

## Busy Pirates

Filename: *pirates*

Time Limit: *3 seconds*

The group of pirates sailing under the flag of the Unruly Captains Federation (UCF for short) have become quite busy with all of their marauding. Stealing, pillaging, and burying the loot is quite a tedious task for the pirates, and they are seeking help in optimizing their daily routes so that they have more time for merriment.

Every day, a pirate visits  $n$  different locations to steal valuables. Once the buccaneer has finished their plundering, they travel to a set of  $n$  different secret islands to bury the treasure, one treasure per island. They would like to know what the minimum time they need to first visit all  $n$  treasure locations and then all  $n$  islands without revisiting any location. The pirates are very good at their jobs, so they take no time to pillage or bury treasure. Assuming they can start at any treasure and end at any island, help the pirates find the shortest time.

### **The Problem**

Determine the minimum amount of time it will take the pirates to start at any treasure location, visit the rest of the treasure locations, followed by burying each treasure at each of the secret islands, ending at any of the islands.

### **The Input**

The first line of input will consist of a single positive integer,  $c$  ( $c \leq 50$ ), representing the number of input cases to process. Each input case begins with a single positive integer  $n$  ( $n \leq 6$ ), representing the number of treasures to steal. The following  $2n$  lines each contain  $2n$  space separated integers, where the  $j^{\text{th}}$  integer on the  $i^{\text{th}}$  line is the time required to travel from location  $i$  to location  $j$ , where  $1 \leq i, j \leq n$  represent the treasure locations and  $n+1 \leq i, j \leq 2n$  represent islands to bury the treasures. The time required to travel in one direction may not be the same in the opposite direction as the wind can add additional travel time. Clearly the time required to travel from two identical locations is 0. All other non-trivial times will be positive and no more than  $10^5$ .

### **The Output**

For each input case, output the corresponding shortest distance to obtain and bury each of the treasures on a line by itself.

**Sample Input**

```
2
3
0 9 9 5 10 1
4 0 10 8 2 3
2 1 0 5 5 4
10 6 3 0 4 4
8 6 9 5 0 10
7 2 2 4 3 0
4
0 10 9 6 5 2 2 2
6 0 7 1 10 10 2 1
1 9 0 10 1 5 4 10
3 3 3 0 1 1 6 9
2 4 1 6 0 5 1 1
7 5 1 5 7 0 10 8
10 8 10 10 10 7 0 2
7 8 8 7 3 7 4 0
```

**Sample Output**

```
14
17
```