**Follow the Methods - Test Review**

**Problem # 1**

System.***out***.println(*adapt*(23));

**Problem # 2**

**int** num2 = *manipulate*(7);

System.***out***.println(*modify*(num2));

**Problem # 3**

**int** num3 = *produce*(*change*(9));

System.***out***.println((*alter*(num3)));

**Problem # 4**

System.***out***.println(*manipulate*(*produce*(*change*(10))));

**Problem # 5**

**if**(*check*(*produce*(5),*adapt*(6))) System.***out***.println(*change*(4));

**else** System.***out***.println(*alter*(2));

**Problem # 6**

**if**(*flip*(*check*(4, 3))) System.***out***.println(*change*(7));

**else** System.***out***.println(*modify*(12));

**Problem # 7**

**int** num7 = *manipulate*(*alter*(12 - *produce*(15)));

**if**(*examine*(*alter*(8)) || *validate*(1, *check*(4, 5) )) num7 = *modify*(num7);

**else** **if** (*flip*(*inspect*(num7, *verify*(num7), *examine*(num7 + 2)))) num7 %=(*produce*(num7));

System.***out***.println(num7);

**Methods**

**public** **static** **int** change(**int** x)

{

 x/=2;

 **return** (**int**) (Math.*pow*(x, 3));

}

**public** **static** **int** produce(**int** x)

{

 **return** x % 11;

}

**public** **static** **int** modify(**int** x)

{

 **return** 10 - x;

}

**public** **static** **int** manipulate(**int** x)

{

 **while**(x <25)

 x +=3;

 **return** x;

}

**public** **static** **int** adapt(**int** x )

{

 **while**(x > 5)

 {

 x++;

 x/=2;

 }

 **return** x;

}

 **public** **static** **int** alter(**int** num)

{

 **if**(num <= 5) **return** num;

 **return** num -3;

}

**public** **static** **boolean** verify(**int** x)

{

 **if** (x % 2 ==0) **return** **true**;

 **return** **false**;

}

**public** **static** **boolean** flip(**boolean** test)

{

 **return** !test;

}

**public** **static** **boolean** examine(**int** x)

{

 **if**(x \* x \* x < (**int**)(Math.*pow*(2, x))) **return** **true**;

 **return** **false**;

}

**public** **static** **boolean** inspect(**int** x, **boolean** a, **boolean** b)

{

 **return** x > 10 || a && b;

}

**public** **static** **boolean** check(**int** x, **int** y)

{

 **return** x ==y;

}

**public** **static** **boolean** validate(**int** x, **boolean** test)

{

 **if**(test && x > 0) **return** **true**;

 **else** **if** (!test && x < 0) **return** **true**;

 **else** **return** **false**;

}