{

//================【开发篇】

1）d 其他，先跟他打声招呼，了解一下具体逻辑入参跟返回值，如果有新的想法就跟同时商量一下需不需要更改，再调用该功能

解释：避免代码冗余，如果同样的功能需求有多种方法实现，会导致后面的版本升级需要更改多个方法。

2）b~c~d 变量名问题、业务需求不一致，代码不一定相同

解释：每个人的写法不一样，公司要求的规范也不一样，搜索到的代码应该检查之后转为符合公司规范的代码使用

3）b

解释：变量名是让自己、同事、程序员能够快速的了解该变量的用途，如果随意命名，会导致该变量名的用途被曲解、遗忘

4）b

解释：注释能够让人理解即可。

5）b

解释：内存，分配给全局变量的内存要等程序结束才会释放并且当全局变量太多，代码太多，后期维护会很麻烦

6）b

解释：要看时间复杂度跟空间复杂度，短不代表精

7）b

解释：敲代码快，如果导致不易读或者浪费内存、运行速度，敲代码快也就没意义

8）c

解释：正常情况下，运行速度越快是好的，但是如果为了运行速度的快，而浪费大量的空间，在某些情况下，显然是不划算的

9）b

解释：合理的代码发布途径是 git

10）b

解释：合理的代码发布途径是 git

11）b

解释：合理的代码发布途径是 git

12）b

解释：合理的代码发布途径是 git

//================【测试篇】

1）d 根据低难度、中难度、高难度、实际情况分别测试

解释：写多种难度的测试，根据不同难度的测试结果分析程序的情况

2）b

解释:简单的测试也会出错，测试就要把所有的可能性测试清楚

3）a

解释：有多种难度的测试对该功能进行测试，放过一次，后续的测试也会对该功能继续测试，如果选择 b，出现的错误太多，导致测试不可信，那测试的意义也就不大了

4）d 使用自动化测试，低难度、中难度、高难度、实际情况分别测试，再人工简单的测试一下

解释：写多种难度的自动化测试来保证。

5）a~b~c~d

解释：都有可能，自己先排查清楚情况

6）f 查看是不是遇到 bug 或者检查一下是不是自己不小心改了测试

解释：abcde 都是从外部找原因，应该先从自己排查问题

7）a~b~c~f 根据低难度、中难度、高难度、实际情况分别测试

解释：测试的结果就是要检查出程序的 bug,测试的运行时间只要不是特别长，可以接受时间的浪费，开发文档或 type 规范中明令禁止的情况代表是还没解决的问题、或者有一定的要求，已经知道会有问题了，没有必要再浪费时间测试

//=================【综合篇】

1）e 使用自动化测试测试一下所有功能

解释：使用自动化测试检查整个程序，影响到的地方就会被测试出来，测试过后发现没问题就可以慢慢修改，这样就不会怕修改

2）b

解释：不一定，开发专注于开发，测试专注于测试，偶尔可以兼任，但不应该一起做，术业有专攻。

3）b

解释：应该是一起的，一边开发一边测试

4）数学

//=================【基本功篇】

1）

2）

3）O(n*ln[n])

4）O(1)

5）O(1)

6）O（logN)

7）O（n）

8）O（n）

9）

//=================【开放问题篇】

1）

{

先将拿到的资料整体的看一遍，对自己的任务有大概的了解，总结一下自己理解的任务，再去询问上级自己的任务是不是自己理解的

}

2)

{

大概的看一下代码有没有明显的 bug，再去写一个自动化脚本测试这个除法程序有没有 bug，再看一下使用这个除法程序的时间是不是比编程语言自带的除法更节省时间或者空间，是否有必要使用这个除法程序

附加题：使用搜索引擎查找编程语言自带的除法程序的时间复杂度和空间复杂度，或者直接查看源代码，算出时间复杂度和空间复杂度，再看一下这个除法程序的时间复杂度和空间复杂度，对比一下，是不是比编程语言自带的除法更节省时间或者空间，是否有必要使用这个除法程序

//该程序测试这个这个除法程序有没有 bug

```
using System;
```

```
namespace ConsoleApp1
{
    class Program
    {
        //假设这是别人自己写的一个 除法程序，并且声称比编程语言自带的 除法 更好
        static double division(double d1, double d2)
        {
            double d = d1 / d2;
            return d;
        }

        static void Main(string[] args)
        {
             try
             {
                 //检查该除法程序当d1,d2为正数的除法运算
                 for (int i = 0; i <= 100; i++)
                 {
                     Random rn1 = new Random();
                     Random rn2 = new Random();
                     double d1 = rn1.Next(10000000);
                     double d2 = rn2.Next(10000000);
                     //如果出现异常，方便查看出现异常的数值情况
                     Console.WriteLine(d1 + "/" + d2);
                     double d = division(d1, d2);
                     Console.WriteLine("d1,d2为正数的结果" + d);
                 }
                 //检查该除法程序当d1,d2为负数的除法运算
                 for (int i = 0; i <= 100; i++)
                 {
                     Random rn1 = new Random();
                     Random rn2 = new Random();
                     double d1 = rn1.Next(10000000)*-1;
                     double d2 = rn2.Next(10000000)*-1;
                     //如果出现异常，方便查看出现异常的数值情况
                     Console.WriteLine(d1 + "/" + d2);
                     double d = division(d1, d2);
                     Console.WriteLine("d1,d2为负数的结果" + d);
                 }
                 //检查该除法程序当d1为正数,d2为负数的除法运算
                 for (int i = 0; i <= 100; i++)
                 {
                     Random rn1 = new Random();
```

```
                Random rn2 = new Random();
                double d1 = rn1.Next(10000000);
                double d2 = rn2.Next(10000000)*-1;
                //如果出现异常，方便查看出现异常的数值情况
                Console.WriteLine(d1 + "/" + d2);
                double d = division(d1, d2);
                Console.WriteLine("d1为正数,d2为负数的结果" + d);
            }
            //检查该除法程序当d1为负数,d2为正数的除法运算
            for (int i = 0; i <= 100; i++)
            {
                Random rn1 = new Random();
                Random rn2 = new Random();
                double d1 = rn1.Next(10000000)*-1;
                double d2 = rn2.Next(10000000);
                //如果出现异常，方便查看出现异常的数值情况
                Console.WriteLine(d1 + "/" + d2);
                double d = division(d1, d2);
                Console.WriteLine("d1为负数,d2为正数的结果" + d);
            }
            //检查该除法程序当d1为0或者d2为0时的处理情况
            //d2为零是为了查看该除法程序当遇到d2为零的情况时，如何处理
            Random rn3 = new Random();
            double d3 = 0;
            double d4 = rn3.Next(10000000);
            double dR1 = division(d3, d4);
            Console.WriteLine("d1为0结果" + dR1);
            Random rn5 = new Random();
            double d5 = rn5.Next(10000000);
            double d6 = 0;
            double dR2 = division(d5, d6);
            Console.WriteLine("d2为0结果" + dR2);
        }
        catch (Exception e)
        {
            Console.WriteLine("出现异常" + e.ToString() + "该程序有bug");
        }
        Console.ReadKey();
    }
}
}
```

3)
{

```
//分数列表
static List<Integer> listNum = new ArrayList<>();
//答案列表
static List<String> listResult = new ArrayList<>();
//分数
static int grade=0;
//假设这是前端交卷时调用到的方法,每做一道题就提交一次
//num代表题号，result代表选择的答案，type,代表类型，是单选题还是多选题
//多选题返回格式为a~b...
static int test(int num,String result,String type) {

    if(type.equals("单选题")) {
        if(listResult.get(num-1).equals(result)) {
            grade +=listNum.get(num-1);
        }
    }else {
        //第一种情况，全对
        if(listResult.get(num-1).equals(result)) {
            grade += listNum.get(num-1);
        }else {
            //第二种情况，部分对或者全错
            String[] s1 = result.split("~");
            String[] s2 = listResult.get(num-1).split("~");
            int flag=0; //代表符合正确答案的个数，或者说答对了几个
            for(int i = 0;i <s1.length;i++) {
                for(int j = 0;j <s2.length;j++) {
                    if(s1.equals(s2)) {
                        flag += 1;
                    }
                }
            }
            //分数的给定
            switch (flag) {
                case 0:
                    grade +=0;
                    break;
                case 1:
                    grade +=1;
                    break;
                case 2:
                    grade +=2;
                    break;
```

```
                    case 3:
                        grade +=3;
                        break;
                }
            }
        }

        return grade;
    }
    public static void main(String[] args) {
        //假设有十道题
        Scanner sc = new Scanner(System.in);
        for(int i = 1;i <= 10;i++ ) {
            System.out.println("请输入第"+i+"道题的分数跟答案,多选题答案请用~隔
开,多选题分数应大于3分");
            listNum.add(sc.nextInt());
            listResult.add(sc.next());
            System.out.println();
        }
        System.out.println(grade);
    }}
}
```