

# Java程序设计编程作业-5

## 作业说明

请按照本文档的题目要求和代码框架完成编程作业。代码框架的示例结构如下所示，在 `src` 文件夹目录下，每一个题目对应一个文件夹(package)。在package内部，通常题目要求实现若干个类(class)，分别对应一个java源代码文件（未完整实现）。**本次作业不需要使用Test测试类。**

```
1  └─hw1
2      └─README.pdf
3
4      └─src
5          └─drawarc
6              └─DrawArc.java (待实现)
7          └─colorword
8              └─ColorWord.java (待实现)
```

同学们完成作业的建议步骤如下：

1. 按照文档，在代码框架中完成题目所要求的功能类的完整实现。
2. **严格按照以下文件结构和命名提交源代码文件，并打包成 zip 格式上传，文件命名为 学号\_姓名.zip。**

```
1  └─2020123456_张三 (学号_姓名)
2      └─report.pdf/txt/md (实验报告，必选)
3
4      └─src
5          └─drawarc
6              └─DrawArc.java (待实现)
7          └─colorword
8              └─ColorWord.java (待实现)
9      └─output
10         └─drawarc.jar
11         └─colorword.jar
```

请注意：

- 编程作业为个人作业，请独立完成。如果提交的代码参考了资料或个人，请在report中标注来源，否则视为抄袭。
- 编程作业采用自动化评判，因此不按照规定格式提交作业可能导致错误。
- 请严格按照代码框架和文档要求对成员变量、类、函数、文件命名，函数参数类型、名称、顺序和函数返回值类型请严格按照文档实现，此外类、成员变量和函数访问权限请按照文档要求指定。**Java严格区分大小写，不符合要求的命名会导致错误。**
- 请不要更改代码框架的文件结构和命名，请不要删除代码框架中java源文件开头的包声明 `package ...;`。
- 除了题目要求外，可以根据需要实现其他类、函数和成员变量。可以提交其他新增的源代码文件，在新增的源代码文件开头必须进行包声明 `package ...;`。
- **本次实验报告是必选的，包括程序运行的4个截图，截图内容在下面文档中说明，实验报告中每4个截图总共占20分。**实验报告的其他内容可选，为除了源代码外额外需要说明的内容，例如参考的资料，复杂题目的实

现思路等。

- 本次实验要求提交Java ARchive 文件(.jar 文件), 这是一种 Java 平台上的归档文件格式, 用于将多个 Java 类文件 (编译后的 Java 代码) 及相关元数据和资源 (如文本、图片等) 打包到一个文件中。打包生成的 jar 文件可以直接运行 (双击)。在此后编程作业和大作业中, 评测过程的一部分是运行同学提交的 jar 文件并检查结果。本次作业首次涉及 jar 文件, 因此只要求同学提交 jar 文件, 但 jar 文件的运行结果不计入评分, 评测过程会重新编译运行源代码文件, 但强烈建议同学在本次作业中学习 jar 文件的生成和使用方式。若在本次作业中没有提交 Java ARchive 文件, 对应的题目得分折扣 50%。

Jar 文件的简要教学请参考<http://web.eecs.utk.edu/~bvanderz/cs365/notes/jar.html>, 更详尽的说明文档参考 Oracle 官网<https://docs.oracle.com/javase/tutorial/deployment/jar/>。下面在本文档中介绍使用命令行生成 jar 文件的最直接方式。

关于 Java ARchive 文件的生成, 可以在命令行中使用 jar 命令。

```
1 jar -cvfe <output-filename> <entry-point> <files>
```

其中 output-filename 是输出 jar 包的路径, entry-point 是入口类, 即 main 函数所在的类。files 是要打包的文件, 本次作业中需要使用的两个命令如下:

```
1 jar -cvfe drawarc.jar drawarc.DrawArc drawarc\DrawArc.class
2 jar -cvfe colorword.jar colorword.ColorWord colorword\*.class
```

#### 注意事项:

1. 请在编译产生的 class 文件的上级目录下 (一般为 src) 运行上面两个命令。其中, entry-point 不可以使用绝对路径, 必须是 package.Class 的形式; files 不可以使用绝对路径, 必须是相对路径。这是由于 jar 文件本质是一个压缩包, 内部的结构由 files 参数决定: 若 files 参数指定为某绝对路径, 例如 "D:\files\hw\src\drawarc\DrawArc.class", 压缩包中就会相应建立 "files\hw\src\drawarc" 这个目录, 导致程序入口 "drawarc.DrawArc" 无法定位; 正确的结果应当是压缩包内只有 "drawarc" 这个目录。因此, 最简单的方式是使用相对路径指定需要压缩的 files 列表。如果需要使用绝对路径, 请使用 -C 选项, 例如: ==

```
1 jar -cvfe drawarc.jar drawarc.DrawArc -C D:\files\hw\src drawarc\DrawArc.class
```

2. jar 文件内必须包括编译生成的所有目标文件(.class), 因此 files 参数不应当是 java 源代码, 同时如果生成多个目标文件, 必须全部进行打包, 因此本次作业 colorword 的 jar 文件打包命令中, 可以通过 \* 符号匹配生成的所有 .class 文件。
3. 请同学双击运行生成的 jar 文件检查程序结果, 如果无法正常运行, 可以通过以下命令检查 jar 文件内容:

```
1 jar tf <jar-file>
```

4. 如果命令行中运行上述 jar 命令时, 显示“无法找到”、“无法识别”等内容, 可能是环境变量没有正确设置。这个问题在 Windows 系统中较为常见, 请参考<https://www.runoob.com/w3cnote/windows10-java-setup.html> 正确设置环境变量, 然后重启命令行。如果在 Mac 系统中发生上述问题, 建议参照 hw0 重新从官网或 IDE 中安装 JDK。
5. 某些版本的 jdk 无法使用 e 参数, 此时可以使用如下命令:

```
1 jar -cvfm myJar.jar myManifest.txt myApp.class
```

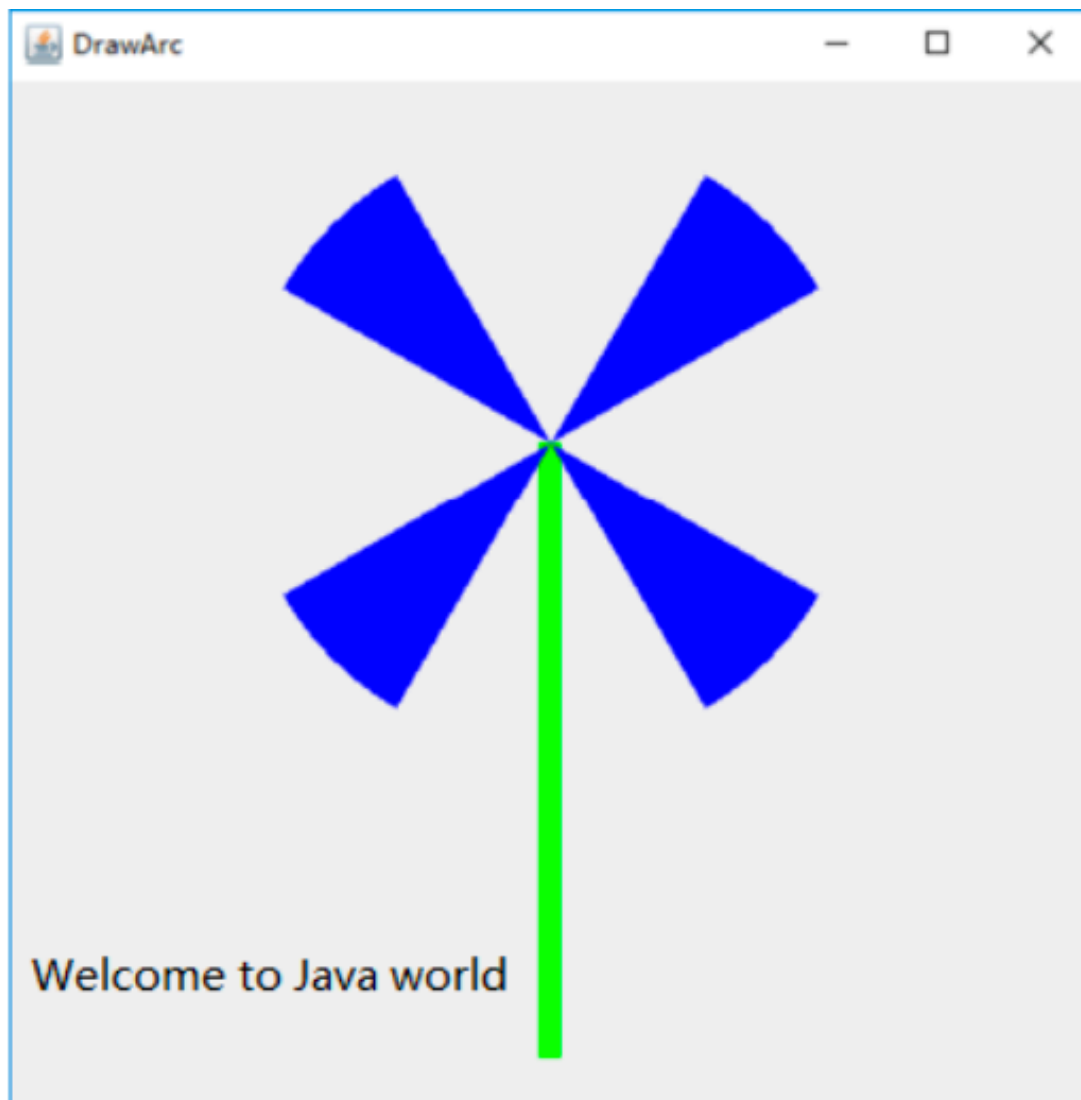
manifest文件参考<https://docs.oracle.com/javase/tutorial/deployment/jar/manifestindex.html>。

6. 使用IDEA的同学可以不使用命令行生成jar文件，请参考 <https://blog.csdn.net/gslsxqrj/article/details/82624157>。使用Eclipse的同学可以不使用命令行生成jar文件，请参考<https://www.jianshu.com/p/28d54b737c18>。请注意打包jar文件时需要选择将依赖同时打包在文件中，这样得到的jar文件又称Runnable Jar或Fat Jar。

- 文档和实验框架的错误、歧义和bug可联系助教[wujy22@mails.tsinghua.edu.cn](mailto:wujy22@mails.tsinghua.edu.cn)。
- Java API官方在线文档：<https://docs.oracle.com/en/java/javase/19/docs/api/>
- 本次作业要求实现的所有类和方法均为公有的。

## Problem 1: DrawArc (50')

在包 `drawarc` 中实现功能类 `DrawArc`，绘制如下图形。



1. 窗口标题为 `DrawArc`，窗口大小为 `600*600`。
2. “Welcome to Java world”的位置、字体、大小自行设置。
3. 风车的半径、风车叶的大小等参数自行设置。

评分标准：

1. 运行结果截图(10")：如上所示。
2. 矩形绘制(10")：宽度小于高度，矩形边横平竖直。
3. 风车叶的扇形绘制(10")：四个扇形，圆心均为矩形上边的中点，四个扇形所在的角度分别为45、135、225、

315度。

4. 颜色绘制(10'')：矩形为绿色，扇形为蓝色。
5. 文字绘制(10'')：内容为“Welcome to Java world”，不能与风车图形重叠。

## Problem 2: ColorWord (50')

在包 `colorword` 中实现功能类 `ColorWord`，实现一个可以给便文字颜色的图形界面程序。



1. 界面上有一个静态文本框，里面有一段文字。
2. 界面上有三个单选按钮，用于设置文字颜色：红色、蓝色、黑色。当用户选定某个单选按钮之后，文字颜色变成选项对应的颜色。

评分标准：

1. 运行结果截图(10'')：共3个截图，如上所示。
2. 绘制按钮(10'')：共3个按钮，对应文字为red, blue, black。按钮可以通过点击选中。
3. 绘制文字(10'')：内容为“Welcome to Java world”，不与按钮重叠。
4. 按钮响应(20'')：按钮点击后文字变为对应颜色。