

蓝桥杯全国大学生软件和信息技术大赛组委会

第十七届蓝桥杯全国大学生软件和信息技术大赛 软件赛（软件测试科目） 竞赛规则及说明

一、参赛对象

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）。

每位选手配备一名指导教师，同一名指导教师可指导多位选手。省赛和全国总决赛比赛后指导教师原则上不能更改。

二、组别设置

该科目设置大学组。

三、竞赛赛制

本届大赛采用校赛、省赛、全国总决赛三级竞赛体系。

校赛：由参赛院校自行组织并进行评审。

省赛：由大赛组委会统一组织。比赛时长为 4 小时。

全国总决赛：由大赛组委会统一组织。比赛时长为 4 小时。

详细赛程安排以组委会公布信息为准。

四、竞赛形式

1. 省赛、全国总决赛均采用封闭、限时方式举办。选手以个人为单位，独立进行作答。
2. 选手原则上需在线下赛点集中参赛。
3. 选手机器通过局域网连接到各个赛场的比赛服务器。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）。
4. 比赛系统以“服务器-浏览器”方式发放赛题、回收选手答案。

五、参赛选手机器环境

1. 硬件设施

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

2. 软件测试开发环境

工具套件：Office 工具套件（包括 Excel、Word）、WinRAR 5.9 解压缩工具。

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

六、赛题形式

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。

(1) 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，使用提供的 Excel 模板来填写答案。

(2) 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。

选手可自行选择 Java 或 Python 语言分别进行自动化测试和单元测试答题。

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，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 模板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。务必严格遵循规范，根据题意操作，否则可能影响最终阅卷的准确性。

七、赛题考查范围

赛题重点考察选手解决实际测试问题的综合能力，要求选手能够准确阅读、分析和理解需求，并灵活运用各类测试技术对被测对象实施有效测试及缺陷发现。具体涵盖功能测试中的黑盒测试用例设计方法、测试用例文档规范及级别定义；缺陷管理中的缺陷发现能力、缺陷报告编写规范及缺陷级别判定；自动化测试中的 Selenium 工具应用、Junit/Unittest 框架使用、Java/Python 编程基础及前端技术（HTML/CSS/JS）运用；以及单元测试中的代码理解能力、白盒测试方法及 Junit/Unittest 工具实践。通过多维度考核，全面评估选手在测试流程、技术应用和规范执行等方面的专业水平。具体考查知识范围详见软件赛竞赛大纲，链接：dasai.lanqiao.cn/notices/846。

八、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。选手应使用比赛指定的网页来提交答案，任何其他方式的提交（如邮件、U 盘）均不作为评测依据。

选手可在比赛中的任何时间查看自己之前提交的答案，也可以重新提交任何题目的答案，对于每个赛题，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。

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。

九、评分

全部题目将使用自动化测试技术完成机器自动评分。

对于文档类答案，将采用自动化测试技术机器自动评分 + 人工复核的方式。

对于代码类答案，将根据判分规则采用自动化测试技术完成机器自动评分。

十、历届真题及样题

1. 历届真题

参赛选手可在大赛官方网站-关于大赛-历届真题里查看以往蓝桥杯大赛的历届真题。链接：www.lanqiao.cn/courses/2786。

2. 样题

样题详见文档附录。

十一、奖项设置及评选办法

1. 省赛

省赛该组别设立一、二、三等奖，原则上各奖项的比例为 10%、15%、25%，总获奖比例不超过 50%。获奖比例仅作为参考，组委会专家组将根据赛题难易程度及整体答题情况，制定各奖项获奖最低分数线，未达到获奖最低分数线者不得奖。省赛一等奖选手获得全国总决赛参赛资格。

2. 全国总决赛

全国总决赛该组别设立一、二、三等奖，原则上各奖项的获奖比例为 10%、25%、40%，总获奖比例不超过 75%。获奖比例仅作为参考，组委会将根据赛题难易程度及整体答题情况，制定各奖项获奖最低分数线，未达到获奖最低分数线者不得奖。

十二、奖项查询

省赛及全国总决赛评审完成后，大赛组委会将在报名系统开放奖项查询。参赛选手若对奖项有异议，可在 3 个工作日内按照大赛组委会相关要求提出复核申请。

十三、监督反馈

详见《蓝桥杯全国大学生软件和信息技术大赛章程》。

十四、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，则取消竞赛奖项，收回获奖证书及奖品等，并在大赛官方网站公示。

2. 参赛选手应严格遵守蓝桥杯大赛比赛管理办法（链接：dasai.lanqiao.cn/notices/844），服从大赛组委会的指挥和安排，爱护竞赛场地的设备。未尽事宜请参照组委会在大赛官方网站公布的通知、章程、比赛管理办法及相关要求并遵照执行。

3. 赛项采用智能化评审系统自动评分与人工复核双轨校验机制，确保评审效率与精度。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，并把发现的缺陷记录到指定的文档中。

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】

【测试用例答案示例】

用例编号	模块名称	功能名称	用例方法	用例标题	前置条件	操作步骤	预期结果	用例优先级
YHZX-DL-001	用户中心	登录	等价类	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】。	成功登录系统，并跳转至个人中心。	高
YHZX-DL-002	用户中心	登录	边界值	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】。	登录失败，提示：用户名或密码错误。	中

【测试用例答案示例】

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“收获地址不能为空。”	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【悦购图书商城】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】	使用完整的 XPATH 进行元素定位 插入断言，使用用户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，建议使用完整的 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.firefox.FirefoxDriver;
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",
"driver/geckodriver.exe");
    //使用 FireFox 浏览器打开以下页面
    driver =new FirefoxDriver();
    //打开悦购图书商城前台页面
    driver.get("http://127.0.0.1/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“登录失败”
    Assert.assertEquals(driver.getTitle(), "张三", "登录失败");
    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，在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 testSubtract() {
        Assert.assertEquals(1, Calculator.subtract(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。

完整样题可以复制下方链接到浏览器中打开：

<https://www.lanqiao.cn/courses/18372/learning/>