# C Language – Arrays and Pointers

ZenIndo Solutions Pvt Ltd
T-Hub, Hyderabad
EMail : info@zenindo.com

# Arrays

```
#include <stdio.h>
// sample program : array
main()
{
  int x[] = { 10,20,30,40,50,60,70,80,90,100  } ;

  int i ;
  for (i = 0; i< 10; i++)
     printf("x[%d] value is %d \n " , i, x[i]   );
 //------------

  int y[10] = {1,2,3,4,5,6,7,8,9,10 } ;
  for (i = 0; i< 10; i++)
     printf("Y[%d] value is %d \n " , i, y[i]   );
//------------
  int z[3];
  z[0] = 1;
  z[1] = 2;
  z[2] = 3;

  for (i = 0; i< 3; i++)
     printf("z[%d] value is %d \n " , i, z[i]   );
}
```

```
 #include <stdio.h>
// sample program : Passing single element to
function

main()
{
  int x[10] = { 10,20, 30,40, 50,60, 70,80, 90,100  } ;

  printf(" Element Value before calling function %d
\n", x[2] );
  printElement( x[2] ) ;
  printf(" Element Value after calling function %d \n",
x[2] );



}
printElement(int mElement)
{
      printf("Element value in function  %d \n" ,
mElement);
      mElement = mElement+100;
      printf("Element value in function - after
increasing by 100 %d \n" , mElement);

}
```

Arrays –
passing single value

# Multi Dimensional Arrays

```
 #include <stdio.h>
// sample program : 2 dimensional array
main()
{
  int x[5][2] = { { 10,20}, {30,40}, {50,60}, {70,80},
{90,100}  } ;

  int i, j ;
  for (i = 0; i< 5; i++)
  { for (j = 0; j <2; j++)
        { printf("x[%d][%d] value is %d \t" , i,j, x[i]
[j]   );
        }
         printf("\n");

   }
    printf("end \n");

}
```

# Return Array from function

```
 #include <stdio.h>
// sample program : return array from function
float * getAvgSum(int *myarray, int msize)
{
  static float avgSum[2];

  int mtotal = 0;
  int i;

  for(i=0; i < msize ; i++ )
        mtotal = mtotal + myarray[i] ;

  avgSum[0] = mtotal;
  avgSum[1] = mtotal / msize;

  return avgSum ;
}

main()
{
  int x[10] = { 10,20, 30,40, 50,60, 70,80, 90,100  } ;
  float *arrayPointer ;

  arrayPointer = getAvgSum( x, 10 ) ;

  printf(" sum  = %f, avg = %f ", arrayPointer[0], arrayPointer[1] );


}
```

# Assignment

1. Define array with 10 elements, print array in reverse order ( write in the main program, do not use functions)
2. Define array, with 8 integers, pass 2nd, 4th and 5th elements to a function and print values in function
3. Write a function to receive 5 integer values in an array (use array pointer), increase each value by 100. Print array from main function. Observe if there is any change in the values.
4. Write a function to return an array of 5 prime numbers. You can hard code the 5 prime numbers.