Santa Claus is Coming to Town

"You better watch out! You better not cry! You better not pout, I'm telling you why. Santa Claus is coming to town!"

Today is Christmas, at least according to Santa Claus! "Ho ho ho!," Santa exclaims. "I've got presents to deliver!"

Yes, today is finally Christmas, at least according to Santa Claus. It is a wonderful holiday season and Santa cannot wait to deliver the gifts that he has prepared for children around the world.

Santa, as we all know, has his priced sled and reindeers that will take him around the globe, visiting the homes of children worldwide to deliver the gifts during Christmas eve, so that when the children wake up in Christmas morning, they are greeted by the wonderful present placed wonderfully under the Christmas tree.

Santa Claus will start his journey from his home, the North Pole. After finishing his task, Santa will return to his workshop in the North Pole again.

"I am ready to deliver the gifts! Oh, I am so excited!," Santa shouted merrily. "But, where should I go first? In fact, what should be my route for tonight?," Santa wonders. "I can go magically from a city to another city using my magical sled and reindeers, but I do not want them to be too exhausted either."

The truth is, it will be very hard for Santa to manually come up with the most efficient tour. It will be too tedious if he were to analyze all possible tours and pick the best one by himself. If only there were someone who could help him. Therefore, Santa summons you, his head elf, to devise a strategy for Santa. Santa then briefs you about the problem for tonight:

"I have to visit **N** cities before sunrise. I can go from a city to any other cities. I do not want my reindeers to be exhausted and I also need to save time. I need a tour which starts and ends at the North Pole and **visits all N cities in any order**, such that the **total distance travelled is minimized**. I also **do not want to revisit cities** I have already visited.".

As head elf, you quickly go and help Santa Claus so that the children will not be disappointed when they wake up at Christmas morning.

Good luck!

Input

The first line contains a single integer **N (1 <= N <= 10),** the number of cities that Santa needs to visit.

The next **N + 1** lines consist of **N + 1** space-separated integers $D_{i,j}$ (1 <= $D_{i,j}$ **<= 1000,** except when i=j). The integer at row `i`, column `j` denotes the distance from city `i` to city `j`. The North Pole is considered to be city 0. All integers at position (`i,j`) will be the same as (`j,i`), meaning that it takes the same distance to travel both ways between any two cities. The integer at (`i,j`) where `i` = `j` is 0.

Output

Print the minimum cost of the optimal tour for Santa Claus to deliver the presents, starting from and returning to the North Pole. Your output should contain a newline character.

Sample Input 1
```
3
0 1 1 1
1 0 1 1
1 1 0 1
1 1 1 0
```

Sample Output 1
```
4
```

Sample Input 2
```
3
0 9 1 1
9 0 1 1
1 1 0 1
1 1 1 0
```

Sample Output 2
```
4
```

Hint

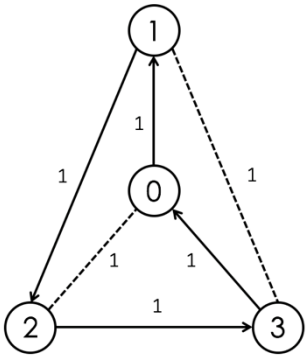
You can store the distance array as a 2D-array, usually called an "adjacency matrix". An adjacency matrix of size **(N+1) x (N+1)** will store information of the distances of each city to all other cities, including the North Pole. The number located at row `i` and column `j` indicates the distance to go from city `i` to city `j`. In our case, the number at (`i,j`) will be the same as (`j,i`) and the integer at (`i,j`) where `i` = `j` is 0.

In sample input 2 above, the first row indicates the distance from the North Pole to all other cities. The distance from the North Pole to city 1 is 9, city 2 is 1, and city 3 is 1. The next row contains the distances from city 1 to all other cities (including the North Pole), and so on.

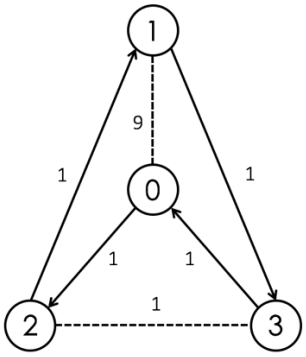
Explanation

In the first sample input, all tours will give you the same cost: 3. An example of the tour is 0-1-2-3-0, with the distance from 0-1, 1-2, 2-3, and 3-0 being 1, giving the total cost of 4 for that tour. The left graph below represents the cities with the appropriate numbers and distances for the first sample input, with the arrows indicating one of the possible optimal tours.

In the second sample input, a valid tour would be 0-1-2-3-0, but we see that this tour will give Santa a cost of 12 (0-1: 9, 1-2: 1, 2-3: 1, and 3-0: 1). A more efficient tour would be 0-2-1-3-0 with cost 4 (0-2: 1, 2-1: 1, 1-3: 1, and 3-0: 1). Hence, the answer for this case is 4. The right graph below represents the cities with the appropriate numbers and distances for the second sample input, with the arrows indicating one of the possible optimal tours.



Graph 1 (Sample Input 1)



Graph 2 (Sample Input 2)

Skeleton

You are given the skeleton file `SantaClaus.java`. You should see a non-empty file when you open the skeleton, otherwise you might be in the wrong directory.

```
/**
 * Name          :
 * Matric No.    :
 * PLab Acct.    :
 */

public class SantaClaus {

    public void run() {
        // treat this as your "main" method
    }

    public static void main(String[] args) {
        SantaClaus santa = new SantaClaus();
        santa.run();
    }
}
```

Notes

1. You should develop your program in the subdirectory **ex2** and use the skeleton file provided.
2. You must use **recursion** to solve this problem. Otherwise, you will get 0 for this problem.
3. This problem is worth **70%** of the total Practical Exam marks.
4. Please be reminded that the marking scheme is:

   Input : 10%
   Output : 10%
   Correctness : 50%

   Programming Style : 30% (awarded if you score **at least 20% from the above**):

   - Meaningful comments (pre- and post- conditions, comments inside the code): 10%
   - Modularity (modular programming, proper modifiers [public / private]): 10%
   - Proper Indentation: 5%
   - Meaningful Identifiers (for both method and variable names): 5%

   Compilation Error : Deduction of **50% of the total marks obtained**.