Test Driven Development & Refactoring

1. Assume you are doing Test Driven Development. Write a Strong Robust Equivalence Class test for a function that takes an employee's hourly pay and the number of hours they worked in a week and returns their gross pay according to the legal requirement that they must be paid time-and-half for any time over 40 hours and double-time for anything over 60 hours. Assume the hourly pay is valid and does not need to be checked.
2. Write a C++ function to meet the requirement in Problem 1.
3. Explain how your tests from Problem 1 would pass the code you wrote in Problem 2.
4. Refactor the following code by moving the System.out.println lines of code to a separate new function and replace the old code with a call to the function.

```
void printOwing() {
  printBanner(); //prints a standard banner

  // Print details.
  System.out.println("name: " + name);
  System.out.println("amount: " + getOutstanding());
}
```

1. I1 - Invalid, V1 - Regular Time, V2 - Time and a half, V3 - Double Time

Test Case	Hourly Rate	Hours Worked	Expected Output	Class Covered
T1	10	-1	Error	I1
T2	10	0	0	V1
T3	10	40	400	V1
T4	10	41	415	V2
T5	10	60	700	V2
T6	10	61	720	V3

2. .

```cpp
#include <stdexcept>
double computeGrossPay(double rate, double hours){
    if(hoursWorked<0){
        throw std::invalid_argument("hoursWorked must be >= 0");
        }
    double pay = 0.0;
```

```
7
8       double regular = std::min(hours, 40.0); //First 40 hours is regualr pay
9       pay += regular*rate;
10      hours -= ragular;
11
12      double overtime = std::min(hours, 20.0); //Next 20 hours is 1.5 pay
13      pay +=overtime*rate;
14      hours -= overtime;
15
16      //double time for remaining hours
17      if (hours > 0){
18          pay+= hours*rate*2.0;
19      }
20
21      return pay;
22
23  }
```

3. T1(hours = -1) hits the if(hoursWorked<0) check and throws an exception which matches the error for invalid class I1. T2(hours=0) and T3(hours =40) both satsify the regular = 0 and = 40. T4 causes regular =40 and pay =400 + 1 hour at 1.5 + 15 = 415. T5 regular = 40 and overtime =20, so pay = 700 which is expected. T6 regular =40 and overtime =20 + 1 hour at double time = 720, which is expected.
4. .

```
1   void printDetails(){
2       System.out.println("name " + name);
3       System.out.println("amount: " + getOutstanding());
4   }
5
6   void printOwing(){
7       printBanner();
8       printDetails();
9   }
```