

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。

