ACM-ICPC Live Archive

5956 Change by Mass

Your team is working on software for automated check-out machines (such as are found in supermarkets and large home-improvement stores). The usual change-making process for these machines minimizes the number of coins dispensed when dispensing amounts under one dollar. However, a customer wants to change the logic in the machines to minimize the mass of the coins dispensed (to “lighten the load” for their customers). Minimizing the number of coins dispensed remains a secondary goal.

The name, value, and mass of each United States (No, our Canadian friends weren’t forgotten. We had to pick only one set of values, though.) coin with a value less than one dollar are as follows:

Name	Value	Mass
Cent (“penny”)	0.01	2.500 grams
Five cents (“nickel”)	0.05	5.000 grams
Dime	0.10	2.268 gram
Quarter dollar	0.25	5.670 gram
Half dollar	0.50	11.340 gram

Half dollar coins may or may not be available, as they do not circulate to the extent of the other coins.

Your team is to write a program that will, given an amount in cents, determine the coins to be dispensed that will minimize the total mass of the dispensed coins, and secondarily minimize the number of coins dispensed.

Input

Input to your program will be a series of change requests, one per line. Each change request will consist of the amount to be dispensed (integer cents, in the range 1–99 inclusive) starting in the first column, followed by a single space and the number of half-dollar coins in the machine (as an integer ≥ 0).

Input will be terminated by the end-of-file.

Output

For each request, your program should produce a line containing the coins to be dispensed. The output line is to show the denominations being dispensed in decreasing order, in the format ‘*n*x*d*’ (meaning *n* coins of denomination *d* are to be dispensed). For example, 35 cents would be dispensed as ‘`1x25 1x10`’. Denomination entries are to be separated from each other by single spaces, and no leading or trailing whitespace is to appear on an output line.

Sample Input

```
35 1
1 0
18 3
60 0
```

Sample Output

```
1x25 1x10
1x1
```

```
1x10 1x5 3x1
2x25 1x10
```