
83409205ACE11D
```

【样例解释】

灰阶'AB'、'CF'和'FF'出现 14 次，'00'出现 10 次，'CB'出现 9 次，'09'出现 7 次，'AC'出现 6 次，'07'出现 5 次，'10'、'11'和'98'出现 4 次，'01'、'1B'、'67'、'76'和'FC'出现 3 次。