MCQ:

Week 4 : Assignment 4

Your last recorded submission was on 2025-02-16, 17:14 IST **Due date: 2025-02-19, 23:59 IST.**

1) Which of the following options provides the general formula for the magic constant of a magic square of size n, where all elements are distinct numbers from 1 to n^2? ***1 point***

- [x] $\frac{n(n^2+1)}{2}$
- [ ] $\frac{n^3}{2}$
- [ ] $\frac{n^3+2n}{2}$
- [x] $\frac{n^4+n^2}{2n}$

2) What would the magic constant be for a magic square of size 7, given that all elements in the square are distinct numbers from 1 to 49? ***1 point***

- ○ 260
- ○ 111
- ◉ 175
- ○ 165

3) Does transposing any magic square change the sum across some rows/columns/diagonal? ***1 point***

- ○ Yes
- ◉ No

4) Which of the following are valid magic squares ? ***1 point***

- [x] $\begin{bmatrix} 10 & 3 & 13 & 8 \\ 5 & 16 & 2 & 11 \\ 4 & 9 & 7 & 14 \\ 15 & 6 & 12 & 1 \end{bmatrix}$
- [ ] $\begin{bmatrix} 20-e & 6-e & 26-e & 16-e \\ 10-e & 32-e & 4-e & 22-e \\ 8-e & 18-e & 14-e & 28-2e \\ 30-e & 12-e & 24-e & 2-e \end{bmatrix}$
- [x] $\begin{bmatrix} 1 & 14 & 4 & 15 \\ 8 & 11 & 5 & 10 \\ 13 & 2 & 16 & 3 \\ 12 & 7 & 9 & 6 \end{bmatrix}$
- [x] $\begin{bmatrix} \pi & 14\pi & 4\pi & 15\pi \\ 8\pi & 11\pi & 5\pi & 10\pi \\ 13\pi & 2\pi & 16\pi & 3\pi \\ 12\pi & 7\pi & 9\pi & 6\pi \end{bmatrix}$

5) What is the minimum number of people required to ensure that at least three of them share the same 30-minute birth interval? The intervals start from 12:00 AM and each interval lasts for half an hour.

97

1 point

6) Calculate the magic constant for 5x5 square, where all elements are distinct numbers from 1 to 25, is it same as the magic constant for Ramanujan's magic square ?

If yes, enter 0, else enter the absolute difference between the two.
Hint: Search the about Ramanujan's magic square.

74

7) What task does **mystery1()** perform? ***1 point***

```
def mystery1(number):
  list1 = []
  for i in range(1, number):
    if number % i == 0:
      list1.append(i)
  return list1

def mystery2(n1,n2):
  flag = False
  list2 = []
  for i in mystery1(n1):
    for j in mystery1(n2):
      if i == j:
        flag = True
        list2.append(i)
  if len(list2) > 1:
    print("Completed")
```

- ○ Calculate factorial of number n.
- ○ Calculate factors of number n.
- ○ Calculate factors of number n+1 excluding n.
- ◉ Calculate factors of number n excluding n.

8) For what **n1, n2** will the variable **flag** inside **mystery2()** be equal to **True**? ***1 point***

- ☐ 1,2
- ☑ 2,3
- ☑ 3,4
- ☐ 0,0

9) If all possible pairs of prime numbers between 0 and 10, are given to n1 and n2, for how many pairs would 2 print "Completed" ? ***1 point***

- ○ It will print "Completed" only for pairs (2,3)(7,2),(2,5), and for the remaining it would not print "Completed".
- ○ It will print "Completed" only for pair (2,3), and for the remaining other pairs of primes it would not print "Completed".
- ○ It will print "Completed" for all pairs of primes between 0 and 20.
- ◉ It will not print "Completed" for any pair.

10) If numbers of pairs of primes which result in mystery2 to print "Completed" are lesser than 1, Can we edit the code in **mystery2()** so that "Completed" is always printed for any pair of primes ? ***1 point***

- ○ Yes, we can change the logic for setting **flag** variable to False.
- ○ No, it is logically not possible.
- ◉ Yes, we can change/decrease the threshold for length of **list2** in the last **if** block.
- ○ Yes, we can change the initial value of flag to False, instead of True.

Question - 1

Week 4: Programming Assignment 1

Due on 2025-02-20, 23:59 IST

Create a Python program that checks whether a given square matrix is diagonal. A **diagonal matrix** is a square matrix (same number of rows and columns) where **all the entries outside the main diagonal are zero**. The program should prompt the user to input the dimensions of the matrix and then input the matrix elements. The program should then determine whether the matrix is diagonal and print `1` if it is, otherwise print `0`.

Input Format:
The first input is an integer r , the number of rows and columns in the matrix.
The next r lines each contain r integers, representing the elements of each row of the matrix.

Output Format:
The output is `1` if a matrix is diagonal, otherwise `0`.

Example:

Input:

```
2
1 0
0 1
```

Output:

```
1
```

```python
~~~THERE IS SOME INVISIBLE CODE HERE~~~
def is_diagonal_matrix(matrix, r):
    for i in range(r):
        for j in range(r):
            if i != j and matrix[i][j] != 0:
                return 0
    return 1

r = int(input())
matrix = [list(map(int, input().split())) for _ in range(r)]
print(is_diagonal_matrix(matrix, r),end="")

```

Compilation : Passed

Public Test Cases: 3 / 3 Passed

Note: These tests may not be considered while scoring. *Know more.*

Public Test Cases	Input	Expected Output	Actual Output	Status
Test Case 1	2 1 0 0 1	1	1	Passed
Test Case 2	2 1 2 0 3	0	0	Passed
Test Case 3	3 5 0 0 0 3 0 0 0 9	1	1	Passed

Question - 2

Week 4: Programming Assignment 2

Due on 2025-02-20, 23:59 IST

Create a Python program that adds the transpose of a given matrix by a scalar. The program should prompt the user to input the dimensions of the matrix, the elements of the matrix, and the scalar value. The program should then compute the transpose of the matrix, add it by the scalar, and print the resulting matrix.

Input:
The first input is an integer r , the number of rows in the matrix.
The second input is an integer c , the number of columns in the matrix.
The next r lines each contain c integers, representing the elements of the matrix.
The final input is an integer s, representing the scalar value.

Output Format:
The output consists of c lines, each containing r integers, representing the elements of the resulting matrix after adding the transpose of the original matrix by the scalar.

Example:

Input:

```
2
3
1 2 3
4 5 6
2
```

Output:

```
3 6
4 7
5 8
```

```
~~~THERE IS SOME INVISIBLE CODE HERE~~~
r = int(input())
c = int(input())
matrix = [list(map(int, input().split())) for _ in range(r)]
s = int(input())

transpose_matrix = [[matrix[j][i] + s for j in range(r)] for i in range(c)]

for i in range(c):
    print(" ".join(map(str, transpose_matrix[i])), end="\n" if i < c - 1 else "")

```

Compilation : Passed

Public Test Cases: 3 / 3 Passed

Note: These tests may not be considered while scoring. Know more.

Public Test Cases	Input	Expected Output	Actual Output	Status
Test Case 1	2 3 1 2 3 4 5 6 2	3 6\n 4 7\n 5 8	3 6\n 4 7\n 5 8	Passed
Test Case 2	3 2 1 2 3 4 5 6 3	4 6 8\n 5 7 9	4 6 8\n 5 7 9	Passed
Test Case 3	2 2 -1 0 2 -3 4	3 6\n 4 1	3 6\n 4 1	Passed

Question - 3

Week 4: Programming Assignment 3

Due on 2025-02-20, 23:59 IST

Create a Python program that checks whether a given square matrix is symmetric. A matrix is symmetric if its transpose is equal to the matrix itself. The program should prompt the user to input the dimensions of the matrix and then input the matrix elements. The program should then determine whether the matrix is symmetric and print `1` if it is, otherwise print `0`.

Input Format:
The first input is an integer r , the number of rows and columns in the matrix.
The next r lines each contain r integers, representing the elements of each row of the matrix.

Output Format:
The output is `1` if a matrix is symmetric, otherwise `0`.

Example:

Input:

```
2
1 2
2 1
```

Output:

```
1
```

```
~~~THERE IS SOME INVISIBLE CODE HERE~~~
r = int(input())
matrix = [list(map(int, input().split())) for _ in range(r)]

# Check if matrix is equal to its transpose
symmetric = all(matrix[i][j] == matrix[j][i] for i in range(r) for j in range(r))

print(1 if symmetric else 0,end="")

```

Compilation : Passed

Public Test Cases: 3 / 3 Passed

Note: These tests may not be considered while scoring. Know more.

Public Test Cases	Input	Expected Output	Actual Output	Status
Test Case 1	2 1 2 2 1	1	1	Passed
Test Case 2	2 2 3 4 5	0	0	Passed
Test Case 3	3 1 2 3 2 5 7 3 7 9	1	1	Passed

Petchiammal N (CoE)