

4911 Dazzling to the Store

It's a long way from home to the pet store. Recently retired roamer Earl delights in taking his dazzling dapple dachshunds, also known as *the girls*, to buy new treats, if only to escape often overbearing Opal, his dear wife of 100 years. Earl obeys all speed limits, but he wants to get to the store as fast as possible.

Earl has heard about navigation systems and is asking you to build him his very own to tell him when to accelerate, break, and coast, so that he and the girls can reach the store in no time. The navigation system assumes that acceleration and deceleration are constant. Using double precision floating point arithmetic, the system can always find a way where Earl won't even have to stop for zero seconds!



Input

The input file contains several test cases, each of them as described below.

The first line of input contains two positive integers, s and t , separated by white space. s is the number of streets which Earl has to traverse on his way to the store and t is the number of seconds which it takes his convertible to get from 0 to 100 miles per hour, or equivalently to slow back down.

Each of the remaining s lines of input describes a street and contains two positive integers, separated by white space, which define the length of the street in feet and the speed limit in miles per hour (one mile is 5280 feet), respectively.

Output

For each test case, the output must follow the description below.

The output contains floating point numbers, rounded to exactly three decimal places. The first line is the number of seconds it takes to reach the store.

Each of the remaining s lines of your output corresponds to one street line in the input and contains instructions for Earl, at most 3 of the following components in some order:

- the letter 'a' immediately followed by a positive number of seconds to accelerate,
- the letter 'b' immediately followed by a positive number of seconds to break,
- the letter 'c' immediately followed by a positive number of seconds to coast (neither accelerate nor break).

Sample Input

```
3 6
1200 45
800 30
900 35
```

Sample Output

```
56.468
a2.700 c16.082 b0.900
c18.182
a0.300 c16.204 b2.100
```