Daily Challenge **Happy Coding from necse**  **SkillRack**

Maximum Sum - No Adjacent

The program must accept an integer matrix of size **2*N** as the input. The program must print the maximum sum such that no two chosen integers are adjacent vertically, diagonally or horizontally in the matrix.

Boundary Condition(s):
2 <= N <= 100
1 <= Matrix element value <= 10^5

Input Format:
The first line contains N.
The next two lines, each contains N integer values separated by a space.

Output Format:
The first line contains the maximum sum.

Example Input/Output 1:
Input:
3
2 5 6
3 1 1

Output:
9

Explanation:
The **4** possible ways are given below.
(1, 1) and (1, 3) => 2 + 6 = **8**.
(1, 1) and (2, 3) => 2 + 1 = **3**.
(2, 1) and (1, 3) => 3 + 6 = **9**.
(2, 1) and (2, 3) => 3 + 1 = **4**.
Here the maximum sum is **9**, which is printed as the output.

Example Input/Output 2:
Input:
5
1 3 5 7 8
2 4 6 8 10

Output:
18

Explanation:
The maximum sum is obtained as given below.
(2, 1), (2, 3) and (2, 5) => 2 + 6 + 10 = **18**.

Example Input/Output 3:
Input:
4
9 8 7 3
7 6 4 9

Output:
18

Max Execution Time Limit: 50 millisecs

Ambiance

Java (12.0)

```java
import java.util.*;
public class Hello {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int n = sc.nextInt();

        int arr1[] = new int[n];
        int arr2[] = new int[n];

        for(int i=0;i<n;i++) arr1[i]=sc.nextInt();

        for(int i=0;i<n;i++) arr2[i]=sc.nextInt();


        int arr[] = new int[n];

        for(int i=0;i<n;i++)
            arr[i] = Math.max(arr1[i],arr2[i]);

        int dp[] = new int[n];
        dp[0] = arr[0];
        dp[1] = Math.max(arr[0],arr[1]);

        for(int i=2;i<n;i++)
            dp[i] = Math.max(dp[i-1],dp[i-2]+arr[i]);


        System.out.println(dp[n-1]+"");


    }
}
```

1912067@nec

Code did not pass the execution

Input:

```
5
1 3 5 7 8
2 4 6 8 10
```

Expected Output:

```
18
```

Your Program Output:

```
0
[2, 4, 22, 0, 0]
```

Save Run