

That's that Me Espresso (espresso)

Sabrina has been working late as her usual, so today she is sitting at the bar having a late breakfast while watching a young and inexperienced barista handle a flood of coffee orders. The coffee machine has a single group head (the part where water comes out, where a portafilter can be attached or removed) and a button to start brewing coffee.

The machine supports two types of portafilters:

- A single-shot portafilter, which produces one shot of coffee.
- A double-shot portafilter, which produces two shots of coffee at once.



Figure 1: An example of double-shot portafilter.

The double-shot portafilter can be used with a single cup as in the picture, or also with two cups placed next to each other so that both cups will receive a single shot of coffee.

On the bar's menu there are six types of coffee:

- **Espresso:** requires 1 shot of coffee.
- **Espresso doppio:** requires 2 shots of coffee.
- **Cappuccino:** requires 1 shot of coffee.
- **Affogato:** requires 2 shots of coffee.
- **Dead eye:** requires 3 shots of coffee.
- **Irish coffee:** requires 2 shots of coffee.

In principle, each of these can be prepared with a single push of the “brew” button, except for the “Dead eye” which will always require at least two pushes since there is no portafilter that produces 3 shots at once.

The barista must fulfill the orders from the first to the last, in that order: it would make a customer angry to see the barista prepare an order for someone that arrived after them. Still, the barista is free to combine “adjacent” orders however they want to optimize their throughput. For example it might make sense in some cases to prepare an Espresso and a Cappuccino together, or to prepare a Cappuccino while at the same time starting to prepare the next order (for example, an Affogato) by pouring the first necessary shot for it.

Sabrina, being a problem-solver, decides to create an algorithm to help the barista press the “brew” button the **minimum number of times** while fulfilling all the coffee orders.

Input

The first line contains one integer N , the number of requests the barista received at a certain moment. Each of the next N lines contain a string S_i indicating the type of coffee the i -th client requested to the barista.

Output

You need to write a single line containing the minimum number of times the barista should press the “brew” button to fulfill all orders.

Constraints

- $1 \leq N \leq 20\,000$.
- S_i is always one of the following strings: `espresso`, `espresso-doppio`, `cappuccino`, `affogato`, `dead-eye`, `irish-coffee`.

Examples

input	output
2 espresso cappuccino	1
3 espresso espresso-doppio cappuccino	2
2 dead-eye irish-coffee	3