

Exercise: While Loops

1. Dishwasher

John works in a restaurant and is responsible for loading the dishwasher at the end of the day.

Your task is to write a program that calculates whether the purchased quantity of bottles of dishwasher detergent is enough to wash a certain

number of kitchenware.

It is known that each bottle contains 750 ml of detergent.

For each plate 5 ml is needed, and for each pot 15 ml is needed.

Keep in mind that on every third load, the dishwasher is loaded only with pots, and at other times with plates.

Until you receive the command “END” you will continue to receive the number of kitchenware that needs to be washed.

Input

Read from the console:

- Number of bottles of detergent that will be used for washing plates - integer in range [1...10]
- On each subsequent line, until the command “End” or the detergent has run out, the number of kitchenware that needs to be washed - integer in range [1...100]

Output

- In case the amount of detergent was sufficient for the washing of the kitchenware, print three lines of output:
 - “Detergent was enough!”
 - “{Number of clean plates} plates and {number of clean pots} pots were washed.”
 - “Leftover detergent {amount of remaining detergent} ml.”
- If the amount of detergent was not sufficient for the washing of the kitchenware, print the following line:
 - “Not enough detergent, {required quantity of detergent} ml more necessary!”

Example

Input	Output
2	Detergent was enough!
53	118 plates and 55 pots were washed.
65	Leftover detergent 85 ml.
55	
End	

Comments

- Detergent quantity: $2 * 750 = 1500$ ml
- 53 plates are loaded: $53 * 5 = 265$ ml,
 - $1500 - 265 = 1235$ ml. (leftover)
- 65 plates are loaded: $65 * 5 = 325$ ml
 - $1235 - 325 = 910$ ml (leftover)
- 55 pots are loaded: $55 * 15 = 825$ ml
 - $910 - 825 = 85$ ml (leftover)
- We receive the command "End", therefore the corresponding message is printed: 118 plates and 55 pots were washed.
 - number of plates = $53 + 65 = 118$
 - number of pots = 55

2. Report System

At a charity event, payments for the purchased products are always alternated: cash payment and card payment.

First payment method is always cash.

The following payment rules have been established:

- If the product exceeds 100 dollars, it cannot be paid in cash
- If the product is priced under 10 dollars, it cannot be paid by credit card

The program ends either after we receive the command "End", or after the funds are collected.

Input

Read from the console:

- The amount expected to be collected from sales - integer in range [1...10000]
On each subsequent line, until the "End" command is received or until the necessary funds are collected:
 - The prices of items to be purchased - integer in range [1...500]

Output

Print on the console:

- In case of a successful transaction: "Product sold!"
- In case of unsuccessful transaction: "Error in transaction!"
- If the sum of all purchased products exceeds or reaches the expected amount, the program must be completed and two lines are printed to the console:
 - "Average CS: {average payment in person's cash}"
 - "Average CC: {average card payment per person}"Payments must be formatted to the second digit after the decimal point.

- When the “End” command is received, one line is written:
 - “Failed to collect required money for charity.”

Example

Input	Output
500	Error in transaction!
120	Error in transaction!
8	Product sold!
63	Product sold!
256	Product sold!
78	Product sold!
317	Average CS: 70.50
	Average CC: 286.50

Comments

- The condition is rotated first in cash payment, then through credit card
- $120 > 100$ transaction is rejected
- $8 < 10$ transaction is rejected
- $63 \leq 100 \Rightarrow$ the transaction was successful
- $256 \geq 10 \Rightarrow$ the transaction was successful
- $78 \leq 100 \Rightarrow$ the transaction was successful
- $317 \geq 10 \Rightarrow$ the transaction was successful
- Total amount collected: $63 + 256 + 78 + 317 = 714$
- $714 \geq 500$
- Total cash: $63 + 78 = 141$; Average cash: $141/2 = 70.50$
- Total credit cards: $256 + 317 = 573$; Average credit cards: $573/2 = 286.50$

3. Graduation

Description

Write a program that calculates the average grade of a student from his entire education.

Input

- On the first line you will receive the name of the student, and on each following line his annual grades. - The student passes to upper class, if his annual grade is 4.00 or greater.

Output

- If his grade is less than 4.00, he has to repeat the class. You have to print:
 - “repeat class”
- If the student graduates 12th class, you have to print:
 - “{student name} graduated. Average grade: {average grade from his entire education}”

The grade should be formatted to the second decimal point.

Example

Input	Output
John	John graduated. Average grade: 5.37
4	
5.5	
6	
5.43	
4.5	
6	
5.55	
5	
6	
6	
5.43	
5	

4. Old Books

Andreea goes to her hometown after being abroad for a long time.

When she comes home, she sees her grandmother’s library and remembers her favourite book.

Help Andreea write a program in which Andreea enters the name of the book she’s searching for (String) and the capacity of the library (integer).

Until Andreea finds her favourite book or doesn't check all books in the library, the program has to read every time the name of the next book on a separate line.

Input

- First line of input is the name of the book Andreea's searching for - string
- Second line is the capacity of the library - integer
- On every next line - name of book from library - string

Output

- If Andreea does not find the book, print two lines:
 - "The book you search is not here!"
 - "You checked {count} books."
- If Andreea finds the book, print a single line:
 - "You checked {count} books and found it."

Example

Input	Output
Troy	You checked 2 books and found it.
8	
Stronger	
Life Style	
Troy	

Comments

- Andreea is searching for a book with the name "Troy", and the library's capacity is 8 books.
- The first book is "Stronger", the second one is "Life Style", the third one is desired - "Troy" and the program ends.

Example

Input	Output
The Spot	The book you search is not here!
4	You checked 4 books.
Hunger Games	
Harry Potter	

Toronto	
Spotify	

Comments

- Andreea is searching for a book with the name "The Spot". The library contains 4 books.
- The first book is "Hunger Games", the second - "Harry Potter", the third - "Toronto", the fourth - "Spotify"
- Since there aren't other books in the library, reading book names has stopped. Andreea didn't find the book.

5. Exam Preparation

Write a program in which Martin solves problems for exams until he receives a message from his trainer: "Enough".

Every time he solves a problem, he gets a grade. The program should end when Martin receives either "Enough" command, or obtain the number of poor grades.

A poor grade is a grade less or equal to 4.00.

Input

- On the first line – number of poor grades – integer in range [1...5]
- After that repeatedly two lines:
 - Name of a problem - text
 - Grade - integer in range [2...6]

Output

- If Martin reaches "Enough" command, print 3 lines:
 - "Average score: {average grade}"
 - "Number of problems: {number of ALL problems}"
 - "Last problem: {last problem's name}"
- If he gets the specified number of poor grades:
 - "You need a break, {number poor grades} poor grades."

The average grade should be formatted to the second decimal point.

Example

Input	Output
3	Average score: 5.25
Money	Number of problems: 4
6	Last problem: Bus
Story	

4	
Spring Time	
5	
Bus	
6	
Enough	

Comments

- The number of allowed poor grades is 3.
- First problem's name is Money, Martin's grade is 6.
- Second problem - Story, grade - 4.
- Third problem - SpringTime, grade - 5.
- Fourth problem - Bus, grade - 6.
- Next command is Enough, the program ends.
- Average grade: $21 / 4 = 5.25$
- Number of solved problems: 4
- Last problem: Bus

Example

Input	Output
2	You need a break, 2 poor grades.
Income	3
Game Info	6
Best Player	4

Comments

- The number of allowed poor grades is 2.
- The first problem's name is Income, Martin' grade is 3.
- Second problem - Game Info, grade - 6.
- Third problem - Best Player, grade - 4.
- Martin reaches the number of allowed poor grades, it is time for break.

6. Walking

Gaby wants to start a healthy life and she sets a goal to walk 10000 steps a day.

Write a program that reads from the console how many steps Gaby walks every time she is out and when she reaches the set goal, print a message:

- "Goal reached! Good job!".

In case she wants to go home before reaching the goal, she will enter a "Going home" command and the steps she had walked while she was going home.

After that, if she didn't manage to reach her goal, you have to print the following message on the console:

- "{difference in steps} more steps to reach goal."

Example

Input	Output
1000	Goal reached! Good job!
1500	
2000	
6500	

Input	Output
1500	2500 more steps to reach the goal.
300	
2500	
3000	
Going home	
200	

7. Stream Of Letters

Write a program that reads a hidden message in a sequence of symbols.

You will receive each of them on a single line until the "End" command.

The words are initiated from the letters in the order of their reading.

Symbols which are not Latin letters should be ignored.

The words, hidden in the stream, are separated from a secret command by three letters - "c", "o" and "n".

When you first receive one of these letters, you have to mark it as visited, but it is not saved in the word.

Every time you receive the same letter, it is saved normally in the word.

After you have found all three symbols from the command, you have to print the word and a space " ".

A new word is started the same way, waiting for the secret command to be printed.

Input

- Read a sequence of lines with a single symbol each, until you receive the "End" command

Output

- Print on the console every word after the secret command followed by space

Example

Input	Output
H	Hello
n	
e	
l	
l	
o	
o	
c	
End	

Comments

- "H", "n", "e", "l", "l", "o", "o", "c" are all read letters.
- First we read "H" and we add it to the word. The next symbol is "n". It's part of the command and we do not add it to the word as we meet it for the first time.
- The next symbols are "e", "l", "l" and we add them to the word. We read "o" and we mark it as visited, but again we do not add it to the word. The next letter is "o" again and it's added. The next is "c" and all three symbols for the secret command are available.
- We print "Hello " and we will receive the "End" command and then the program ends. The result is "Hello ".