Assignment 1 on Session1: Introduction to JAVA and DATA TYPES

Problem Statement1:

Write a program to print the sum of two numbers without using + operator. Other operators can be used.

Solution:

```java
import java.util.Scanner; //To import utility files for Scanner

class AddTwoInt
{
 public static void main(String a[])
 {
   Scanner sc=new Scanner(System.in); //To get keyboard input
   System.out.println("Enter only integer values");
   System.out.println("Enter First number");
   int num1=sc.nextInt(); //To assign first keyboard input
   System.out.println("Enter Second number");
   int num2=sc.nextInt(); //To assign second keyboard input
   while(num2 != 0)
   {
   int temp=num1&num2; //Binary and operation is applied
   num1=num1^num2; //XOR is applied
   num2=temp<<1; //Binary shift is applied
   }
   System.out.println("Added value is : "+num1);
   System.out.println("This is the end of the program");
 }
}
```

Explanation of the code:

This program gets two integer values from the user. The Binary operators are used in this program to add two integer values. The flow is explained below;

num1 = 11, num2 = 12; so the expected result here is 23.

When checking the condition num2 not equal to 0, the flow enters into the while loop. Then the temp is calculated as temp = 1011 (binary for 11) & 1100 (binary for 12). Hence temp = 1000 (decimal value of 8). num1 = 1011 ^ 1100 ->111 = 7. num2 = 1000<<1 -> 10000 = 16.

So after the first iteration the values are;

temp = 1000 (decimal value 8)

num1 = 111 (decimal value 7)

num2 = 10000 (decimal value 16)

In the second iteration num2 not equal to 0, hence the values are
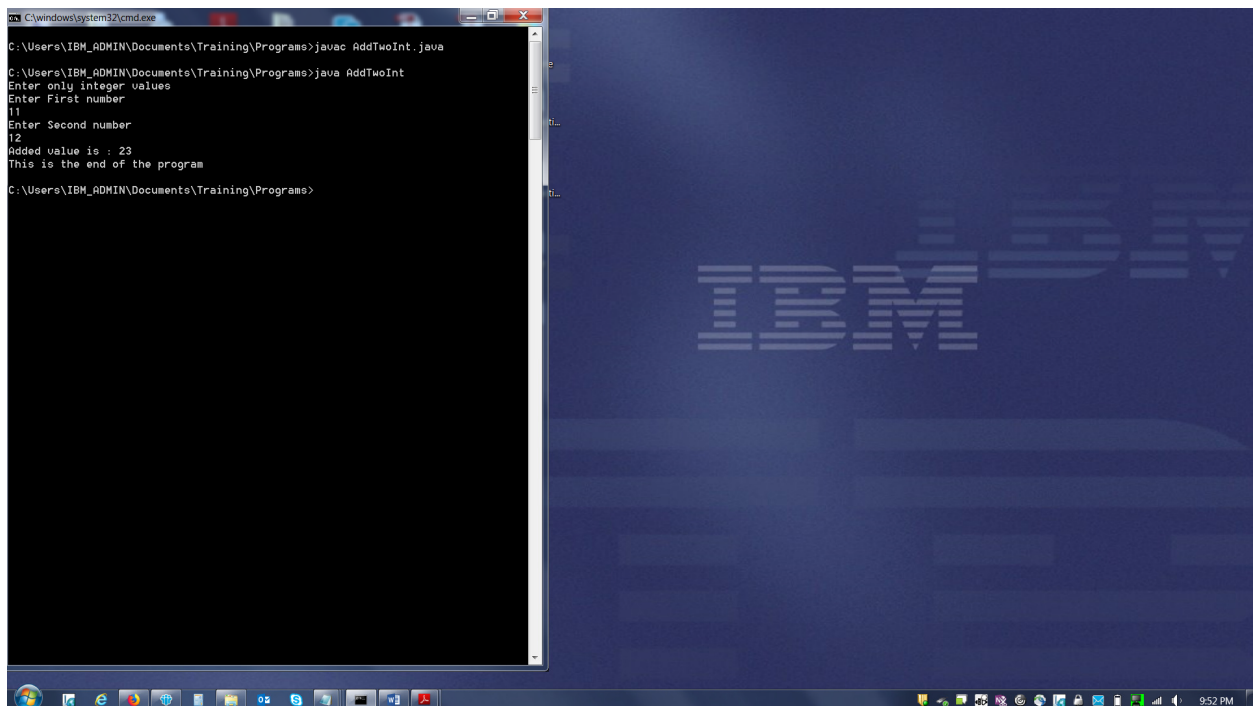
temp = 111 & 10000 = 0000

num1 = 111^10000 = 10111 (decimal value 23)

num2 = 0<<1 = 0

In the third iteration num2 is equal to 0, hence the iteration ends.

The final value is stored in num1, that's 23.

Result flow & Screen shot:

Problem Statement2:

Write a program to print the result of the following expressions provided the integer variable a is 20 and b is 10.

Int b=a-- - --a;

Int c=a--;

Int d=a>>2;

Int e=a&b;

Solution:

```
class PrgForExp
{
  public static void main(String x[])
  {
  int a=20, b=10;
  System.out.println("Value of a is "+a+" and Value of b is : "+b);
  b=a-- - --a;
  System.out.println("a-- - --a value calculated and stored in b, and the value of b is : "+b);
  int c=a--;
  System.out.println("a-- assigned to c and the value of c is : "+c);
  int d=a>>2;
  System.out.println("a>>2 value calculated and stored in d, and the value of d is : "+d);
  int e=a&b;
  System.out.println("a&b value calculated and stored in e, and the value of e is : "+e);
  System.out.println("This is the end of the program");
  }
}
```

Explanation of the code:

Two integer values a = 20 and b = 10 is used in this program to calculate the below;
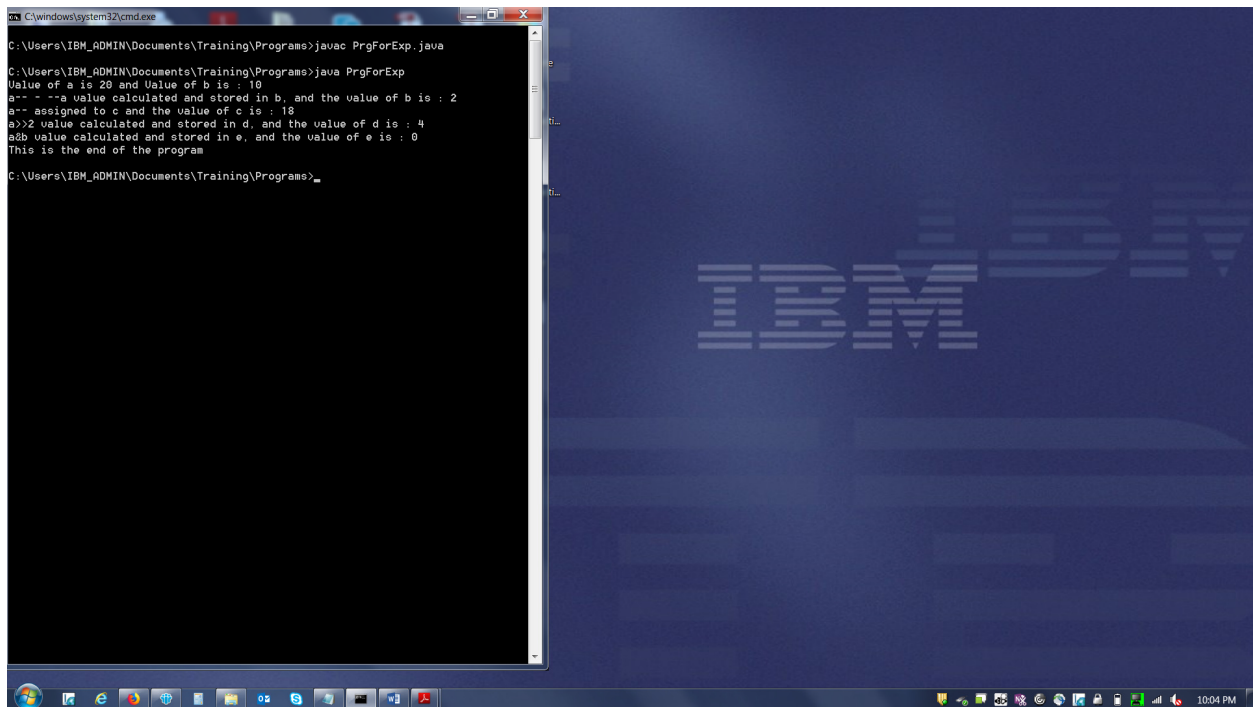
b=a-- - --a. in the expression 20 is first assigned b and then it get decreased by 1 as it's a--. Now the value of a is 19. Then as per –a the value 19 get decreased by 1, and the value of a is now 18. Finally the calculation would be 20-18 = 2. So the value of b is 2.

c=a--. in this expression the value 18 first assigned to c and then get decreased by 1, hence the final value of a is now 17. The value of c would be 18

d=a>>2. Here the binary operator is used. The value stored in a is 17 and the binary value of 17 is 10001. When 00 added in front 00100. This is equal to the decimal value of 4. Hence the value of d is 4.

e=a&b. here the binary AND is used. The value of a 17 and its binary value is 10001. Value of b is 2 and its binary value is 1010. When binary AND is applied the value would be 0000. This the decimal value of 0.

Result flow and Screen shot: