

GESP C++三级样题卷

(满分：100 分 考试时间：90 分钟)

学校：_____

姓名：_____

题目	一	二	三	总分
得分				

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															

- 下列关于负数的原码、反码、补码的描述中，正确的是()
 - 原码和反码互为按位取反（符号位除外），补码为反码加 1
 - 原码和反码互为按位取反（符号位除外），补码为原码加 1
 - 反码和补码互为按位取反（符号位除外），原码为反码加 1
 - 补码和原码互为按位取反（符号位除外），反码为补码加 1
- 一个 `int` 类型的值乘以 8，等价于以下哪个位运算？()
 - 左移 3 位
 - 右移 3 位
 - 左移 8 位
 - 右移 8 位
- 以下哪个属于 C++语言中的位运算符？()
 - `+`
 - `-`
 - `*`
 - `&`

4. 若有以下代码，则数组 **arr** 的长度是()

```
int arr[] = {1, 2, 3, 4, 5};
```

- A. 3
- B. 4
- C. 5
- D. 6

5. 在 C++语言中，可以定义一个一维整型数组的是()

- A. **int array[5];**
- B. **int array[];**
- C. **int[5] array;**
- D. **int[] array;**

6. 枚举算法的主要特点是()

- A. 以空间换时间
- B. 逐个尝试所有可能的解
- C. 动态规划
- D. 贪心策略

7. 对于一个十进制数 37，以下哪个是它的二进制表示()

- A. 10101
- B. 100101
- C. 101001
- D. 1000101

8. 下列关于十六进制的描述中，正确的是()

- A. 使用 0-9 和 A-F 表示
- B. 使用 0-9 和 A-E 表示

- C. 使用 1-9 和 A-F 表示
- D. 使用 1-9 和 A-E 表示

9. 下列哪个是 C++语言中用于获取字符串长度的函数()

- A. `length()`
- B. `len()`
- C. `getLength()`
- D. `strlen()`

10. 通常用下列哪种方式来描述算法?

- A. 汇编语言
- B. 伪代码
- C. SQL
- D. CSS

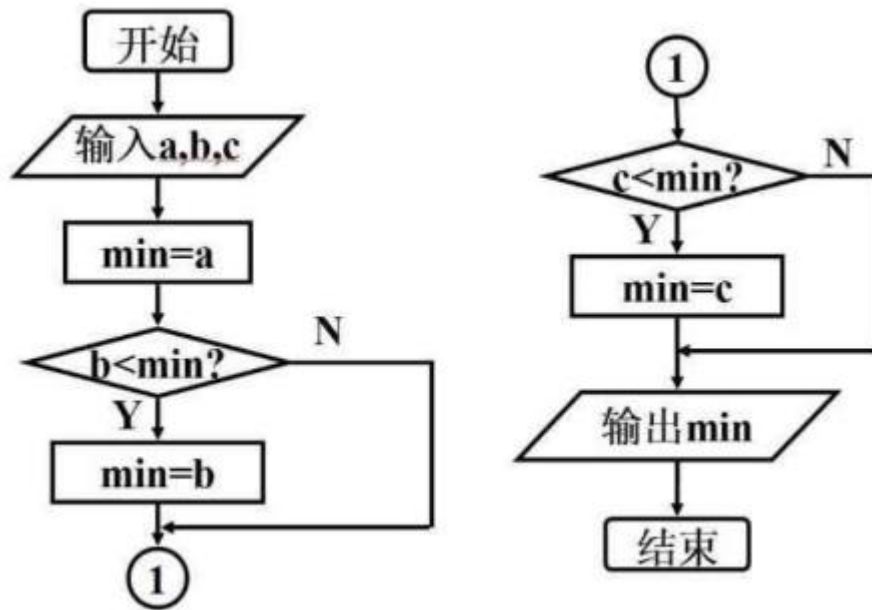
11. 如果 **a** 和 **b** 均为 `int` 类型的变量, 下列表达式能正确判断 “**a** 等于 0 且 **b** 等于 0” 的是 ()

- A. `((~a) && (~b))`
- B. `((a & b) == 0)`
- C. `((a | b) == 0)`
- D. `((a ^ b) == 0)`

12. 如果 **a** 为 `int` 类型的变量, 下列哪个表达式可以正确求出满足 “大于等于 **a** 且是 4 的倍数” 的整数中最小的?

- A. `(a & (~3))`
- B. `(a / 4 * 4)`
- C. `((a - 1) | 3) + 1`
- D. `(a << 2)`

13. 下面流程图，输入 1 2 3，会输出()



- A. 无输出
- B. 1
- C. 2
- D. 3

14. 在下列代码的横线处填写 ()，可以保证输出是“1357”，不会有多余字符。

```

#include <iostream>
#include <string>
using namespace std;
int main() {
    char str[] = "1234567";
    for ( _____ ) // 在此处填入代码
        cout << str[i];
    return 0;
}
  
```

- A. `int i = 0; i < strlen(str); i++`
- B. `int i = 0; str[i] != '\0'; i++`

- C. `int i = 1; i <= 7; i += 2`
- D. `int i = 0; i <= 6; i += 2`

15. 在下列代码的横线处填写 (), 可以使得输出是 “17 11”。

```
#include <iostream>
using namespace std;
int main() {
    int a = 11, b = 17;
    a = _____; // 在此处填入代码
    b = a ^ b;
    a = a ^ b;
    cout << a << " " << b << endl;
    return 0;
}
```

- A. `a + b`
- B. `a - b`
- C. `a ^ b`
- D. `a & b`