**Question 01**

A) avg = (double) tot/10;

**Question 02**

B) 4.0 4.5 4.5 5.0

**Question 03**

A : 3

B : -3

C : 3

D : -3

E : 3

F : -3

G : 3

Only sign of Dividend number is considered on modular operator.

**Question 04**

A) 17

B) -10

C) -17

D) -3

E) 7

F) -3

**Question 06**

100101

104

104

105

\* x++ excecated after the SOP command and SOP command get the previous value of the x.

\* But when it comes to ++x, its excecated along with the SOP command.

**Question 07**

101 100

102 101

103 102

y=x++; This statement assigns the current value of x to y and then increments the value of x. So, after this line, y will have the current value of x before the increment.

**Question 08**

101 101

102 102

103 103

y=++x; This statement increments the value of x and then assigns the result to y. So, after this line, both x and y will have the incremented value of x.

**Question 09**

100

100

100

101

102

103

**Question 10**

A) 30 : Basic addition operation with default (+) sign for both.

B) -10 : Equipped with (-) sign for ‘b’ and ‘b’ become -20. so arithmetic operation is 10 – 20 = -10

C) 31 : pre-increment operation for ‘a’ (++a) and a value become 11 and operation is 11+20 = 31

D) 30 : no change for ‘a’. ‘b’ has post increment (b++).In the program execution considered only b`s current value when it is post increment which is 20. So calculation is 10+20 = 30

E) 31 : a has pre-increment (++a) and its current value goes to 11 b has post increment and program consider only previous value which is 20 so calculation is 11+20 = 31

F) 30 : Both has post increment and program consider current value so calculation is 10+20. If we check the values of a and b with SOP after this operation it will indicate that a=11 and b=21

G) 32: Both has pre increment. So both values increment by 1 before calculation. So calculation is 11+21=32

F) 31 : a has post increment and its only consider previous value before calculation. b has pre increment and its value became 21 before calculation. so calculation is 10 + 21 = 31

**Question 11**

12 - 4 \* 2 : 4

(12 - 4) \* 2 : 16

12 - (4 \* 2) : 4

Java Order of operation as following

1. Parentheses()
2. Prefix (unary) ++x, - - x, -x
3. Postfix x++ , x - -
4. Multiplicative \*, / ,%
5. Additive + -
6. Assignment = += -= \*= /= %=

x= 12 - 4 \* 2 : In this statement first operation is 4\*2 and second operation is 12 -8. So answer is 4.

x= (12 - 4) \* 2 : In this statement first operation is inside Parentheses 12-4 and second operation is Multiplicative 8\*2. So answer is 16

x= 12 - (4 \* 2) : In this statement first operation is Parentheses -8 and second operation is Additive 12 – 8 So answer is 4

**Question 12**