It is amazing how many different games can be played with a simple deck of playing cards. In this problem you are playing the game 'Get Straight' in your local casino and you will write a program that calculates the optimal strategy for it.

A standard deck consists of 52 playing cards, each having a value and a suit. Possible values are: Ace, Two, Three, Four, Five, Six, Seven, Eight, Nine, Ten, Jack, Queen and King; possible suits are: Spades, Hearts, Diamonds and Clubs. In this problem we'll use one character abbreviations for values (A, 2, 3, 4, 5, 6, 7, 8, 9, T, J, Q and K) and suits (S, H, D and C), so we can identify each playing card with a two-character string value+suit (AH means Ace of Hearts, 7S means Seven of Spades).

In the game of Get Straight we are only concerned with the value of the cards, and more specifically with getting runs of consecutive cards. Cards that are listed next to each other in the above list are considered consecutive, but also a King and an Ace are considered consecutive. So both 7, 8, 9, T and Q, K, A, 2 are considered runs of four cards (independend of the suits of the individual cards).

You start the game by paying one dollar (or euro, yen, taka, whatever) to the dealer, who gives you 5 cards at random. Now you can either stay (do nothing) or exchange one of your cards. If you exchange, you discard the selected card by placing it face down in front of the dealer and pay another dollar. The dealer gives you one of the 47 remaining cards at random so you have 5 cards again. Now you order your cards to form one or more runs and form a so called combination. If a combination is winning, the dealer pays you a certain amount. Possible combinations are:

Name	Runs	Payment
Straight	run of 5	100
Invite-the-Neighbours	run of 4	10
Bed-and-Breakfast	run of 3 + run of 2	5
Menage-a-Trois	Run of 3	3
Double Dutch	Run of 2 + run of 2	1
Dutch	Run of 2	
Dummy	no runs	

Only the highest possible combination pays, so if you have KS, AH, 2S, 7D and 8D, you can form a Bed-and-Breakfast, a Menage-a-Trois, a Double Dutch , a Dutch or a Dummy, but only the Bed-and-Breakfast pays you 5 dollars. A run consists of consecutive cards of different values, so 6S, 6H, 7S, 8D and 8H contains a run of 3, not a run of 5, and is evaluated as a Menage-a-Trois. However, 6S, 6H, 7S, 7D and 8H, contain both a run of 3 and a separate run of 2, and is therefor evaluated as a Bed-and-Breakfast.

Given an initial deal of five cards, your program has to decide the best strategy: either stay or exchange one of your cards. The best strategy is the strategy that gives you the highest expected earning. An example:

If you initially have 2H, 3S, 5D, 6C and TC, your highest combination is Double Dutch, which earns you nothing (you payed one dollar and you get one back). If you decide to exchange the Ten of Clubs, you have

- a 4/47 chance of getting a Straight;

- a 8/47 chance of getting a Bed-and-Breakfast;

- a 35/47 chance of being stuck with your Double Dutch.

This gives you an expected earning of: $4/47 * 100 + 8/47 * 5 + 35/47 * 1 - 2 = 381/47 = 8.11$ dollar, so exchanging the Ten of Clubs is a better strategy then staying with your initial cards.

Input

The input consists of about 1000 lines, each containing one initial deal of five cards. The cards are printed in random order in the two-character format defined above, with one space separating them. There are no leading or trailing spaces, so each line is exactly 14 characters long. Input is terminated by a line which contains a single #. This line should not be processed.

Output

For each line of input except the last one, produce one line of output containing the best strategy for that deal. If the best strategy is to stay with the initial cards, print the word 'Stay'. If the best strategy is to exchange, print the line 'Exchange XY', where XY is the two-charcter string of the card you want to discard, in the format defined above.

If there is more than one optimal strategy for a given deal, print any one of them. A special judge will evaluate your output. Don't use extra spaces or blank lines, and don't print the surrounding quotes.

Sample Input

```
TH JH QH KH AH
2H TC 5D 3S 6C
2S 5S 8D JC JH
#
```

Sample Output

```
Stay
Exchange TC
Stay
```