2015 Fall Computer Science I Program #7: A Game of Games
Please Consult WebCourses for the due date/time

All semester, you've been writing programs dealing with games! Naturally, Gene and Josiah has decided that they will go into business together and profit from having to grade all of the students' programs. They've created a Mega-Game App for hand-held devices!!! They've taken all of the games from the semester: Sudoku, Word Search, Mastermind, Maze, Risk, Boggle and a couple more and weaved them in to one ultimate game.

You start off at your "home base". From there, you can travel to any one of the "challenge houses". Due to various threats you face in traveling to the challenge houses, it takes some time for your character to successfully travel between these destinations. Luckily, you've played the game enough that you know exactly how long it takes to travel between any pair of locations of interest. (The locations of interest are the home base and all of the challenge houses.)

The challenge houses are only open for some fixed window of time after you've started playing the game. For example, the Boggle House might only be open from time t = 500 to t = 6,500, where t represents the number of seconds after the game has started. If you arrive at a challenge house during its open window, you can stay as long as it takes you to complete that game. Thus, it's okay to arrive at the Boggle House at time t = 6000 and spend 1000 seconds playing the game to complete it, before moving onto the next challenge house.

The goal of the game is to start at your home base, travel to each of the challenge houses, one by one, complete each challenge, and return to the home base. **Note: You are NOT allowed to visit any challenge house more than once, even if doing so reduces your travel time between another pair of challenge houses. (For example, if you have already completed challenge #1 and are currently at challenge #3 and want to do challenge #7 next, you must take the direct path from challenge #3 to challenge #7 instead of traveling from #3 to #1 and then #1 to #7, even if the latter is faster.)**

Unfortunately, the games themselves are quite challenging and you don't always finish them in a fixed amount of time. Gene and Josiah have set up a grand prize for the person who can play the game and win.

To give yourself the greatest chance of winning, you decide that you'll calculate the optimal order to visit each challenge house such that you leave yourself the maximal amount of time to complete each game. Thus, as long as you complete each game in that amount of time or less, based on your chosen ordering, you'll succeed in winning the game!!! Your goal will be to calculate the maximum amount of time you can leave yourself for each game such that you can still complete all the challenges. ***Namely, determine the largest time X such that, if you spent exactly X seconds on each challenge, you could still complete each challenge, under the constraints, given that you visit the challenges in the appropriate order.***

The Input (read from standard input)

The first line of input will contain a single positive integer, c ($c \leq 80$), representing the number of input cases. The first line of each input case will contain a single positive integer, n ($2 \leq n \leq 8$), the number of games to complete within the game. Let the home base be labeled p_0 and the challenge houses for each of the games be labeled p_1 through p_n. The i^{th} ($1 \leq i \leq n$) line of the next n lines will contain two space-separated positive integers, s_i ($0 \leq s_i \leq 10^5$) and e_i ($s_i < e_i \leq 10^5$), respectively, representing the starting time (in seconds) and ending time (in seconds), of the window of time for which challenge house p_i is open for you to come and play its game. The final n+1 lines of each test case will contain n+1 space-separated integers each. The $(j+1)^{th}$ of these values on the $(i+1)^{th}$ of these lines represents the amount of time it takes for your character to travel from location p_i to location p_j in the game. These values will all be non-negative integers in between 0 and 10,000, inclusive.

The Output (to standard out)

For each input case, output a single integer, on a line by itself, representing the maximum whole number of seconds you can leave yourself to play each game and finish all of the games inside the game. You are guaranteed that each input case will have an answer that is 1 second or greater; namely, the windows for each game and the travel times are such that there will always exist at least one way to leave yourself at least one second for each game to complete all the games.

Sample Input

```
2
2
10 30
5 15
0 5 10
3 0 50
5 2 0
3
1000 1200
1 5
500 510
0 1000 2 1000
1000 0 1000 1000
1000 1000 0 450
1000 5 1000 0
```

Sample Output

```
18
58
```

Use of Global Variables

Note: Just like previous assignments, you may use global variables if it aids the elegance and readability of your code without obfuscating it too much.

Deliverables

You must submit a single file, *gameofgames.c*, over WebCourses. Please use stdin, stdout.