

Practice Array 2D 1 (09-08-2021)

Q0. Create following program:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(){
    int x[5][5],y[5][5], i, j, temp    ;
    srand(time(0));
    for (i=0;i<5;i++){
        for (j=0;j<5;j++){
            x[i][j] = rand()%10;
            printf("%2d ", x[i][j]);
        }
        printf("\n");
    }
    printf("-----\n");
    for (i=0;i<5;i++){
        for (j=0;j<5;j++){
            y[i][j] = rand()%10;
            printf("%2d ", y[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

Q1. Extend above program, compare corresponding elements, print 1 if elements match, otherwise print 0, see sample run and explanation for your understanding

Explanation: After comparison 0 & 1 are printed in bold at the bottom. Last element of first row is 7 in both arrays, therefore we have 1 in last column of first row. Next fourth element of second row in both arrays is 4, therefore we have 1 in fourth column of second row. Similarly, in third row, second column we have 3 in both arrays, therefore 1 is printed in second column of third row.

```
2 9 2 5 7
7 0 6 4 1
1 3 8 3 2
1 8 4 9 4
8 0 7 3 5
```

```
-----
0 7 5 2 7
5 4 5 4 6
6 3 6 6 0
5 4 9 5 3
4 0 2 1 5
```

```
-----
0 0 0 0 1
0 0 0 1 0
0 1 0 0 0
0 0 0 0 0
0 1 0 0 1
```

Move to next page for more questions

Q2. Consider x & y as two matrices. Add and print them. Again see output for your understanding:

5	4	3	0	7
4	9	9	1	9
4	1	8	5	3
0	3	9	2	5
7	4	8	4	3

7	7	4	6	7
6	8	8	0	9
6	6	2	9	8
6	8	0	6	2
2	0	1	3	9

12	11	7	6	14
10	17	17	1	18
10	7	10	14	11
6	11	9	8	7
9	4	9	7	12

Q00. Create and run following program: #include <stdio.h>

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(){
    int x[5][5], i, j, temp;
    srand(time(0));
    for (i=0;i<5;i++){
        for (j=0;j<5;j++){
            x[i][j] = rand()%100;
            printf("%2d ", x[i][j]);
        }
        printf("\n");
    }
    printf("-----\n");
    return 0;
}
```

Q3. Extend **Q00** program and print diagonal in single line. See output for your understanding:

```
60 28  9 30 14
35 10 50  8 27
64 30 51 17  0
 2 43 40 44 26
50 38 59  3 23
```

```
-----
60 10 51 44 23
```

Q4. Extend **Q3**, print 0 for non-diagonal values and print diagonal at its position. . See output for your understanding:

```
60 28  9 30 14
35 10 50  8 27
64 30 51 17  0
 2 43 40 44 26
50 38 59  3 23
```

```
-----
60 0 0 0 0
0 10 0 0 0
0 0 51 0 0
0 0 0 44 0
0 0 0 0 23
```