

1.

```
#include<stdio.h>
int main(void)
{
    int index=0;
    char s[4][12]={"Sunbeam","Karad","Marketyard",
                   "Hinjawadi"};
    char *str=NULL;
    str = "%-15s";
    for(index=1; index<=4 ; index)
        printf(str, s[+index-2]);

    return 0;
}
```

- A. Sunbeam Karad Marketyard Hinjawadi
- B. Karad Marketyard Hinjawadi garbage value
- C. Run time error
- D. Compile time error

Answer: A

2.

```
#include<stdio.h>
#define ROW 5
#define COL 8
int main(void)
{
    static char arr[ROW][COL] = {"PRECAT", "PG-DAC",
                                "PG-DMC", "PG-DBDA", "PG-DESD"};
    printf("%c, %c", *(arr[3]+3), *(*arr+3)+3);
    return 0;
}
```

- A. D, D
- B. C, C
- C. B, B
- D. A, A

Answer: A

3.

```
#include<stdio.h>
#define NO 10
#define LEN 9
int main(void)
{
    char str[NO][LEN]={"PG-DAC", "PG-DESD", "PG-DMC",
                        "PG-DBDA", "PreCat"};
    printf("%d %d %d", sizeof(str), sizeof(str[LEN]) ,
           sizeof(str[NO-1][LEN-1]));
    return 0;
}
```

- A. 90 9 1
- B. 45 9 1
- C. 90 9 4
- D. 90 9 8

Answer: A

4.

```
#include<stdio.h>
#define COL 3
int main(void)
{
    short int arr[][COL]={{11,22},{33,44,55},{66}};
    printf("%d %d %d %d", sizeof(arr),arr[0][1],
           arr[1][2],arr[2][2]);
    return 0;
}
```

- A. 18 22 55 0
- B. 36 22 55 garbage value
- C. 18 22 55 garbage value
- D. 4 22 55 0
- E. Compile time error

Answer: A



5.

```
#include<stdio.h>
int main(void)
{
    static char *s[] ={ "PG-DAC" , "PG-DMC", "PG-DBDA",
                       "PG-DESD"};
    char **ptr[] = {s+3, s+2, s+1, s};
    char ***p=NULL;

    p = ptr;
    p++;
    ++p;
    printf("%s", **p+3);
    return 0;
}
```

- A. PG-DMC
- B. PG-DBDA
- C. PG-DESD
- D. DMC

Answer: D

6.

```
#include<stdio.h>
#define ROW 0
#define COL 0
int main(void)
{
    char arr[][12] = {"SunBeam", "Pune", "Karad",
                      "Hinjawadi"};

    printf(" %s - %c", (*(arr + ROW ) + COL),
           *(*(arr + ROW + 2) + COL));

    printf("\n %s - %c", (arr[ROW+2] + COL),
           (arr[ROW+3][COL]));
    return 0;
}
```

- A. SunBeam - K
Karad - H
- B. Run time error
- C. SunBeam - P
Hinjawadi -
- D. Complile time error

Answer: A

7.

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

- A. 10,10,10,10
20,20,20,20
20,20,20,20
30,30,30,30



- B. 10,10,10,10
30,30,30,30
20,20,20,20
20,20,20,20
- C. 30,30,30,30
20,20,20,20
20,20,20,20
10,10,10,10
- D. 20,20,20,20
10,10,10,10
20,20,20,20
30,30,30,30
- E. 30 30 30 30
40 20 40 20
20 40 20 40
30 30 30 30

Answer: E

8.

```
#include<stdio.h>
#define ROW 5
#define COL 8
int main(void)
{
    char arr[ROW][COL] = { "DAC", "DESD", "DMC" , "DBDA"
                          , "PreCAT"};
    char *ptr = arr[ROW-1];
    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. Compiler error
- E. None of the above

Answer: B

9.

```
#include<stdio.h>
void fun(int **p);
int main(void)
{
    int a[3][3] = {{1,2,3},{4,5,6},{7,8,9}};
    int *ptr_a = &a[1][0];

    printf(" %d ", *ptr_a);
    fun(&ptr_a);
    printf(" %d ", a[1][0]);

    ptr_a = &a[2][1];
    printf(" %d %d \n", *ptr_a, **a);

    return 0;
}
void fun(int **p)
{
    **p = **p+1;
    printf(" %d ", **p);

    return;
}
```

- A. 4 5 5 8 1
- B. 4 4 5 8 1
- C. 4 5 4 8 8
- D. 4 4 5 8 8

Answer: A

10.

```
#include<stdio.h>
#define ROW 2
#define COL ROW
int main(void)
{
    int a[][][COL] = {{1,2},{2,1}}, r,c;

    for(r=0; r<ROW; r++)
        for(c=0; c<COL; c++)
            printf("a[%d][%d] %d %d\n",r,c,
                  *((*(a+r)+c)),*((*(a+c)+r)));

    return 0;
}
```

- A. a[0][0] 1 1
a[0][1] 2 2
a[1][0] 2 2
a[1][1] 1 1
- B. a[0][0] 1 1
a[1][0] 2 2
a[0][1] 2 2
a[1][1] 1 1
- C. a[0][0] 1 2
a[0][1] 1 2
a[1][0] 2 2
a[1][1] 2 2
- D. a[0][0] 1 1
a[0][1] 2 2
a[1][0] 1 1
a[1][1] 2 2

Answer: A

11.

```
#include<stdio.h>
void fun(int (*p)[4]);
int main(void)
{
    int arr[3][4] = {
        {11, 22, 33, 44},
        {44, 33, 22, 11},
        {77, 88, 99}
    };

    int (*ptr)[4]=NULL;

    ptr =arr;
    arr[2][3]=111;

    fun(ptr);

    return 0;
}
void fun(int (*p)[4])
{
    p[2][3]=222;
    printf("%d", *(*(p+2)+3));
    return ;
}
```

- A. 0
- B. 11
- C. compile time error
- D. 111
- E. 222

Answer : E

12.

```
#include<stdio.h>
#define ROW 4
#define COL 12
int main(void)
{
    int index=0;
    char arr[ROW][COL] = {
        "SunBeam", "Karad",
        "Marketyard",
        "Hinjawadi"
    };
    for(index=0; index < ROW-1; index++)
        printf("\n%s - %s.", *(arr),*(arr+index+1) );
    return 0;
}
```

- A. SunBeam - Karad.
SunBeam - Marketyard.
SunBeam - Hinjawadi.
SunBeam - null.
- B. SunBeam - Karad.
SunBeam - Marketyard.
SunBeam - Hinjawadi.
- C. SunBeam - Karad.
Karad - Marketyard.
Marketyard - Hinjawadi.
- D. SunBeam - SunBeam.
Karad - Karad.
Marketyard - Marketyard.
Hinjawadi - Hinjawadi.

Answer: B

13.

If following program having name cmdline is run from the command line as

"./cmdline.out PG-DBDA PG-DESD PG-DMC PG-DAC"

then what would be the output?

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

- A. .PPPP
- B. /DSCC
- C. .DSCC
- D. /PPPP
- E. h-DMC

Answer: A

14.

If following program having name cmdline is run from the command line as

"./cmdline.out PG-DMC PG-DAC PG-DESD PG-DBDA"

then what would be the output?

```
#include<stdio.h>
int main(int argc, char *argv[],char *envp[])
{
    argc--;
    while(argc)
    {
        printf("%c ", argv[argc--][4]);
    }
    return 0;
}
```



- A. M A E B
- B. B E A M
- C. D D D D
- D. Compile time error

Answer : B

15.

If following program having name cmdline is run from the command line as "./cmdline.out" then what would be the output?

```
#include<stdio.h>
int main(int argc, char *argv[], char *envp[])
{
    int index=1;
    for(;;)
    {
        printf("\n [%d] %s",index, envp[index]);
        if( (*(envp+index)==NULL) )
            break;
        else
        {
            index++;
            continue;
        }
    }
    return 0;
}
```

- A. List of all en variables with last value as null
- B. List of all en variables without last value as null
- C. cannot have more than two arguments in main()
 ie. compile time error
- D. Run time error

Answer: A [Note : en stards for environment]

16.

If following program having name cmdline is run from the command line as

"./cmdline.out PG-DAC PG-DMC PG-DESD PG-DBDA"
then what would be the output?

```
#include<stdio.h>
int main(int argc, char *argv[],char *envp[])
{
    int index;

    for(index=argc-1; index>=0; index--)
        printf("%d] %s\t", index, argv[index]);

    return 0;
}
```

- A. 4] PG-DBDA 3] PG-DESD 2] PG-DMC 1] PG-DAC
0] cmdline.out
- B. 0] cmdline.out 1] PG-DAC 2] PG-DMC 3] PG-DESD
4] PG-DBDA
- C. 1] PG-DAC 2] PG-DMC 3] PG-DESD 4] PG-DBDA
- D. 4] PG-DBDA 3] PG-DESD 2] PG-DMC 1] PG-DAC
- E. None of above

Answer : A

17.

If following program having name cmdline is run from the command line as

"./cmdline.out PG-DBDA PG-DESD PG-DMC PG-DAC"

then what would be the output?

```
#include<stdio.h>
int main(int argc, char *argv[])
{
    int index;
    for(index=0; index<argc; ++index)
        printf("%-10s", *++argv);
    return 0;
}
```

- A. G-DBDA -DESD MC C (null)
- B. cmdline.out G-DBDA -DESD DMC AC
- C. cmdline.out PG-DBDA PG-DESD PG-DMC PG-DAC
- D. PG-DBDA PG-DESD PG-DMC PG-DAC (null)

Answer: D