# TUTORIAL 2

Objective:-

- To make the students learn how to solve a problem using a structured programming language.
- To strengthen the basic programming skills in order to prepare them for advanced programming concepts.

Outcome:-

CO1) Students will be able to apply basic C programming constructs to solve problems

# QUESTION 1: HOW MANY TIMES “HELLO” WILL BE PRINTED?

```
#include <stdio.h>
void main()
{
int i;
for (i = 1024; i; i >>= 1)
    printf(“Hello”);
}
```

Answer:
11 times

# QUESTION 2: FIND THE OUTPUT

```
int i = 1;
while(i<=5)
{
if(i>=3)
continue;
printf("Hello ");
i++;
}
```

Answer:

Hello Hello Infinite Loop

# QUESTION 3: FIND THE OUTPUT WHEN i=0 AND i=1

```
int i ;
switch (i)
{
    case '0': printf(“Hello");
            break;
    case '1': printf(“Welcome");
            break;
    default: printf(“Bye");
}
```

Answer:

Bye, Bye

# QUESTION 4: FIND THE OUTPUT

```
#include<stdio.h>
void main()
{
   int n;
   for (n = 9; n!=0; n--)
      printf("n = %d", n--);
}
```

Answer:

Infinite Loop

# QUESTION 5: FIND THE OUTPUT

```
#include <stdio.h>
int main()
{
    int c = 5, no = 90;
    do {
        no /= c;
    printf ("%d  ", no);
        c--;
    } while(c!=1);
}
```

Answer:

18  4  1  0

# QUESTION 6: FIND THE OUTPUT

```
for (int i=0; i<20; i++)
{
  switch(i)
  {
    case 0:
      i += 5;
    case 1:
      i += 4;
    case 5:
      i += 3;
      break;
    default:
      i += 6;
  }
  printf("%d  ", i);
}
```

Answer:

12  19

# QUESTION 7: FIND THE OUTPUT

```
int x=0;
do
{
if(x==3)
break;
printf("%d ",++x);
} while(1);
```

Answer:

1 2 3

# QUESTION 8: HOW MANY “X” ARE PRINTED?

```
for(i=0,j=10; i < j ; i++,j--)
printf("x");
```

Answer:
5

# QUESTION 9: FIND THE OUTPUT

```
int a=2, b=1, c=2;
switch(a)
{
case b: printf("You are in b ");
        break;
case c: printf("You are in c ");
        break;
default: printf("You are in default");
}
```

Answer:
Compilation Error. Variables cannot be given as case values

# QUESTION 10 : HOW MANY TIME HELLO IS PRINTED?

```
int i = - 5;
 while (i <= 5)
{
    if (i >= 0)
        break;
    else
    {   i++;
        continue;
    }
    printf(“Hello");
}
```

Answer:

0 times

# QUESTION 11 : FIND THE OUTPUT

```
int x = 2;
if (x == 2) x++;
if (x == 3) x++;
if (x<5) x*=2;
else x += 2;
printf("x = %d", x);
```

Answer:

x = 8

# QUESTION 12 : FIND THE OUTPUT

```
int x = 2;
if (x == 2) x++;
else if (x == 3) x++;
else if (x<5) x*=2;
else x += 2;
printf("x = %d", x);
```

Answer:

x = 3

# QUESTION 13 : FIND THE OUTPUT

```
int i;
goto LOOP;
for (i = 0 ; i < 10 ; i++)
{
   LOOP:
   printf("Hello ");
   break;
}
  printf("i= %d",i);
```

Answer:

Hello i=Garbage_Value

# QUESTION 14 : FIND THE OUTPUT

```
for (i=0; i<10; ++i)
printf("%d", i&1);
```

Answer:
0101010101

# PROGRAM IT!!!

1. Starting from 100, print the first 20 numbers whose sum of digits is equal to 12.
2. Input a number and print whether it has an odd digit or not. Print “Yes” or “No”.

   Example:

   468 : NO

   579: YES
3. Print the pattern

   1

   10

   101

   1010

   10101

# PROGRAM IT

4. Check if a number is a palindrome/not.
   - Palindromes are numbers whose reverse will also be equal to the same number:
   - Example: 121, 13431, 515 etc…
5. Check if a number is an Armstrong number/not.
   - Armstrong number is the one where,

**$\Sigma$ (digits$^p$) = number itself,**

**where p is the number of digits in the number**

   - Example:
   - $1634 = 1^4 + 6^4 + 3^4 + 4^4$
   - $153 = 1^3 + 5^3 + 3^3$