

For each of the following create a new project with an appropriate name and then write a program that solves the given problem. Remember to use Shift+Ctrl+F to format the program, or Shift+Command+F on the Mac. For each program, you will be submitting the java code file through MyClasses, along with either a Microsoft Word docx file, LibreOffice Writer odt, or a text file (which you can create with NotePad++) which contains the output of at least three runs of each program on different data inputs.

## 1 Programming Exercises

1. Write a program that will take as input a decimal number representing the user's yearly taxable income and return the amount of income tax they must pay the government. Income tax is calculated as a percentage of the users income according to what bracket they fall in. Here is the tax scheme. If the person makes less than \$40,000 they pay 15% of their income in tax. If the person makes \$40,000 or more up to but not including \$65,000 they pay \$6,000 plus 20% of their income that exceeds \$40,000. If the person makes \$65,000 or more up to but not including \$100,000 they pay \$11,000 plus 25% of their income that exceeds \$65,000 in tax. If the person makes \$100,000 or more up to but not including \$200,000 they pay \$19,750 plus 27.5% of their income that exceeds \$100,000 in tax. If the person makes \$200,000 or more they pay \$47,250 plus 30% of their income that exceeds \$200,000 in tax. Several sample runs are below.

```
Input your income: 25254
Your tax is: $3788.10
```

```
Input your income: 62980
Your tax is: $10596.00
```

```
Input your income: 100000
Your tax is: $19750.00
```

```
Input your income: 250000
Your tax is: $62250.00
```

2. Write a program that will ask the user to input their name on a single line in informal style (e.g. John Doe), and their year of birth (in yyyy format, such as 1984). The program should calculate their age and then print out the users formal name (e.g. Doe, John) followed by their age. Then if the user's age is less than or equal to 12 print out "You are just a kid.", if the user's age is greater than 12 but less than 20 then print out "You are a teenager.", if the user's age is greater than or equal to 20 but less than 40 then print out "You are getting up there.", and finally if the user's age is greater than or equal to 40 print out "Man, you are old!". A sample run is below.

```
Input Name (informal format): Don Spickler
Input Year of Birth (yyyy): 1965
Spickler, Don
Age: 54
Man, you are old!
```

## 2 Challenge Exercise

Challenge Exercises are optional, they will be graded as extra credit.

Write a program that will solve a quadratic equation given the values of the coefficients. Specifically, the program will solve the equation  $ax^2 + bx + c = 0$  for the given values of  $a$ ,  $b$ , and  $c$ . Recall from algebra that the solutions to  $ax^2 + bx + c = 0$  are given by the quadratic formula  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ . One must be a little careful here, since if  $a = 0$  then you cannot apply this formula because of division by 0. But if  $a = 0$  then the equation is linear, that is,  $bx + c = 0$  which has a solution of  $x = -\frac{c}{b}$ . Then again if  $b = 0$  as well then you cannot do this calculation. In this case, we have the equation  $c = 0$ . Here, if the value of  $c$  is 0 then the equation is valid for all possible values of  $x$  and hence all real numbers are solutions to the equation. Now if  $a \neq 0$  then we could apply the formula  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ , but now we have another problem. If  $b^2 - 4ac < 0$  then the square root function in the Math package will not be able to evaluate  $\sqrt{b^2 - 4ac}$ . To get around that, you will need to check if the value of  $b^2 - 4ac \geq 0$ , if it is then you can apply the quadratic formula and get the two solutions. On the other hand, if  $b^2 - 4ac < 0$  then the two solutions are complex numbers. The solutions are  $x = -\frac{b}{2a} \pm \frac{\sqrt{|b^2 - 4ac|}}{2a}i$ . Your program must work for all values of  $a$ ,  $b$ , and  $c$ , and print out the complex number solutions in the format above using  $i$ . so test your code completely. If you get answers like NaN or Infinity then you missed a case. Before you begin programming, write down all of the possibilities by hand and organize them in the charting software, it will save you time and aggravation. Several test runs are below.

```
Quadratic equation solver for ax^2+bx+c = 0
Input a: 5
Input b: 7
Input c: 1
Solutions are x = -0.16148351928654964 and x = -1.2385164807134506
```

```
Quadratic equation solver for ax^2+bx+c = 0
Input a: 0
Input b: 1
Input c: 2
Solution: x = -2.0
```

```
Quadratic equation solver for ax^2+bx+c = 0
Input a: 0
Input b: 0
Input c: 54
No Solution
```

```
Quadratic equation solver for ax^2+bx+c = 0
Input a: 0
Input b: 0
Input c: 0
All real numbers are solutions.
```

```
Quadratic equation solver for ax^2+bx+c = 0
Input a: 1
Input b: 2
Input c: 3
Solutions are complex.
Solutions are x = -1.0 + 1.4142135623730951i and x = -1.0 - 1.4142135623730951i.
```