

Help Bob

Bob loves Pizza but is always out of money. One day he reads in the newspapers that his favorite pizza restaurant, Alfredo's Pizza Restaurant, is running a competition: they will donate a big pizza to the first person who will tell them the lowest price per area that can be achieved by buying any of the pizzas at most once. "That task is easy!", thinks Bob, "For each pizza I just calculate the average price and the lowest quotient will be the answer."

Unfortunately the problem is a bit more complicated: with some pizzas Alberto gives out discount coupons for getting another pizza cheaper and even worse, those coupons can be combined. The pizzas have to be bought one after the other, and it is not possible to use a coupon to get a discount retrospectively for a pizza which has already been bought. Can you help Bob to become the first to solve this task and to get a pizza for free?

Input Specification

The input file contains several test cases. Each test case starts with a number m , the number of pizzas Alfredo offers. Input is terminated by $m=0$. Otherwise, $1 \leq m \leq 15$. Then follow m lines describing the pizzas. Each of those following lines describes pizza i ($1 \leq i \leq m$) and starts with 3 integer numbers p_i , a_i and n_i specifying the price of the pizza, its area and the number of discount coupons you get when buying it, $1 \leq p_i \leq 10000$, $1 \leq a_i \leq 10000$ and $0 \leq n_i < m$. Then follow n_i pairs of integer numbers $x_{i,j}$ and $y_{i,j}$ specifying the index $x_{i,j}$ ($1 \leq x_{i,j} \leq m$, $x_{i,j} \neq i$) of the pizza you get a discount coupon for and the discount in percentage terms $y_{i,j}$ ($1 \leq y_{i,j} \leq 50$) you get when buying pizza $x_{i,j}$. You may assume that for each i the values $x_{i,j}$ are pairwise distinct.

Output Specification

For each test case print one line containing the lowest price per area that can be achieved by buying any of the pizzas at most once. Round this number to 4 places after the decimal point. Note that you can combine an arbitrary number of discount coupons: for a pizza with price 10 and two rabatt coupons for that pizza with a 50 and a 20 on it, you would only have to pay $10 * 0.8 * 0.5 = 4$ monetary units.

Sample Input

```
1
80 30 0
2
200 100 1 2 50
200 100 0
5
100 100 2 3 50 2 50
100 100 1 4 50
100 100 1 2 40
600 600 1 5 10
1000 10 1 1 50
0
```

Sample Output

```
2.6667
1.5000
0.5333
```