

# 编译原理实验

20308003 曾伟超

## 运行环境

系统: Debian 11 (bullseye)

Java 环境:

```
1 | openjdk 11.0.18 2023-01-17
2 | OpenJDK Runtime Environment (build 11.0.18+10-post-Debian-1deb11u1)
3 | OpenJDK 64-Bit Server VM (build 11.0.18+10-post-Debian-1deb11u1, mixed mode)
```

Java 编译器:

```
1 | javac 11.0.18
```

## 文件结构

所提交的 `exp0.zip` 解压后应该有着如下的文件的结构

```
1 | exp0
2 |   ├── class
3 |   |   ├── *.class
4 |   ├── design.pdf
5 |   ├── README_zh.pdf
6 |   ├── result
7 |   |   ├── *.log
8 |   ├── scripts
9 |   |   ├── build.sh
10 |   |   ├── clean.sh
11 |   |   ├── run.sh
12 |   |   └── test.sh
13 |   ├── src
14 |   |   ├── argParser.java
15 |   |   ├── shell.java
16 |   |   ├── TaxCalculator.java
17 |   |   └── tax.java
18 |   └── test
19 |       ├── test1.csv
20 |       └── test2.csv
```

其中, `class` 与 `result` 两个文件夹在提交前做了清空, 便于方便在不同的平台重新编译运行

`src` 目录: 包含了本次实验的所有源代码

`scripts` 目录：包含了编译，运行和测试所需要的脚本，在运行前，需要赋予其可执行权限，即 `chmod +x ./scripts/*`，具体运行的时候，先运行 `build.sh` 进行编译，编译出来的 `class` 文件将被放置于 `class` 目录下；`run.sh` 会将传入的参数传递到实际的程序中运行，即 `./run.sh --help` 会将 `--help` 传递到程序中运行；`clean.sh` 则是负责清理已编译的文件，即将 `class` 目录清空；`test.sh` 则是一个进行测试的，会对 `test` 目录下所有的文件当成一个配置文件传递进程序，并随机设置一个免征额和工资，进行测试，测试的结果会写入 `result` 目录下以 `test_{timestamp}.log` 的文件名存储

`test` 目录：每一次测试的征税规则，使用 `csv` 来做管理，具体的，`test1.csv` 如下

```
1 0-3000,3%
2 3000-12000,10%
3 12000-25000,20%
4 25000-35000,25%
5 35000-55000,30%
6 55000-80000,35%
7 others,45%
```

即每行为一个规则，形式如 `范围,税率`，特别的 `others` 在计算时会转换为  $(max, inf)$  其中 `max` 为所有规则的一个最大值，即要求输入的测试为  $(0, max)$ ，对于不满足的，会在实际测试的时候打印出一条 `warning` 的信息

同时我们知道，税率会从 0 开始，通过文件载入的假如说出现了负值，例如 `-1-3000,5%`，这种情况下 `-` 将会被忽略，而假如 `-5--1,5%` 则会在载入阶段直接出错，退出程序

## 运行方式

`cd scripts && ./run.sh --help` 会给出程序运行可供的参数，如下

```
1 TaxCalculator
2 Support Arguments:
3     --interactive : boolean, default=false, whather to enable interactive
   mode
4     --start      : int, At how much should someone start to pay taxes
5     --detail     : boolean, default=true, To display detail info
6     --salary     : double, Salary someone get
7     --config     : string, A CSV file that contain tax rules
```

具体的，设置 `interactive` 为 `true` 则会进入到交互模式，在交互模式下，通过 `help` 指令能看到如下的信息

```

1 TaxCalculator Interactive Mode
2 Supported commands: help, set, load, show, remove, clean, calc, exit
3     help : to display this info
4     set  : to set salary, rules, startpoint, use like:
5           set [range] [percentage], like set 0-3000 3%
6           set salary [val], like set salary 50000
7           set start [startval], like set start 5000
8     load : to load tax rule from csv file, use like load [csvFile], like
9           load test1.csv
10    show : to display all tax rules and set values
11    remove: to remove one rule, like remove 1, it will remove the rule NO.1
12    clean : to remove all rules and other sets
13    calc  : to calc how much tax someone should pay
14    exit  : to exit this program

```

即在交互模式下可以使用上述命令来对程序进行交互，需要注意的是，交互模式下会忽略设置的 `detail` 值，永远会以详细的方式打印，但是对于 `start`，`salary`，`config` 的数值则会尝试进行加载，也就是说

```

1 ./run.sh --interactive true --config ../test/test1.csv --start 5000 --salary
  50000

```

这样运行的程序，等价于

```

1 ./run.sh --interactive true

```

之后再在交互模式中

```

1 set salary 50000
2 set start 5000
3 load ../test/test1.csv

```

而如果不设置交互模式的话，默认则是以非交互模式运行，在这种情况下就必须给出一个配置文件，征税起始额度，当前工资；一个可选项则是是否更加详细的打印；例如，使用 `test/test1.csv`，其内容如下

```

1 0-3000,3%
2 3000-12000,10%
3 12000-25000,20%
4 25000-35000,25%
5 35000-55000,30%
6 55000-80000,35%
7 others,45%

```

假设一个人的工资为 50000，征税起始为 5000，那么

- 通过 `./run.sh --start 5000 --salary 50000 --config ../test/test1.csv --detail false` 运行得到的结果为

```
1 Your salary is: 50000.00, start point is: 5000
2 Base on all the rules provided, he/she needs to pay 9090.00 taxes
```

- 而如果通过 `./run.sh --start 5000 --salary 50000 --config ../test/test1.csv --detail true` 运行, 则

```
1 Your salary is: 50000.00, start point is: 5000
2 Base on rule No.1, he/she needs to pay 90.00 taxes
3 Base on rule No.2, he/she needs to pay 900.00 taxes
4 Base on rule No.3, he/she needs to pay 2600.00 taxes
5 Base on rule No.4, he/she needs to pay 2500.00 taxes
6 Base on rule No.5, he/she needs to pay 3000.00 taxes
7 Base on rule No.6, he/she needs to pay 0.00 taxes
8 Base on rule No.7, he/she needs to pay 0.00 taxes
9 Base on all the rules provided, he/she needs to pay 9090.00 taxes
```