# Pointer & Functions
# Pointers & Arrays

Example Programs

Dr.P.Keerthika

Associate Professor/SCOPE

VIT, Vellore

# Pointer to functions

```
void fun(int a)
{
   printf("Value of a is %d\n", a);
}

int main()
{
      void (*fun_ptr)(int);   //Declaring a ptr to a function
      fun_ptr = &fun;

   (*fun_ptr)(10);  //fn.call

   return 0;
}
```

# Function Returning a 1D array

```
int* fn_arrincrement(int *,int );
#include <stdio.h>
void main()
{
   int *a;
   int *res,i;
   for(i=0;i<5;i++)
   {
      scanf("%d",&a[i]);
   }

   res=fn_arrincrement(a,5);

   for(i=0;i<5;i++)
   {
      printf("%d",res[i]);
   }
}
```

```
int* fn_arrincrement(int *ar,int n)
{
   int i;
   for(i=0;i<n;i++)
   {
      ar[i]=ar[i]+1;
      //printf("%d",ar[i]);
   }
   return ar;

}
```

# Pointer to the first element of the array Pointer to an Array

```
#include<stdio.h>

int main()
{
  int arr[5] = { 1, 2, 3, 4, 5 };
  int *ptr = arr;
    int i;
  printf("Address %d\n", ptr);

  for(i=0;i<5;i++)
    {
        printf("\nValues %d",*ptr);
        ptr++;
    }
  return 0;
}
```

# Array of Pointers

```
#include <stdio.h>
//Array of pointers
// Syntax: data_type (*var_name)[size_of_array];
// Example: int (*ptr)[10];
int main()
{
   // declaring some temp variables
   int var1 = 10;
   int var2 = 20;
   int var3 = 30;
   // array of pointers to integers
   int* ptr_arr[3] = { &var1, &var2, &var3 };
   // traversing using loop
   for (int i = 0; i < 3; i++) {
      printf("Value of var%d: %d\tAddress: %p\n", i + 1,
*ptr_arr[i], ptr_arr[i]);
   }
   return 0;
}
```

```
//Array of pointers to strings

#include <stdio.h>

int main()
{

   char* arr[3] = { "hai", "hello", "welcome" };

   for (int i = 0; i < 3; i++) {
      printf("%s\n", arr[i]);
   }

   return 0;
}
```