

# NOI '08 P6 - Tournament Matching

**Time limit:** 4.5s    **Memory limit:** 512M

## National Olympiad in Informatics, China, 2008

With the Olympics right around the corner, students' interest in physical education is skyrocketing. As NOI 2008 approaches, the school is currently planning to host a ping-pong tournament. Being a very avid ping-pong enthusiast, this is the perfect opportunity for little Z to show off his skills. Thus, he quickly registers himself and begins his intensive preparation.

The tournament will follow a single-elimination format where only the winners of a round advance. There just happens to be  $n$  ( $n$  is a power of 2, so we may as well let  $n = 2^k$ ) participating students. After the first round,  $2^{k-1}$  students will be eliminated, and the other  $2^{k-1}$  will advance to the next round. After the second round, only  $2^{k-2}$  students will advance, and so on until after round  $k$ , where the champion and runner-up will be determined. Every contestant is assigned a number from 1 to  $n$ , where little Z is contestant number 1. The IDs of all students are unique, and all students will be assigned to  $n$  **slots**. Then, the tournament process will follow a structure like the one below.

Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8
Winner of Slots 1 and 2		Winner of Slots 3 and 4		Winner of Slots 5 and 6		Winner of Slots 7 and 8	
Winner of Slots 1, 2, 3, and 4				Winner of Slots 5, 6, 7, and 8			
Champion							

*The tournament bracket for when  $n = 8$ .*

To attract more students, the prizes of the tournament are very generous. The competitors being eliminated in round  $i$  are awarded  $a_i$  dollars, and the champion will receive the largest prize of  $a_{k+1}$  dollars. Needless to say, the prize amounts will satisfy  $a_1 < a_2 < \dots < a_{k+1}$ .

In the warm-up rounds prior to the real tournament, little Z was repeatedly defeated. After some careful analysis, he concluded that the key reason behind his failure was not an issue of his skill, but was because the students who defeated him just happened to clash with his playing stylistically. Little Z thought to himself, how nice would it be if these students could be avoided in the actual tournament!

Let's assume that we already know the probabilities of winning for all contestants in pairwise matches. The probability of contestant  $A$  winning against contestant  $B$  is  $P_{A,B}$  (where it is guaranteed that  $P_{A,B} + P_{B,A} = 1$ ). Little Z wishes to know for a certain tournament structure (by reassigning the slots for each contestant), what is the largest possible prize that he can obtain. Can you help little Z design a matching scheme for this tournament so that the expectations for his prize money is as large as possible?

## Input Specification

There are 10 test cases (`match1.in`) to (`match10.in`) that will be given to your program (through standard input).

They can be downloaded here for you to study: [match.zip](#)

The first line of each test case contains an integer from 1 to 10, specifying the test case number (`matchx.in` will have

(x) on the first line). The second line contains a single integer  $n$ , representing the total number of contestants. The data guarantees that  $k$  will be a nonnegative integer for  $2^k = n$ .

For the following  $n$  lines, each line contains  $n$  real numbers  $P_{i,j}$  between 0 and 1 inclusive, representing the probability of contestant  $i$  defeating contestant  $j$ . Each real value will be given to two decimal places. Especially take note of when  $P_{i,j} = 0.00$ .

For the following  $k + 1$  lines, each line contains an integer describing the prize money for a particular round. The  $i$ -th of these lines contain  $a_i$ .

## Output Specification

---

Output  $n$  lines, where line  $i$  represents the number of the contestant that is matched to slot  $i$ . **Little Z's number must be matched to the first slot.**

## Sample Input

---

```
0
4
0.00 0.70 0.60 0.80
0.30 0.00 0.60 0.40
0.40 0.40 0.00 0.70
0.20 0.60 0.30 0.00
1
2
3
```

## Sample Output

---

```
1
4
2
3
```

## Explanation

---

After the first round is over, the probability for contestant 1 (little Z) to advance is 80%, the probability for contestant 2 to advance is 60%, the probability for contestant 3 to advance is 40%, and the probability for contestant 4 to advance is 20%.

In the second round (the championship match), the probability of contestant 1 (little Z) winning the first two matches is  $80\% \times (60\% \times 70\% + 40\% \times 60\%) = 52.8\%$ . Thus, the probability of little Z losing his first match is  $P_1 = 1 - 0.8 = 0.2$ , the probability of him winning the first match but losing the second match is

$P_2 = 0.8 - 0.528 = 0.272$ , and the probability of him being the champion is  $P_3 = 0.528$ . From this, the expected value for his prize money is  $0.2 \times 1 + (0.8 - 0.528) \times 2 + 0.528 \times 3 = 2.328$ .

## Experimentation

---

We supply a tool [match\\_check.py](#) for you to test whether your solutions are accepted. The usage for this program is

```
match_check.py <input-file> <output-file>
```

When you execute `match_check.py`, the program will process your input and output files and print the result to standard output. Possible results of the execution include:

- `Abnormal termination`: An unknown error occurred.
- `Format error`: The output file is incorrectly formatted.
- `Not a permutation`: The output file is not a permutation of the integers from 1 to  $n$ .
- `OK. Your answer is x`: The output will be accepted.  $x$  will be the expected prize money.

## Scoring

---

Each test case will be graded independently.

For each test case, if your output is illegal (i.e. has invalid formatting, doesn't fit with the requirements, etc.) then you will receive a score of 0.

Otherwise, if the expected prize money for your output is `your_ans`, the prize money determined by the contest organizers' solution is `our_ans`, and a parameter for grading is  $d$ , then your score out of 10 for the test case will be:

- 12, if `your_ans` > `our_ans`.
- 1, if `your_ans` < `our_ans` ×  $d$ .
- Otherwise, your score will be:

$$\left\lfloor \frac{\text{your\_ans} - \text{our\_ans} \times d}{\text{our\_ans} - \text{our\_ans} \times d} \times 8 \right\rfloor + 2$$

## Hint

---

### "Expected Value"

The mathematical expected value is the most fundamental property of a random variable. It reflects the size of the average value of a random variable, also known as the expectation or mean. It is an extension of the simple arithmetic mean, or a weighted average of all the possible values. Ex. If a city has 100 000 families, 1 000 of which do not have children, 90 000 of which has one child, 6 000 of which have two children, and 3 000 of which have three children, then the number of children in a family within this city is a random variable that can take on the values of 0, 1, 2, or 3. The probability for 0 children is 0.01, of 1 child is 0.9, of 2 children is 0.06, and of 3 children is 0.03. Its expected value is  $0 \times 0.01 + 1 \times 0.9 + 2 \times 0.06 + 3 \times 0.03 = 1.11$ , and therefore the average family has 1.11 children.

To calculate the random variable for this problem, assume that little Z's chances of losing the first round is  $P_1$ , chances of winning the first round but losing the second is  $P_2$ , the chances of winning the first two rounds but losing the third is  $P_3$ , and so on. Then the expected value for little Z's prize money is:

$$P_1 \times a_1 + P_2 \times a_2 + \cdots + P_{k+1} \times a_{k+1}$$

Problem translated to English by [Alex](#).