

**PF LAB:8**

**24K-0514**

**MUHAMMAD  
UMER FAROOQ**

## QUESTION 1:

Write a C program to generate the following pattern:

```
1
1 2
1 2 3
1 2 3 4
```

```
1  #include <stdio.h>
2  int main()
3  {
4      for(int i=1;i<=4;i++)
5      {
6          for(int j=1;j<=i;j++)
7          {
8              printf("%d ",j);
9          }
10
11         printf("\n");
12     }
13     return 0;
14 }
```

OUTPUT:

```
C:\Users\HP>gcc pattern1.c -o pattern1.exe

C:\Users\HP>pattern1.exe
1
1 2
1 2 3
1 2 3 4

C:\Users\HP>
```

## QUESTION 2:

Write a program to take a 2D array input from the user and display its transpose.

```
1  #include <stdio.h>
2  int main()
3  {
4      int array[100][100];
5      int r,c;
6      printf("\n Enter number of rows:");
7      scanf("%d",&r);
8      printf("\n Enter number of columns:");
9      scanf("%d",&c);
10
11
12      printf("\n Enter matrix:\n");
13      for(int i=0;i<r;i++)
14      {
15          printf("\n Enter row %d of matrix:\n",i+1);
16          ;
17          for(int j=0;j<c;j++)
18          {
19              scanf("%d",&array[i][j]);
20          }
21          printf("\n");
22      }
23      printf("\n Actual matrix is :\n");
24      for(int i=0;i<r;i++)
25      {
26          for(int j=0;j<c;j++)
27          {
28              printf("%d ",array[i][j]);
29          }
30          printf("\n");
31      }
32      printf("\n Transpose matrix is:\n");
33      for(int i=0;i<r;i++)
34      {
35          for(int j=0;j<c;j++)
36          {
37              printf("%d ",array[j][i]);
38          }
39          printf("\n");
40      }
41      return 0;
42  }
```

## OUTPUT:

```
C:\Users\HP>gcc transpose.c -o transpose.exe
C:\Users\HP>transpose.exe
Enter number of rows:3
Enter number of columns:3
Enter matrix:
Enter row 1 of matrix:
1
2
3
Enter row 2 of matrix:
4
5
6
Enter row 3 of matrix:
7
8
9
Actual matrix is :
1 2 3
4 5 6
7 8 9
Transpose matrix is:
1 4 7
2 5 8
3 6 9
C:\Users\HP>_
```

### QUESTION 3:

Create a 3D array representing 2 pages of a 3x3 matrix. Initialize it and find the sum of all the elements on each page.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int array[2][3][3]=
6      {
7          {{1,2,3},{4,5,6},{7,8,9}},{{10,11,12},{13,14,15},{16,17,18}}
8      };
9      int sum[2]={0};
10     for(int i=0;i<2;i++)
11     {
12         for(int j=0;j<3;j++)
13         {
14             for(int k=0;k<3;k++)
15             {
16                 sum[i]=sum[i]+array[i][j][k];
17             }
18         }
19     }
20     for(int i=0;i<2;i++)
21     {
22         printf("sum of page %d is %d\n",i+1,sum[i]);
23     }
24     return 0;
25 }
```

OUTPUT:

```
C:\Users\HP>gcc 3darray.c -o 3darray.exe

C:\Users\HP>3darray.exe
sum of page 1 is 45
sum of page 2 is 126

C:\Users\HP>
```

## QUESTION 4:

Write a C program that generates a sequence of prime numbers within a given range using nested loops.

```
1  #include<stdio.h>
2  int main()
3  {
4      int n1,n2,count=0;
5      printf("\n Enter starting numnber:");
6      scanf("%d",&n1);
7      printf("\n Enter ending number:");
8      scanf("%d",&n2);
9
10     for(int i=n1;i<=n2;i++)
11     {
12         for(int j=2 ;j<i;j++)
13         {
14             if(i%j==0)
15             {
16                 break;
17             }
18             else
19             {
20                 count++;
21                 if(count==i-2)
22                 {
23                     printf("%d ",i);
24                 }
25             }
26         }
27         count=0;
28     }
29     return 0;
30 }
31
32 }
```

OUTPUT:

```
C:\Users\HP>gcc prime1.c -o prime1.exe

C:\Users\HP>prime1.exe

Enter starting numnber:3

Enter ending number:100
3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

## QUESTION 5:

Generate a pattern of odd numbers in decreasing order starting from a user-specified number using nested loops.

```
1  #include<stdio.h>
2  int main()
3  {
4      int n;
5      printf("\n Enter starting numnber:");
6      scanf("%d",&n);
7
8
9      for(int i=n;i>=1;i--)
10     {
11         for(int j=2 ;j<=2;j++)
12         {
13             if(i%j!=0)
14             {
15                 printf("%d\n",i);
16             }
17         }
18     }
19
20     return 0;
21 }
```

OUTPUT:

```
Enter starting numnber:30
29
27
25
23
21
19
17
15
13
11
9
7
5
3
1
```

## QUESTION 6:

Write a C program to find the saddle point(s) in a given 3x3 matrix. A saddle point is an element that is the smallest in its row and the largest in its column.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int a[3][3], res[3]={0}, res2[3]={0};
6      int count=0;
7
8      for(int i=0; i<3; i++)
9      {
10         printf("Enter row %d:", i+1);
11         for(int j=0; j<3; j++)
12         {
13             scanf("%d", &a[i][j]);
14         }
15
16         printf("\n");
17     }
18
19     for(int i=0; i<3; i++)
20     {
21         int low=1000000;
22         for(int j=0; j<3; j++)
23         {
24             if(a[i][j]<low)
25             {
26                 low=a[i][j];
27                 res[i]=low;
28             }
29         }
30     }
31
32     for(int i=0; i<3; i++)
33     {
34         int high=0;
35         for(int j=0; j<3; j++)
36         {
37             if(a[j][i]>high)
38             {
39                 high=a[j][i];
40                 res2[i]=high;
41             }
42         }
43     }
44
45 }
```



```

for(int i=0;i<3;i++)
{
    for(int j=0;j<3;j++)
    {
        if(res[i]==res2[j])
        {
            printf("\n Saddle element is %d",res[i]);
            count++;
        }
    }
}
if(count==0)
{
    printf("\n No saddle element found");
}

return 0;
}

```

OUTPUT:

```

C:\Users\HP>saddle.exe
Enter row 1:1
2
3

Enter row 2:4
5
6

Enter row 3:7
8
9

Saddle element is 7

```

## QUESTION 7:

Write a C program to multiply two matrices of size 3x3 and display the result matrix.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int m,n,p,q;
6      int a[m][n],b[p][q],res[m][q];
7
8      printf("\n Enter rows of first matrix:");
9      scanf("%d",&m);
10     printf("\n Enter columns of first matrix:");
11     scanf("%d",&n);
12
13     printf("\n Enter rows of second matrix:");
14     scanf("%d",&p);
15     printf("\n Enter columns of second matrix:");
16     scanf("%d",&q);
17     if(n!=p)
18     {
19         printf("\n Matrices cannot be multiplied");
20         return 0;
21     }
22     else
23
24     {
25         printf("\n Matrix 1:\n");
26         for(int i=0;i<m;i++)
27         {
28             printf("\n Enter row %d of matrix 1:",i+1);
29             for(int j=0;j<n;j++)
30             {
31                 scanf("%d",&a[i][j]);
32             }
33
34             printf("\n");
35         }
36         printf("\n Matrix 2:\n");
37         for(int i=0;i<p;i++)
38         {
39             printf("\n Enter row %d of matrix 2:",i+1);
40             for(int j=0;j<q;j++)
41             {
42                 scanf("%d",&b[i][j]);
43             }
44             printf("\n");
45         }
46     }
```

```
for(int i=0;i<m;i++)
{
    for(int j=0;j<q;j++)
    {
        res[i][j]=0;
        for(int k=0;k<n;k++)
        {
            res[i][j]+=a[i][k]*b[k][j];
        }
    }
}

for(int i=0;i<m;i++)
{
    for(int j=0;j<q;j++)
    {
        printf("%d ",res[i][j]);
    }
    printf("\n");
}

return 0;
```

```
}
```

OUTPUT:

Enter rows of first matrix:3

Enter columns of first matrix:3

Enter rows of second matrix:3

Enter columns of second matrix:3

Matrix 1:

Enter row 1 of matrix 1:1

2

3

Enter row 2 of matrix 1:4

5

6

Enter row 3 of matrix 1:7

8

9

Matrix 2:

Enter row 1 of matrix 2:9

8

7

Enter row 2 of matrix 2:6

5|

4

Enter row 3 of matrix 2:3

2

1

30 24 18

84 69 54

138 114 90

## QUESTION 8:

Write a C program to generate a diamond shape pattern using nested loops. The program should take the number of rows for the upper half of the diamond as input from the user.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int n;
6
7      printf("Enter the number of rows: ");
8      scanf("%d",&n);
9
10
11     for (int i=1;i<=n;i++)
12     {
13
14         for (int j=1;j<=n-i;j++)
15         {
16             printf(" ");
17         }
18
19         for (int k=1;k<=2*i-1;k++)
20         {
21             printf("*");
22         }
23         printf("\n");
24     }
25
26
27     for (int i=n-1;i>=1;i--)
28     {
29
30         for (int j=1;j<=n-i;j++)
31         {
32             printf(" ");
33         }
34         for (int k=1;k<=2*i-1;k++)
35         {
36             printf("*");
37         }
38         printf("\n");
39     }
40
41     return 0;
42 }
```

OUTPUT:

```
C:\Users\HP>gcc pattern2.c -o pattern2.exe
```

```
C:\Users\HP>pattern2.exe
```

```
Enter the number of rows: 4
```

```
  *
 ***
*****
*****
*****
 ***
  *
```