Q)Write a C program to do the following by passing matrix as parameter:

1) Matrix addition and subtraction.
2) Matrix multiplication.
3) Sum of principle and non principle diagonal of matrix.
4) Sum of rows and columns.
5) Print the transpose
6) Check if a given matrix is symmetric or not.

14/6/23

Date

YOUVA

Prog 1

Q Write a c / c++ program to do the following

i) Pass the matrices as parameters
i) Addition / Subtraction
ii) Multiplication
iii) Sum of principal / non principle diagonal
iv) Sum of Rows & coloumns
v) Print the transpose of a given Matrix
vi) Symmetric or not.

```
#include <stdio.h>
void sop( int n, int mat1 [n][n])
{
 int sump=0, sumnp=0, i, j, k;
 for (i=0; i<n; i++)
 {
 for (j=0; j<n; j++)
  {
   if (j==i)
  {
   sump+ =mat1 [i] [j];
 }
}
}
for (i=0, i<n; i++)
{ for ( j=0, j<n, j++){
   if (j==i) {
   sump+ = mat1 [i] [n-1-i];
 }
}
}
printf ("sum of principles diagonal is
      %d \n sum of non principle diagonal
      is %d \n", sump, sumnp);
}
void main ()
{ int m, n, choice, i, j;
 printf (" Enter the values of n for nxn
            and matrix\n");
 Scanf (" %d" &n);
 int mat1 [n] [n];
 int mat2 [n] [n];
printf ("Enter the Values for matrices1\n");
for (i =0; i<n i++) {
```

```
for(j=0,j<n,j++){
  Scanf ("%d", & mat2 [i][j]);
 }
}
printf ("\nMenu\n1. Add \n2. Sub\n3, Mul
        \n4. Sum of principal & non principle
        diagonals \n5. Sum of rows & columns
        \n6. Transpose matrix \n7. Check if
        the matrix is symmetric \n8. Exit
        \n\n");
While (1){
 printf ("Enter your choice\n");
 Scanf ("%d" & choice);
 Switch (choice) {
case 1: add (n, mat1, mat2);
 break;
case 2: sub(n, mat1, mat2);
 break
case 3: multiply (n, mat1, mat2);
 break
case4: SoP (n, mat1);
 break
case 5: Sumrc(n, mat1, mat2);
 break.
case 6: transpose (n, mat1);
 break
case 7: sym (n, mat1);
 break;
 case 8: exit (0);
 default : printf ("wrong Choice\n");
 }
 }
}
void add (int n, int mat1 [n][n],
          int mat2 [n][n]){
```

Date:

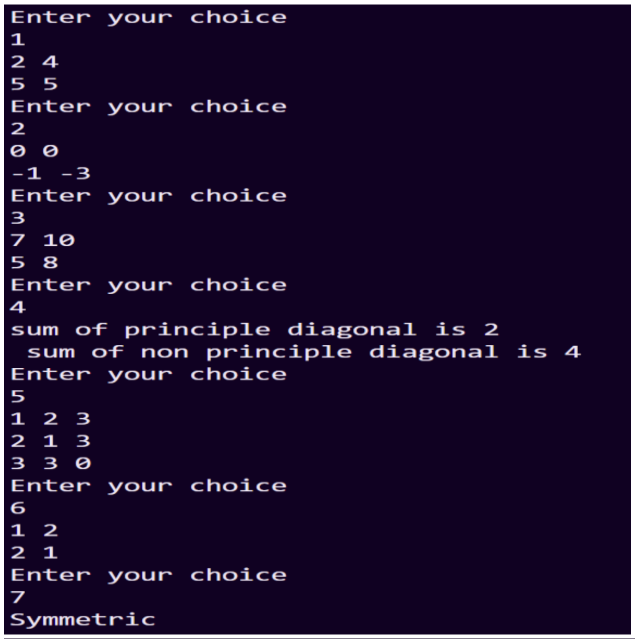
```
int i, j;
int sum [n][n];
 for (i=0; i<n, i++){
   for (j=0; j<n; j++){
    sum [i][j] = mat1 [i][j] + mat2[i] [j];
  }
 }
 for (i=0; i<n; i++){
  for (j=0; j<n; j++){
   printf ("%d", sum [i][j]);
  }
  printf ("\n");
 }
}
void sub (int n, int mat1 [n] [n], int
             mat2[n] [n])
{ int i, j;
  int sum [n][n];
 for (i= 0; i<n; i++){
  for (j=0; j<n; j++) {
  sum [i][j] = mat [i][j] - mat2[i][j];
  }
 }
for (i=0 ; i < n ; i++){
  for (j=0; j<n; j++){
   printf ("%d", sum [i][j]);
  } printf ("\n");
 }
}
Void multiply (int n, int mat1 [n][n],
               int mat2 [n][n]) {
int sum = 0; i, j, k;
 int prod [n][n];
for( int i = 0; i <n; i++){
 for (int j = 0; j<n; j++){
```

```
prod[A][j]=0;
for (int k=0; k<n; k++){
  prod[i][j]+= mat1[i][k] * mat2[k]
 }
 }
}
for (i=0; i<n; i++){
 for (j=0; j<n; j++){
   printf("%d", prod[i][j]);
 }
 printf("\n");
}
}
void sumrc(int n, int mat1[n][n]){
 int sumr, sumc, i, j;
 int mat3[n+1] [n+1];
 for (i=0; i<n; i++){
 sumr =0
 for (j=0; j<n; j++){
mat3[i][j] =mat1[i][j];
 sumr+= mat[i][j];
 }
mat3[i][n] =sumr;
 }
 for (j=0; j<n; j++){
  sum c=0;
 for (i=0; i<n; i++){
   sumc+ = mat1[i][j];
  }
 mat3[n][j] =sumc;
} mat3[n][n]=0;
 for(int i=0; i<n+1; i++){
    for (int j=0; j<n+1; j++){
printf("%d", mat3[i][j])
}
```

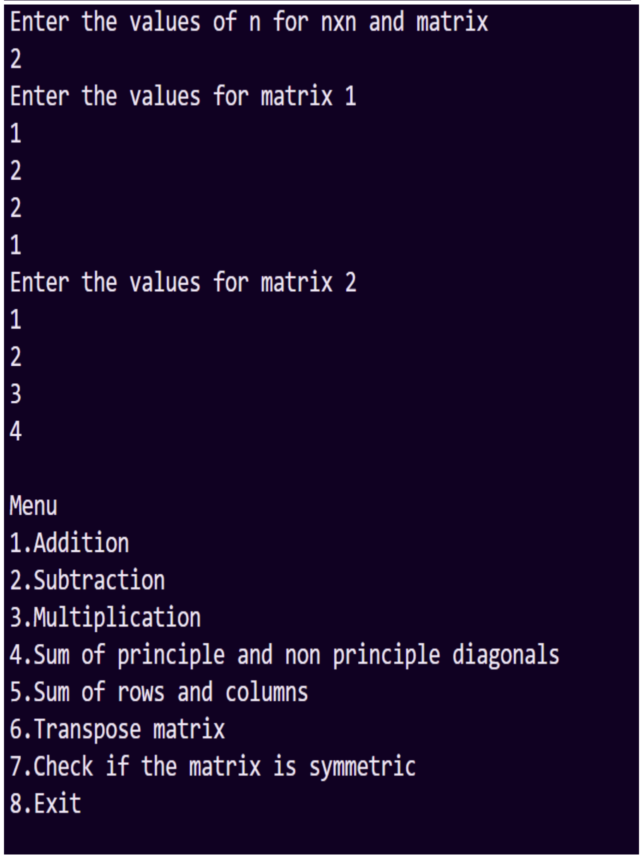
```
printf (" /n");
}
}

void transpose (int n, int mat1 [n][n]){
 int transpose [n][n];
 for (int i =0; i<n; i++){
  for ( int j =0; j<n; j++){
   transpose [j] [i] = mat [i][j];
}
}

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

void sym(int n, int mat1 [n][n]){
 int flag= 1;
 for ( int i = 0; i<n; i++){
  for (int j = 0; j <n; j++){
   if( mat1 [i] [j] != mat1[j][i]){
    flag = 0;
  }
 }
}

if (flag = =0) printf ("not symmetric\n
else printf(" symmetric\n");
}
```

[illegible]
20/6/23

Output:



```
Enter your choice
1
2 4
5 5
Enter your choice
2
0 0
-1 -3
Enter your choice
3
7 10
5 8
Enter your choice
4
sum of principle diagonal is 2
 sum of non principle diagonal is 4
Enter your choice
5
1 2 3
2 1 3
3 3 0
Enter your choice
6
1 2
2 1
Enter your choice
7
Symmetric
```



```
Enter the values of n for nxn and matrix
2
Enter the values for matrix 1
1
2
2
1
Enter the values for matrix 2
1
2
3
4

Menu
1.Addition
2.Subtraction
3.Multiplication
4.Sum of principle and non principle diagonals
5.Sum of rows and columns
6.Transpose matrix
7.Check if the matrix is symmetric
8.Exit
```