CHAPTER 13

COMMAND LINE ARGUMENTS

1. what do the c and v in argc and argv stand for?

Ans:- Count of arguments and vector (array) of arguments.

2. According to ANSI specifications which is correct way of declaring main( ) when it receives command line arguments?
    a. main(int argc, char *argv[])
    b. main(argc, argv)
       int argc; char *argv[];
    c. main( )
       {
          int argc; char *argv[];
       }
    d. None of above.

Ans:- A

3.O/p?

```
/*sample.*/
main(int argc, char **argv)
{
  argc=argc-(argc-1);
  printf("%s", argv[argc-1]);
}
```

Ans:- C:\SAMPLE.EXE

4. If different command arguments are supplied at different times would the output of the following program change?

```
main(int argc, char *argv[])
```

```
{
  printf("%d",argv[argc]);
}
```

Ans:- No.

5. If the following program myprog.c is run from the command line as
   Myprog 1 2 3
   What would be the output?

```
Main(int argc, char *argv[])
{
  int  x;
  for(x=0; x<argc; x++)
     printf("%d", argv[x]);
}
```

   Ans:- C:\MYPROG.EXE 1 2 3

6. If the following program muprog.c is run from the command line as
myprog 1 2 3
What would be the o/p?

```
main(int argc, char *argv[])
{
  int x;
  x=argv[1]+argv[2]+argv[3];
  printf("%d", x);
}
```

a.123
b.6
c.Error
d."123"

Ans:- C

7. If the following program myprog.c is run from the command line as
myprog 1 2 3
What would be the o/p?

```
main(int argc, char *argv[])
{
```

```
  int x, j=0;
  for(x=0; x<argc; x++)
    j=j+atoi(argv[x]);
  printf("5d", j);
}
```

a. 123
b. 6
c. Error
d. "123"

Ans:- B. When atoi( ) tries to convert arv[0] to a number it can not do so, argv[0] being the file name and hence returns a zero

8. If the following program myprog.c is run from the command line as
Myprog one two three
What would be the o/p?

```
main(int argc, char *argv[])
{
  printf("%s", *++argv);
}
```

Ans:- one

9. If the following program myprog.c is run from the command line as
myprog one two three
What would be the o/p?

```
main(int argc, char *argv[])
{
  printf("%s", ++**++argv);
}
```

Ans:- p

10. The variables argc and argv are always local to main?

Ans:- True.

11. The maximum combined length of the command line arguments including spaces between adjacent arguments is
(a:- 128 chars-----b:- 256 chars-----c:- 67 chars------d:- may vary from o.s. to o.s.)

Ans:- D

12. O/p?

```
main(int argc, char *argv[], char –env[])
{
  int x;
  for(x=0; x<arc; x++)
       printf(“%s”, env[x]);
}
```

a. List of all environment variables.
b. List of all command line arguments.
c. Error
d. NULL.

Ans:- B.