ASSIGNMENT-3 SOLUTION

1. Which of the following statement is correct?
   a) Operator precedence determines which operator is performed first in an expression with more than one operator with different precedence. Associativity is used when two operators of same precedence appear in an expression
   b) Operator associativity determines which operator is performed first in an expression with more than one operator with different associativity. Precedence is used when two operators of same precedence appear in an expression
   c) Operator precedence and associativity are same.
   d) None of the above

Solution: (a) Operator precedence determines which operator is performed first in an expression with more than one operator with different precedence, whereas associativity is used when two operators of same precedence appear in an expression

2. The precedence of arithmetic operators is (from highest to lowest)
   a) %, *, /, +, –
   b) %, +, /, *, –
   c) +, -, %, *, /
   d) %, +, -, *, /

   Solution: (a) The precedence order follows the first option (a)

3. What is the output of the following program?
```
#include<stdio.h>
int main()
{
        int x=11, y=5, z;
        float w;
        z=x%y;
        w=x/y;
        printf("Value of z and w are %d and %f respectively",z,w);
        return 0;
}
```
   a) Value of z and w are 1 and 2 respectively
   b) Value of z and w are 1 and 2.200000 respectively
   c) Value of z and w are 1.000000 and 2.200000 respectively
   d) Value of z and w are 1 and 2.000000 respectively

Solution: (d) Modulo (%) gives the reminder and division (/) gives the quotient of a division operation. Here, w is a floating variable. Thus z and w would be 1 and 2.000000 respectively.

4. What will be the output?
```
#include <stdio.h>
int main ()
{
  int a = 4, b = 15, c = 29;
  if (c > b > a)
        printf("TRUE");
  else
        printf("FALSE");
```

```
    return 0;
}
```

a) TRUE
b) FALSE
c) Syntax Error
d) Compilation Error

Solution: (b) FALSE: (c > b > a) is treated as ((c > b) > a), associativity of '>' is left to right. Therefore, the value becomes ((4 > 15) > 29) which becomes (1 > 29) thus FALSE

5. What will be the output?

```
#include<stdio.h>
int main()
{
  int x;
  x= 9<5+3 && 7;
  printf("%d", x);
  return 0;
}
```

a) 0
b) 1
c) 7
d) Compilation error

Solution: (a) 0

This expression is equivalent to:

((9< (5 + 3)) && 7)
i.e., (5 + 3) executes first resulting into 8
then, first part of the expression (9 < 8) executes resulting into 0 (FALSE)
Then, (0 && 7) executes resulting into 0 (FALSE)

6. What will be output?

```
#include<stdio.h>
int main()
{
        int i = 12;
        if (i == 10)
                printf("i is 10");
        else if (i == 15)
                        printf("i is 15");
        else if (i == 20)
                printf("i is 20");
        else
                        printf("i is not present");
                return 0;
}
```

a) i is 10
b) i is 15
c) i is 20
d) i is not present

Solution: (d) i is not present
None of the conditionals are true hence final else is executed

7. What will be the output?

```
#include <stdio.h>
int main()
{
   int x=0;
   if(x=0)
      printf("TRUE");
   else
      printf("FALSE");
   return 0;
}
```

a) TRUE
b) FALSE
c) Compilation Error
d) Compiler Dependent

Solution: (b) FALSE

if(x=0)... "=" is an assignment operator, so 0 will be assigned to x and condition will be false due to if(0).

8. What will be the output?

```
#include <stdio.h>
int main()
{
   if((-10 || 10) && (3>4))
      printf("Condition is true.");
   else
      printf("Condition is false.");
   return 0;
}
```

a) Condition is true
b) Condition is false
c) Error
d) No output possible

Solution: (b) Condition is false

Any non-zero value is treated as true for condition. Consider the expressions: if((-10 || 10) && (3>4))
=if((1) && (0))
=if(0)

9. What will be the output?

```
#include <stdio.h>
int main()
{
  int x= 17, y= 1;
  if (!(!x) && y)
    printf("%d", x);
  else
    printf("%d", y);
  return 0;
}
```

a) 17
b) 18
c) 1
d) 0

Solution: (a)

!(!x) = x i.e. x and y both are non-Zero value and equal to 1 (True). Therefore, the statement is True.

10. What is the output of the following C code?

```
#include <stdio.h>
int main()
{
  int x=7, y=3, z=5;
  printf("%d\n", x-x/y*y%z);
  return 0;
}
```

a) 7
b) 6
c) 5
d) 0

Solution: (b)
x-x/y*y%z here the operators /, * and % have the same precedence. So they will be calculated from left to right.
Therefore, x – x/y*y%z = 7 – 7 / 3 * 3 % 5 = 7 – 2 * 3 % 5 (integer part of 7/3 = 2) = 7 – 6 % 5 = 7 – 1 = 6

11. What will be the output?

```
#include <stdio.h>
int main()
{
  int x;
  x=(11+3)%5/2;
  printf("%d\n", x);
  return 0;
}
```

a) 0
b) 1
c) 2
d) Compilation error

Solution: (c)
x = (11+3)%5/2 = 14%5/2 (As (operator have higher precedence) = 4/2 (As % and / has same precedence and will be operator from left to right) = 2 (as only x is an integer)

12. Which of the following method are accepted for assignment in C?
    a) 8=x=y=z
    b) x=8=y=z
    c) x=y=z=8
    d) None

Solution: (c)

13. Which of the following operator has the highest precedence?
    a) !
    b) %
    c) +
    d) >

Solution: (a) ! (NOT) has the highest precedence

14. Find the output of the following C code

```
#include<stdio.h>
int main()
{
   int a=10, b=3, c=2, d=4, result;
   result=a+a*-b/c%d+c*d;
   printf("%d",result);
  return 0;
}
```

a) -42
b) 24
c) 15
d) -34

Solution: (c) Following the precedence rule, we can conclude that the operation steps are

- ➔ Result=10+10*- 3/2%4+2*4
- ➔ Result=10-30/2%4+2*4
- ➔ Result=10-15%4+2*4
- ➔ Result=10-3+2*4
- ➔ Result=10-3+8
- ➔ Result=7+8
- ➔ Result=15

15. The output of the following program will be

```
#include<stdio.h>
int main()
{
        int a=0,b=1,c=-1;
        if(a)
        printf("IITKGP \n");
        if(b)
        printf("IITM \n");
        if(c)
        printf("IITR \n");
return 0;
}
```

a) IITKGP

b) IITM IITR

c) IITM
   IITR

d) IITKGP
   15

Solution: (c) +1 and -1 is taken as true inside the if statement. Whereas 0 will be considered as false. Thus the printf corresponding to b and c will be printed.