CCO '15 - Timpanist

Canadian Computing Olympiad: 2015 Day 2, Problem 2

Computer scientists don't often help percussionists, but today, that will change. Since we cannot help all percussionists at the same time, we focus on timpanists first. By way of terminology, the *timpani* is the plural of *timpano* and the player of the timpani is a *timpanist*.

A timpano is a large drum which can be tuned to a certain pitch, and a timpanist uses an ordered set of D timpani. On this occasion, they're playing a piece which has N notes. Note i occurs T_i seconds into the piece, and has pitch P_i. P_i is one of the following twelve notes:

$$\{F, F\#, G, G\#, A, A\#, B, C, C\#, D, D\#, E\}$$

At a given time, a timpano can only be used to play the pitch it is currently tuned to, and thus the timpanist can play a note i if and only if one of the timpani is tuned to pitch P_i at time T_i.

Every note in this piece is in the range of a single octave, from F up to E, which means that the above list of possible notes is in ascending order of pitch. In order to make your computation slightly easier, we will use integers from 1 to 12 to indicate these 12 tones:

1	2	3	4	5	6	7	8	9	10	11	12
F	F#	G	G#	A	A#	B	C	C#	D	D#	E

(i.e., `F` will be represented by 1, `F#` by 2, ..., `E` by 12). These are the only pitches to which timpani can be tuned.

Before the piece starts, the timpanist can freely tune each timpano to any pitch they'd like. However, during the piece, they may need to quickly retune them in between notes in order to be able to play the required pitches at the correct times. The drums are numbered from 1 to D. At every point in time, the drum $i+1$ must be kept tuned to a pitch higher than drum i. Retuning a single drum must be done in an uninterrupted interval of time, in which no notes are being played and no other drums are being retuned. Because this is not an easy process, the timpanist would like to give themselves as much time as possible. In particular, they'd like to maximize the amount of time they have for the fastest retuning they must perform within the piece.

Input Specification

The first line contains two integers, N and D, the number of notes to be played and the number of drums available to be played ($1 \leq N \leq 100; 1 \leq D \leq 4$). The next N lines each contain two integers T_i and P_i representing the time and pitch of the i^{th} note played ($0 \leq T_1 < T_2 < \ldots < T_{N-1} < T_N \leq 10^9; 1 \leq P_i \leq 12$ for $1 \leq i \leq N$).

For 60% of the marks for this problem, $N \leq 50$ and $D \leq 3$.

Output Specification

The output is one line containing one real number (rounded off to 2 decimal places), which is the maximum amount of

time (in seconds) that the timpanist can have for their fastest retuning, or `0.00` if no retunings are necessary.

Sample Input 1

```
7 1
100 1
120 3
130 5
140 6
150 8
165 10
170 12
```

Output for Sample Input 1

```
5.00
```

Explanation of Output for Sample Input 1

With just 1 drum, the timpanist must retune it after every note in order to play the following one.

With 2 drums, the answer would instead be 10.00 (achieved by leaving the higher drum tuned to pitch 12).

Sample Input 2

```
12 4
0 1
2 1
3 3
6 1
9 6
12 5
21 1
23 1
24 3
27 1
30 8
33 6
```

Output for Sample Input 2

```
4.50
```

Explanation of Output for Sample Input 2

The first 6 notes include only the 4 pitches 1, 3, 5, and 6. Similarly, the last 6 include only 1, 3, 6, 8.

The single optimal strategy involves pre-tuning the 4 drums to 1, 3, 5, and 6. After the sixth note, the timpanist can take 4.5 seconds to retune the highest drum to an 8, and then another 4.5 seconds to retune the second-highest drum to a 6, finishing just in time to play the seventh note.

Sample Input 3

```
2 4
40287 8
20338194 8
```

Output for Sample Input 3

```
0.00
```

Explanation of Output for Sample Input 3

This is a more typical timpani part, involving just one note, and thus no retuning.