**武 汉 工 程 大 学**

**计算机科学与工程学院**

**《高级程序设计》实验报告[1]**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 专业班级 | |  | 实验时间 | |  | | |
| 学生学号 | |  | 实验地点 | |  | | |
| 学生姓名 | |  | 指导教师 | | 吴静 | | |
|  | | | | | | | |
| 实验项目 | | Java基础程序设计 | | | | | |
| 实验类别 | | 基础性 | 实验学时 | | **6** | | |
| 实验目的及要求 | | **实验目的**：  1．掌握设置 Java 程序运行环境的方法。 2. 了解 Java 语言的概貌。 3．熟悉运算符、表达式以及语句的使用。4. 掌握控制结构：条件分支语句、开关语句和循环语句 5. 掌握类与对象基础内容  **实验要求**：  1掌握运行 Java 程序的基本环境。  2掌握编写Java 程序基本方法。  2运行程序并分析结果 | | | | | |
|  | | | | | | | |
| 成 绩 评 定 表 | | | | | | | |
| 类 别 | | 评 分 标 准 | | 分值 | | 得分 | 合 计 |
| 上机表现 | | 按时出勤、遵守纪律  认真完成各项实验内容 | | **30**分 | |  |  |
| 报告质量 | 程序代码规范、功能正确  填写内容完整、体现收获 | | | **70**分 | |  |
| **说明：**  **评阅教师：**  **日 期： 2020 年 月 日** | | | | | | | |
| **实 验 内 容（-）Eclipse 环境搭建** （一）SDK 的下载与安装  1. 机器要求   Java 对机器运行要求不高。下面给出的是基于Windows 平台的机器要求。  硬件要求：CPU PII 以上，64M 内存，1OOM 硬盘空间即可。  软件要求：Windows98/Me/XP/NT/2000，IE 5 以上。   1. 下载SDK   为了建立基于SDK 的Java 运行环境，需要先下载Sun 的免费SDK 软件包。SDK 包含了一整套开发工具，其中包含对编程最有用的是Java 编译器、Applet 查看器和Java 解释器。  在浏览器中输入http://java.sun.com/，可看到图1.1 所示的oracle 公司网站页面。    图1.1 oracle 公司网站  在页面右边“Top Downloads”栏中可看到最新版的SDK 软件包（早期版本称为JDK），这里显示的是“Java SE 9.0.4”(当前最新版)。单击“[Java SE 9.0.4](http://www.oracle.com/technetwork/java/javase/downloads/index.html)”超链接，可看到图  1.2 所示页面。    图1.2 下载页面 一  点击页面上“Java Platform (JDK) 9”上方的图片，可以选择下载 jdk-9.0.4\_windows-x64\_bin.exe 软件包。注意，在下载前应该先阅读许可协议并选择接受该协议（ACCEPT）    图1.2 下载页面 二   1. 安装JDK   运行下载的“jdk-9.0.4\_windows-x64\_bin.exe”软件包，可安装JDK，在安装过程中可以设置安装路径及选择组件，系统默认的安装路径为C:\ j2sdk1.8.0\_31（这里选择E 盘），默认的组件选择是全部安装。   1. SDK 中的主要文件夹和文件   安装成功后，E:\ j2sdk1.8.0\_31 中的文件和子目录其中，bin 文件夹中包含编译器（javac.exe）、解释器（javac.exe）、Applet 查看器（appletviewer.exe）等可执行文件 （二）设置环境变量 SDK 中的工具都是命令行工具，需要从命令行即MS－DOS 提示符下运行它们。很多人可能会不习惯，但这是Sun 特意采取的策略，为的是把精力更多地投入到Java 语言本身而不是花在开发工具上。  为了能正确方便的使用SDK，可手工配置一些Windows 的环境变量，Windows 98 与Windows/xp/NT/2000 的配置略有不同。下面以Windows/xp/NT/2000为例说明相关配置。   1. 设置Windows XP/NT/2000   在桌面“我的电脑”图标上单击右键，选择“属性”菜单，将出现系统特性设置界面    图1.3 系统特性设置界面  在“高级”选项卡中单击“环境变量”按钮。将出现“环境变量”设置界面    图1.4 “环境变量”设置界面  新建名为“JAVA\_HOME”的变量名，变量值为之前安装jdk的目录，例如本人的为“E:\Program Files\Java\jdk1.8.0\_31”  在“系统变量”框中新建名为“classpath”的变量名，变量值为“.;%JAVA\_HOME%\lib;%JAVA\_HOME%\lib\tools.jar “（注意是”“里的内容）直接复制即可。  在已有的系统变量”path“的变量值加上”;%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin“（注意，每个变量值是以”；“隔开，变量值开头的分号就起这个作用）自此配置完成。    图1.5 系统变量”path“的变量值添加  设置完成后，单击“开始”→“所有程序”→“附件”→“命令提示符”打开DOS 窗口，在命令提示符下输入“java”或“javac”，回车后，如果出现其用法参数提示信息，则安装正确    图1.6 javac运行结果   * 1. 设置DOS 起始位置   为了在打开DOS 窗口时可以直接进入包含你开发的java 程序的路径下，下面以Windows XP/2000为例说明设置过程。  (1)首先建立一个存放java 源程序的路径和文件夹，如将实验中建立的java 源程序都存放在为“e:\java\程序”文件夹中。  (2)单击“开始”→“所有程序”→“附件”→“命令提示符”，选择“命令提示符”选项右击，在出现的下拉菜单中选择“属性”菜单，将打开“命令提示符 属性”对话框，如图1.7 所示。  (3)在“命令提示符 属性”对话框中选择“快捷方式”选项，在“起始位置”栏输入“e:\java\程序”，单击“确定”按钮。  进行了上面的设置后，打开DOS 窗口时将直接进入“e:\java\程序>”路径下。  如果没有进行上述设置，可使用DOS 命令，直接输入“e:”按回车进入到e 盘，再输入“cd java”到e 盘的java 子目录下，再输入“cd 程序”即可到“e:\java\程序>”路径下。    图1.7 “命令提示符 属性”对话框  3. 测试  启动一个shell窗口，输入下列命令：  C:\.>java -version  如果出现如图1-5所示的提示信息，说明安装成功。也就是说可以编译执行Java程序了。  Java提供了一些常用的语言工具，主要包括：  javac——编译器：用于将Java源代码文件（.java）编译成能被java解释器运行的类（字节码）文件（.class）  命令格式：C:\.>javac [选项] 源文件名  例：javac Test.java  java——解释器 即java虚拟机，用于java程序的解释执行，直接从类文件（字节码）执行java应用程序  命令格式：C:\.>java [选项] 类名  例：java Test  appletviewer——小应用程序浏览器：一种执行嵌入在HTML文件上的JavaApplet的Java浏览器，可用于程序的调试  命令格式：C:\.> appletviewer [选项] URL  javah——头文件生成器 :用于从java字节码生成C语言头文件和源文件，这些文件用来在JAVA的类中融入C语言的原生方法，完成从JAVA类中调用C++代码  javadoc——API文档生成器：可以根据java源码即说明生成html文档。  javap——类文件所汇编器： 用于显示编译类文件中的可访问功能和数据，同事显示字节码含义。  jdb——java语言调试器：可以逐行执行程序，设置断点和检查变量。  这些文件包括在c:\jdk1.5.x\bin目录中，并可以在任何目录中运行，前提是设置了运行程序的相应系统路径。  欣赏完这些Java程序所生成的图形之后，你是否有一些好奇呢？那么，让我们一起进入神奇的Java世界吧。 （四）编辑、编译和执行Java源程序 Java程序分为两种类型：一种是Java Application，它以main()方法作为程序入口，由Java解释器加载执行；  另一种是Java Applet，它通过把类名镶嵌到HTML文档中，通过浏览器或Appletviewer加载执行，它没有作为程序入口main()的方法。  初学者可以使用文本编辑器如UltraEdit，生成一个以java为后缀的文本文件。源文件必须经过编译，生成字节码的类文件后才能运行。Java的编译器是javac.  1)编辑Java源程序。  2)编译生成字节码文件。  3)解释与运行字节码文件。  **下面以经典的hello world为例。**  **例1-1** Java Application源程序。  1）打开记事本。这里使用的文本编辑器是记事本，它是Microsoft操作系统自带的文本编辑器。通过“开始”按钮，选择“程序”，再选择“附件”，在附件的清单中，选择“记事本”后就会弹出记事本的窗口。  2）编辑源程序。在打开的记事本窗口中输入下列的代码：  ----------------------------------------------------------------------------------------------------------------  //**Welcome.java**  public class Welcome{  public static void main(String[] args) //程序入口main()的方法  {  System.out.println("Hello World!");//输出字符串Hello World!  }}  ----------------------------------------------------------------------------------------------------------------  3)编译、运行源程序。  启动一个shell窗口，输入下列命令：  C:\.>javac Welcome.java  C:\.>java Welcome  即可运行该程序，运行结果如图1-8所示。    图1-8程序helloworld的运行结果  **注意：源方件存盘时，文件名必须与类名一致，并区分大小写，扩展名为.java。该例的类名为Welcome，所以文件名为Welcome.java。**  例1-1中我们编写的是一个Java Application源程序，下面同样以经典的hello world为例，编写一个Java Applet源程序。  Java中，Applet是使用Java语言编写的一段代码，它可以在浏览器环境中运行。它与Application的区别主要在于其执行方式的不同。Application是从其中的main()方法开始运行的，而Applet是在浏览器中运行的，必须创建一个HTML文件，通过编写HTML语言代码告诉浏览器载入何种Applet以及如何运行。  **例1-2** Java Applet源程序  1）编辑源程序。打开一个文本编辑器，输入下列代码：  ----------------------------------------------------------------------------------------------------------------  **// WelcomeApplet.java**  import java.awt.Graphics;  public class WelcomeApplet extends java.applet.Applet //继承Applet类  {  public void paint (Graphics g)  {  g.drawString("Welcome to Java!",10,10);  }  }  ----------------------------------------------------------------------------------------------------------------  2）保存源程序。源文件编辑好后存盘。文件名必须是WelcomeApplet.java.  3）编译源程序。打开一个shell窗口，进行编译。在命令行键入：  C:\code>javac WelcomeApple.class。  编译的结果是生成字节码文件WelcomeApple.class。  **注意：源文件存盘时，文件名必须与类名一致，并区分大小写，扩展名为.java。该例中的类名为WelcomeApple，所以文件名为WelcomeApple.java。**  4）运行源程序。Java Applet不能直接在命令行下执行，需要编写HTML文件，把Applet嵌入其中。打开一个文本编辑器输入名为helloworldApplet.html的文件源代码：  ----------------------------------------------------------------------------------------------------------------  <html>  <head>  <title> applet例子 </title>  </head>  <body>  <applet code=" WelcomeApplet.class" width=200 height=40>  </applet>  </body>  </html>  ---------------------------------------------------------------------------------------------------------------  **注意：此文件名可随意确定，该例中文件名为appletExample.html.**  有两种方式可以运行Java Applet:  1）在浏览器下运行，如IE、Netscape等，打开HTML文件，即会装载类文件并运行。  2）用appletviewer命令。本例中采用此方式，在命令行键入：  C:\code>appletviewer appletExample.html  这时屏幕上弹出一个窗口，显示：  以上内容请同学自行上机运行。  以下实验内容请参见《实验指导与习题解答》P2-P6的要求完成源代码的编辑并在恰当位置填写实验报告。  **实验内容（二）**  基本数据类型与数组：  1、完成《实验指导与习题解答》P8“实验1 输出希腊字母表”中“5.程序模板”中的【代码1】、【代码2】和【代码3】，将源程序编写在“Eclipse”开发环境中编译运行并观察结果。在代码1，2，3中复制具体的代码内容  【代码1】：  startPosition=(int)cStart;  【代码2】：  endPosition=(int)cEnd ;  【代码3】：  c=(char)i;  运行结果：    2. 完成《实验指导与习题解答》P10“实验2 数组的引用与元素”中“5.程序模板”中的【代码1】、【代码2】、【代码3】和【代码4】，将源程序编写在“Eclipse”开发环境中编译运行并观察结果。  【代码1】：  System.*out*.println(a.length);  【代码2】：  System.*out*.println(a);  【代码3】：  System.*out*.println(b.length);  【代码4】：  b[4]=a;  运行结果：    3. 完成《实验指导与习题解答》P11“实验3 遍历与复制数组”中“5.程序模板”中的【代码1】、【代码2】、【代码3】和【代码4】，将源程序编写在“Eclipse”开发环境中编译运行并观察结果。  【代码1】：  c = Arrays. *copyOf* (a,4) ;  【代码2】：  System.*out*.println( Arrays.*toString*(c));  【代码3】：  d = Arrays.*copyOfRange*(a,4,8);  【代码4】：  c[c.length-1]=-100;  运行结果：    Java 分支与循环语句  4. 完成《实验指导与习题解答》P15“实验1 回文数”中“5.程序模板”中的【代码】，将源程序编写在“Eclipse”开发环境中编译运行并观察结果。  【代码1】：  number<=99999&&number>=1  【代码2】：  d5=number/10000;  【代码3】：  d4=number%10000/1000;  【代码4】：  d3=number%1000/100;  【代码5】：  d5!=0  【代码6】：  d1==d5&&d2==d4  【代码7】：  d4!=0  【代码8】：  d1==d4&&d2==d3  【代码9】：  d3!=0  【代码10】：  d1==d3  运行结果：    5. 完成《实验指导与习题解答》P18“实验2 猜数字游戏”中“5.程序模板”【代码1】、【代码2】和【代码3】，将源程序编写在“Eclipse”开发环境中编译运行并观察结果。  【代码1】：  yourGuess!=realNumber  【代码2】：  yourGuess>realNumber  【代码3】：  yourGuess<realNumber  运行结果： | | | | | | | |

|  |
| --- |
| 实验总结 |
| （实验指导书中每个实验第七项：实验后的练习请将观察和思考的内容填写在本总结中）  试验一：（1）Error:(8, 17) java: 不兼容的类型: 从double转换到float可能会有损失   1. 不能（3）不能，-128   实验二：（1）编译有错，运行有错  （2）（3）      实验三：  （1）    （2）    实验四：  （1）    （2）    （3）    实验五：   1. 不可以 2. 不合理 |
|  |