

Problem 1 – Compound interest

You really want a new TV, however you are a little short on money right now. You have a couple of options. You can **get a loan from a bank** or **get a loan from a friend**. Since you want to become the best programmer who ever lived, you decided to make a program to help you out.

Bank loans have **interest rate** and a **term (number of years)** you have until you are required to pay the money back). Assume the bank loan will be for more than one year and the interest will be accrued on a yearly basis. Use this formula to calculate the amount of money you will have to pay back - ' $FV = PV * (1 + i)^n$ '. Where '**FV**' (future value) is **the money owed at the end of the period**. '**PV**' (present value) is **the money you want to withdraw today**, '**i**' is **the interest rate** and '**n**' is **the term of the loan**. Your friend is a really nice dude and he will loan you the money, however he wants a percentage of the money in return.

You will be given the price of the TV, the term and yearly interest rate for the loan from the bank, and the percentage your friend will ask for.

Your task is to write a program that calculates **the best (cheapest) option to buy the TV**. If the options are the same, choose your friend's offer - you are a nice guy after all. Check the example to get a better understanding of the task.

Input

The input data should be read from the console. It consists of four input values, each at a separate line:

- The number **p** – **price of the TV**.
- The number **n** – number of **years you have until you must pay the bank back (term)**.
- The number **i** – the yearly interest rate for the bank's loan.
- The number **f** – interest rate for your friend's loan.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

- The output data must be printed on the console.
- On the only output line you must print **the best loan price to the second digit after the decimal mark and the lender separated by a single space**.

Constraints

- The number **y** is an integer in the range [0 ... 2 147 483 647].
- The numbers **p, i, f** are floating-point numbers in the range [0 ... 7.9×10^{28}].
- Allowed working time for your program: 0.25 seconds.
- Allowed memory: 16 MB.

Examples

Input	Output	Comments
2600 2 0.07 0.4	2976.74 Bank	2600 leva is needed. Bank loan = $2600 * (1 + 0.07)^2 = 2600 * 1.07^2 = 2600 * 1.1449 = 2976.74$. Friend loan = $2600 * (1 + 0.4) = 2600 * 1.4 = 3640$. $2976.74 < 3640$