ETH Zürich
Institute of Theoretical Computer Science
Prof. Angelika Steger, Prof. Emo Welzl
Dr. Johannes Lengler
Asier Mujika, Miloš Trujić

FS 2018

Algorithmen und Wahrscheinlichkeit
Programming Exercises 1

Exercise 1 – *Jackpot*

In a casino in Monte Carlo, you play at a very peculiar machine. The machine has n wheels, each with k possible values (not necessarily distinct). The wheels may be different from each other, that is it does not necessarily hold that every wheel has the same k values on it.

When you activate the machine, each wheels lands in one of its k possible values chosen uniformly at random and independently of all other wheels. You win a jackpot if the n chosen values form an increasing sequence $x_1 \leq x_2 \leq \ldots \leq x_n$ (the sequence does not need to be strictly increasing). You want to compute your chances of winning a jackpot.

Input The first line of the input file contains an integer $1 \leq t \leq 50$ denoting the number of test cases that follow. Each of the t test cases is described as follows.

- It starts with a line containing two integers `n k`, separated by space, denoting the number of wheels on the machine and the number of different values each wheel has, such that $1 \leq k$, $1 \leq n$, and $k^n \leq 2^{30}$.
- Then n lines follow: the i-th line contains k integers $\mathtt{a}_1 \ldots \mathtt{a}_k$, separated by space, denoting the values on the i-th wheel, such that $1 \leq a_j \leq 2^{20}$, for all $j \in [k]$.

Output Your algorithm should output the probability that activating the machine results in n values that are in increasing order. Each output value should be a real number rounded to *five* decimal places. You should round your result with the following piece of code:

```
DecimalFormat df = new DecimalFormat("0.0####");
df.setRoundingMode(RoundingMode.HALF_DOWN);
System.out.println(df.format(3.5)); // Replace 3.5 with your desired double
```

Points There are two groups of test cases, worth 100 points in total:

- For the first group of test cases, worth 50 points, you may assume $n \leq 2$.
- For the second group of test cases, worth 50 points, there are no additional assumptions.

Sample Input

```
2
2 2
1 2
2 2
2 3
1 2 3
1 2 3
```

Sample Output

```
1.0
0.66667
```