Assignment 4

To Be Submitted by Sunday November 29

- a zipped copy of your NetBeans project to the dropbox
  - use File > Export Project > To ZIP
  - the name of this file should be yourName_a4.zip

Programming Style

- this is to be an individual effort
  - do not work with anybody to complete the assignment
- follow the programming style outlined in the text Chapter 1.7
  - use at least two line comments and two block comments
  - use proper indentation and spacing
  - use either End-of-line or Next-line style

Note:

- up to 40% of the mark will be based on the appearance and ease of use of the program
- the program ends with
  - the appropriate calculation
  - your name and student number should always appear as the last line of output
- use debugging (step through code) to make sure that invalid data is not processed
  - for example, if user enters a negative number, that number should not be used

Program Overview

The program is to prompt the user to type in an Worker's number, name, hours worked and hourly wage rate, which is then used to calculate the weekly pay (hours worked * hourly wage rate), income tax paid (base on the table below) and the net pay (weekly pay – income tax paid).

All the information is to be stored and calculated in a Worker object. The program should then use the Worker object to print out the Workers number, name, weekly pay, taxes paid and net pay.

Note:
Worker class should use set and get methods to work with the data.

If you do not use a Worker object to store and print out the required information, then your mark for this program will be zero.

Basic Code Requirements

- your code should contain above the class name for each class

```
/**
Your Name
Your Student Number
Assignment 4
Date
*/
```

- create a project called **PayTime** that has two classes
  - a Main class that contains the main method
  - a Worker Class
- before the program ends
  - the number of Workers processed should be displayed
  - your name and student number should be printed out
- you must use printf for all output
- you can use print or println for the prompting

The Main Class

- prompts the user for an Worker's
  - number
  - first name
  - last name
  - hours worked
  - hourly wage
- then uses methods found in Worker class to calculate
  - pay
  - income tax
- make sure that the numbers entered are reasonable
  - such as not allowing negative numbers for hours worked
  - use your own judgement
- if the user enters an invalid Worker number then a message should appear saying: "Invalid, enter proper Worker number: "
  - use an Arrays class to check the array of Worker numbers stored in Worker class
- then display the Worker's
  - number
  - name
  - weekly pay (hours worked * hourly wage)
  - income taxes (see table below)
  - net pay (weekly pay – income taxes)
- then give the user the opportunity to enter the data of another Worker
- the process should be repeated until the user declines to process another Worker
- the program ends with
  - a print out of the number of Workers processed

- Student name and number

The Worker Class

- contains a private integer array called empNumbers that contains the following valid Worker numbers: 101, 103, 106, 109, 110, 113, 116, 118, 120
- contains a public method that
  - takes an integer parameter
  - returns true if that integer is found in the empNumbers array
- contains a private method to calculate pay
  - regular pay = hoursWorked * hourlyRate
- contains a private method to calculate the income tax based on the following table

Weekly Pay ($)	Income Tax (%) of Regular Pay
0 to 300.00	10
300.01 to 400.00	12
400.01 to 500.00	15
500.01 and over	20

- **Note:**
  - the above methods and array must be used in your Worker class
  - you will need to add your own methods and variables to the Worker class
  - you must use get and set methods to work with the data