Arrays and Pointers- II

Department of CSE
Amrita Vishwa VidyaPeetham

Predict the output

```
#include<stdio.h>
  int main()
{
    int a[] = { 1, 2, 3, 4, 5} ; //initialise the array with 5 elements
    int *ptr;        //Declare an integer pointer ptr
    ptr = a;       //Assign the base address of array a[] to ptr. Now ptr points to a
    printf(" %d ", *( ptr + 1) ); //What is the output?
    return 0;
}
```

Output : 2 **Here since ptr is integer pointer , ptr+1 will increment the address by 4 bytes and ptr+1 has the address of second element in the array. *(ptr+1) dereferences that ptr, ie access the content of the address and print 2**

Predict the output

```
int main()
{
    int A[5]={15,20,30,55,70};
    int *ptr=A;     //ptr contains the base address of array A
    while(*ptr< 30)
    { if(*ptr%2!=0)
            *ptr=*ptr+2;
      else
            *ptr=*ptr+1;
      ptr=ptr+1;
    }
    for(int i=0;i<5;i++)
    { printf("%d ",A[i]);   //Predict the Output
        }
    return 0;
}
```

Output: 17 21 30 55 70

Predict the Output

```
int main()
 { int A[5]={1,2,3,4,5}; //Declare and initialise Array
   int *ptr=A+2;         // Assign ptr
   for(int i=0;i<5;i++)
         { (*ptr)*=2;    //Computation (*ptr) =(*ptr)*2
           --ptr;
         }
    for(int i=0;i<5;i++)
     printf("%d ",A[i]);

}
```

Output: 2 4 6 4 5

Just for Fun!

```
#include<stdio.h>
int main(){
        int a = 36;
        int *ptr;
        ptr = &a;
        printf("%u %u", *&ptr , &*ptr);
        return 0;
}
```

Output: * and & cancel each other and display the address of a two time

Namah Shivaya!