1.

```
#include<stdio.h>
#define COL 3
#define ROW 3
int main(void)
{
    int arr[ROW][] = {{1,2,3},{1,2},{1}},i,j;
    for(i=0; i<3; i++)
        for(j=0; j<3; j++)
            printf("%d",arr[i][j]);
    return 0;
}
```

A. Compile time error
B. Run time error
C. Prints array elements
D. None of the above

Answer: A

2.

```
#include<stdio.h>
#define COL 3
#define ROW 2
int main( void )
{
    int arr[ROW][COL] = {10,20,30,40};
    int *ptr[] = {(int *)arr+2, (int *)arr+1, (int *)arr};
    printf("%d %d %d %d\n", ptr[0][1], *(*(ptr + 1) + 0),
                            *(ptr + 0)[2], *(ptr[1] + 1));
    return 0;
}
```

A. 40 20 10 0
B. 40 20 30 10
C. 40 20 10 30
D. 40 20 30 0

Answer: C

3.
What will be the output of following program if base address of arr is 4289999264.

```
#include<stdio.h>
int main(void)
{
    int a[2][2] = {{1,2},{3,4}};
    printf("%u %u %u %u\n", a+1, &a+1,(a+1),&(a+1));
    return 0;
}
```

A. 4289999272 4289999280 4289999272 4289999280
B. 4289999268 4289999280 4289999268 4289999280
C. 4289999272 4289999272 4289999272 4289999272
D. Compile time error
E. None of the above

Answer: D

4.

```
#include<stdio.h>
int main(void)
{
    int a[2][2] = {{1,2},{1,2}}, r,c;
    for(r=0; r<2; r++)
     for(c=0; c<2; c++)
      printf("%d %d %d %d\n",r,c,*(*(a+r)+c),*(*(a+c)+r));

    return 0;
}
```

```
A. 0 0 1 1
   0 1 2 1
   1 0 1 2
   1 1 2 2
B. 0 0 1 1
   0 1 1 2
   1 0 2 1
   1 1 2 2
```

```
C. 0 0 1 1
   0 1 2 2
   1 0 1 1
   1 1 2 2
D. 0 0 1 1
   0 1 1 1
   1 0 2 2
   1 1 2 2
```

Answer: A

5.

```
#include<stdio.h>
int main(void)
{
    int a[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
    int *ptr_a = &a[1][0];
    int **ptr_ptr = &ptr_a;
    printf("%d %d %d\n", **ptr_ptr,*ptr_a, **a);
    return 0;
}
```

```
A. 1 1 1
B. 4 4 4
C. 4 4 1
D. 1 4 1
```

Answer: C

6.

```
#include<stdio.h>
int main(void)
{
    int arr[2][3] = {1,2,3,4,5},row,col;
    for(row=0; row<3; row++)
        for(col=0; col<2; col++)
            printf("%d",arr[row][col]);
    return 0;
}
```

A. 012345
B. 123450
C. 12345
D. 1245 [garbage value] [garbage value]

Answer: D

7.

```
#include<stdio.h>
int main(void)
{
    char arr[4][8] = {"PG-DAC","PG-DESD","PG-DBDA"};
    printf("%c%s",**arr,*(arr+1)+1);
    return 0;
}
```

A. PPG-DAC
B. PPG-DESD
C. PPG-DBDA
D. PG-DESD

Answer: D

8.

```
#include<stdio.h>
int main(void)
{
   char arr[4][10]={"Sunbeam","Karad","Pune","Hinjewadi"};
   char *ptr = (char*)arr[3];
   *ptr++;
   printf("%s %s\n",arr[ptr - arr[3]],--ptr);
   return 0;
}
```

A. Sunbeam Hinjewadi
B. Sunbeam Pune
C. Compiler error
D. None of the above
Answer: A

9.

```
#include<stdio.h>
int main(void)
{
    char arr[5][8] = {"DAC","DESD","DMC","DBDA","PreCAT"};
    char *ptr = arr[4];

    printf("%c.%s\n",*(ptr+3) + *(ptr+3) - ptr[4],
                      (ptr+3) - *(ptr+1) + ptr[1]);
    return 0;
}
```

A. P.CAT
B. E.CAT
C. C.CAT
D. None of the above
E. Compiler error

Answer: B

10.

```
#include<stdio.h>
#define so sizeof
int main(void)
{
    char s[4][32];

    printf("%d %d %d",so(s[2][2]),so(s[2]),so(s));

    return 0;
}
```

A. 1 4 128
B. 1 4 64
C. 1 32 128
D. 1 32 64

Answer: C

11.

```c
#include<stdio.h>
int main(void)
{
    char str[4][12] = {"%s","\"SunBeam\""};
    printf(str[0],str[1]);
    return 0;
}
```

A. \"SunBeam\"
B. %s "SunBeam"
C. "SunBeam"
D. SunBeam

Answer: C

12.
If following program is run like this:
./demo.out This is Demo of Commamd Line Arguments
What will be the output?

```c
#include<stdio.h>
int main(int argc, char *argv[])
{
    int i=0;
    while(argv[i])
    {
        printf("%c",argv[i++][0]); argv++;
    }
    return 0;
}
```

A. .TiDoCLA
B. .ioL
C. TDCA
D. /ioL

Answer: B

13.
If following program is run like this:

```
     ./demo.out Karad Marketyard Hinjewadi
```

What will be th output?

```
#include<stdio.h>
int main(int argc, char *argv[])
{
    int i=0;
    while(*argv++)
    {
        printf("%s ",*argv++);
        argv--;
    }
    return 0;
}
```

A. KaradMarketyardHinjewadi
B. Karad Marketyard Hinjewadi
C. Karad Hinjewadi
D. Karad Marketyard Hinjewadi NULL

Answer: D

14.

```
#include<stdio.h>
int main(int argv, char *argc[])
{
    int loop;
    for(loop = argv; loop <= argv; loop++ )
        printf("%s", argc[loop]);
    return 0;
}
```

A. Error - argv and argc are replaced
B. 0
C. NULL
D. Nothing will be printed

Answer: C

15.

```
#include<stdio.h>
int main(int argc, char *argv[], char *envp[])
{
    int i;
    for(i=0; argv[argc] == NULL; i++)
        printf("%s\n", envp[i]);
    return 0;
}
```

A. Error - Exit value other than 0
B. No output
C. Prints List of environment variables with no error
D. Prints List of environment variables with error

Answer: D

16.
What will be the output of following program if run on command line?

```
#include<stdio.h>
int main(int argc, char *argv[], char *envp[])
{
    printf("%d %c\n", argc, **argv++);
    return 0;
}
```

A. 1 .
B. 1 NULL
C. error with exit value -1
D. 1 \

Answer: A