LAB 5 – ASSIGNMENT 2

PROBLEM 1

Description:

We need to design a wrapper class named MyInteger and to test a few static methods in the class

- isEven(), isEven(int), isEven(MyInteger)
- isOdd(), isOdd(int), isOdd(MyInteger)
- isPrime(), isPrime(int), isPrime(MyInteger)
- equals(int), equals(MyInteger)
- parseInt(char[]) and parseInt(String)

Analysis:

- I initialized two objects namely n1 and n2 and set respective values to it.
- Write necessary functions for all the methods and invoked at required places.
- No major difficulties faced, except with enhancing the logic for prime number check, required me to brush up some math concepts.

Source Code:

```
package edu.northeastern.csye6200;
public class LAB5P1 {
public static void main(String[] args) {
// TODO: write your code here
MyInteger n1 = new MyInteger(7);
MyInteger n2 = new MyInteger(24);
System.out.printf("n1:%d is even? %s%n", n1.getValue(), n1.isEven());
System.out.printf("n1:%d is prime? %s%n", n1.getValue(), n1.isPrime());
System.out.printf("15 is prime? %s%n", MyInteger.isPrime(15));
System.out.printf("parseInt(char[]) for {'4', '3', '7', '8'} = %d%n",
MyInteger.parseInt(new char[] {'4', '3', '7', '8'}));
System.out.printf("parseInt(String) for "+"4378 ="+" %d%n",
MyInteger.parseInt("4378"));
System.out.printf("n2:%d is odd? %s%n", n2.getValue(), n2.isOdd());
System.out.printf("45 is odd? %s%n", MyInteger.isOdd(45));
System.out.printf("n1:%d is equal to n2:%d? %s%n", n1.getValue(),
n2.getValue(), n1.equals(n2));
System.out.printf("n1:%d is equal to 5? %s%n", n1.getValue(), n1.equals(5));
}
}
class MyInteger {
// TODO: write your code here
private int value;
public int getValue() {
// TODO: write your code here
```

```
return value;
}
public MyInteger(int value) {
// TODO: write your code here
this.value = value;
}
public boolean isPrime() {
// TODO: write your code here
return isPrime(value);
}
public static boolean isPrime(int num) {
// TODO: write your code here
if (num <= 1) {
return false;
}
else if (num == 2) {
return true;
}
else if (num % 2 == 0) {
return false;
}
else {
for (int i = 3; i <= Math.sqrt(num); i += 2) {
if (num % i == 0) {
return false;
}
}
return true;
}
}
public static boolean isPrime(MyInteger o) {
// TODO: write your code here
return o.isPrime();
}
public boolean isEven() {
// TODO: write your code here
return isEven(value);
}
public boolean isOdd() {
// TODO: write your code here
return isOdd(value);
}
public static boolean isEven(int n) {
// TODO: write your code here
return (n%2 == 0);
}
public static boolean isOdd(int n) {
// TODO: write your code here
return (n%2 != 0);
}
public static boolean isEven(MyInteger n) {
// TODO: write your code here
return n.isEven();
}
public static boolean isOdd(MyInteger n) {
// TODO: write your code here
return n.isOdd();
```

```
}
public boolean equals(int anotherNum) {
// TODO: write your code here
if(anotherNum == this.value)
return true;
return false;
}
public boolean equals(MyInteger o) {
if(this.value == o.value)
return true;
return false;
}
public static int parseInt(char[] numbers) {
// numbers consists of digit characters.
// For example, if numbers is {'1', '2', '5'}, the return value
// should be 125. Please note that
// numbers[0] is '1'
// numbers[1] is '2'
// numbers[2] is '5'
// TODO: write your code here
int result = 0;
for (int i = 0; i < numbers.length; i++) {
int digitValue = Character.getNumericValue(numbers[i]);
result = result * 10 + digitValue;
}
return result;
}
public static int parseInt(String s) {
// s consists of digit characters.
// For example, if s is "125", the return value
// should be 125.
// TODO: write your code here
int result = 0;
for (int i = 0; i < s.length(); i++) {
int digitValue = Character.getNumericValue(s.charAt(i));
result = result * 10 + digitValue;
}
return result;
}
}
```

Output:

```
n1:7 is even? false
n1:7 is prime? true
15 is prime? false
parseInt(char[]) for {'4', '3', '7', '8'} = 4378
parseInt(String) for 4378 = 4378
n2:24 is odd? false
45 is odd? true
n1:7 is equal to n2:24? false
n1:7 is equal to 5? false
```

PROBLEM 2

Description:

Required to find the number of people in the rooms created inside a building using a class named RoomPeople with two attributes namely, numberInRoom and totalNumber.
We need to utilize the following methods to find the total number of people in the rooms.

- addOnetoRoom
- removeOneFromRoom
- getNumber
- getTotal

Analysis:

To begin, we need to initialize the two objects roomA and roomB of type RoomPeople class.
We need to define the required methods and initialize the corresponding variables to generate output as expected.
Increment and decrement the attribute values as per the process.

Source code:

```
package edu.northeastern.csye6200;
public class LAB5P2 {
public static void main(String[] args){
// TODO: write your code here
RoomPeople roomA = new RoomPeople();
RoomPeople roomB = new RoomPeople();
System.out.println("Add two to room a and three to room b.");
roomA.addOneToRoom();
roomA.addOneToRoom();
roomB.addOneToRoom();
roomB.addOneToRoom();
roomB.addOneToRoom();
System.out.println("Room a holds " + roomA.getNumber());
System.out.println("Room b holds " + roomB.getNumber());
System.out.println("Total in all rooms is " + RoomPeople.getTotal());
System.out.println("Remove two from both rooms.");
roomA.removeOneFromRoom();
roomA.removeOneFromRoom();
roomB.removeOneFromRoom();
roomB.removeOneFromRoom();
System.out.println("Room a holds " + roomA.getNumber());
System.out.println("Room b holds " + roomB.getNumber());
System.out.println("Total in all rooms is " + RoomPeople.getTotal());
System.out.println("Remove two from room a (should not change the values)");
roomA.removeOneFromRoom();
roomA.removeOneFromRoom();
System.out.println("Room a holds " + roomA.getNumber());
System.out.println("Room b holds " + roomB.getNumber());
System.out.println("Total in all rooms is " + RoomPeople.getTotal());
}
}
class RoomPeople {
```

```
// TODO: write your code here
private static int totalNumber = 0;
private int numberInRoom;
public static int getTotal(){
// TODO: write your code here
return totalNumber;
}
public RoomPeople() {
// TODO: write your code here
numberInRoom = 0;
}
public void addOneToRoom(){
// TODO: write your code here
numberInRoom++;
totalNumber++;
}
public void removeOneFromRoom(){
// TODO: write your code here
if(numberInRoom>0) {
numberInRoom--;
totalNumber--;
}
}
public int getNumber(){
// TODO: write your code here
return numberInRoom;
}
}
```

Output:

```
n1:7 is even? false
n1:7 is prime? true
15 is prime? false
parseInt(char[]) for {'4', '3', '7', '8'} = 4378
parseInt(String) for 4378 = 4378
n2:24 is odd? false
45 is odd? true
n1:7 is equal to n2:24? false
n1:7 is equal to 5? false
```