CHAPTER 7

POINTERS

1.Can you combine the following two statements into one

```
char *p;
p=malloc(100);
```

Ans: char *p=malloc(100);

2.Can you split the following statement into two statements?

```
Char far *scr=(char far*) 0xb8000000L;
```

Ans:

```
Char far *scr;
Scr=(char far*) 0xb8000000L;
```

3.Are the expressions *ptr++ **and ++***ptr same?

Ans: No. ***ptr++ increments the pointer and not the value pointed by it, whereas ++***ptr increments the value being pointed to by ptr.

4. Can You write another expression which does the same job as ++*ptr?

Ans: (*ptr)++

5.What would be the equivalent pointer expression for referring the same element as
a[i][j][k][l]?

Ans: *(*(*(*(a+i)+j)+k)+l)

6.O/p?

```
main()
{
  int arr[]=(12,13,14,15,16};
  printf("\n%d %d %d",sizeof(arr),sizeof(*arr),sizeof(arr[0]));
}
```

Ans: 10 2 2

7.What would be the O/P of the program assuming that the array begins at 1002?

```
Main()
{
  int a[3][4]={
                        1,2,3,4,
                        5,6,7,8,
                        9,10,11,12
                    };
     printf("\n%u %u %u",a[0]+1,*(a[0]+1),*(*(a+0)+1));
}
```

Ans: 1004 2 2

8.What would be the output of the program assuming that the array begins at location 1002?

```
Main()
{
  int a[2][3][4]={
                           {
                               1,2,3,4,
                               5,6,7,8,
                               9,1,1,2
                           },
                          {
                              2,1,4,7,
                              6,7,8,9,
                              0,0,0,0
                          }
                      }
 printf("\n %u  %u  %u  %d",a,*a,**a,***a);
```

Ans:1002 1002 1002 1

9.In the following program how would you print 50 using p?

```
main()
{
  int a[]={10,20,30,40,50}
  char *p;
  p=(char*)a;
}
```

Ans: printf(“\n %d”,*((int*)p+4));

10.Where can one think of using pointers?

Ans: At lot of places,for eg
 Accessing array or string elements
 Dynamic memory allocation
 Call by referance
 Implementing linked lists, trees, graphs, and many other data structures
 Etc.

11.In the following program add a statement in the function fun() such that the address of a gets stored in j.

```
main()
{
  int *j;
  void fun(int**);
  fun(&j);
}
void fun(int**k)
{
  int a=10;
 /* add statement here*/
}
```

Ans: *k=&a;

12.How would you implement an array of three function pointers where each function receives two ints and return a float?

Ans: float (*arr[3])(int,int);

13. Would the following program give a compilation error or warning?

```
main()
{
  float i=10,*j;
  void *k;
  k=&i;
  j=k;
  printf(“\n%f”,*j);
}
```

Ans: No. Hee no type casting is required while assigning the value to and from k because conversions are applied automatically when other pointer types are assigned to and from void*.

14. Would the following program compile?

```
main()
{
  int a=10,*j;
  void *k;
  j=k=&a;
  j++;
  k++;
  printf(“\n %u  %u”,j,k);
}
```

Ans: No. An error would be reported in the statement k++ since arithmetic on void pointers is not permitted unless the void pointer is appropriately typecasted.

15. Would the following code compile successfully?

```
main()
{
  printf(“%c”,7[“Sundaram”]);
}
```

Ans: YES. Prints m of sundaram.