First Set

Description

First set is an important information in building Parser, when it conform many rule, it can use to decide which route to expend. The definition of First set is:

1. If a Nonterminal A, it rule is A → $\alpha_1 | \alpha_2 | \ldots | \alpha_n$, then

   $\mathrm{First(A)} = First(\alpha_1) \cup First(\alpha_2) \cup \ldots \cup First(\alpha_n)$
2. If a Right Hand Side is $\beta_1\ \beta_2 \ldots \beta_n$, then

   $\mathrm{First}(\beta_1) = First(\beta_1\ \beta_2 \ldots\ \beta_n)$
3. Hence, if $\mathrm{First}(\beta_1) = \varepsilon$, then $\mathrm{First}(\beta_2) = First(\beta_1\ \beta_2 \ldots\ \beta_n)$, and so on。
4. Hence, if $\mathrm{First}(\beta_n) = \varepsilon$，則 $\mathrm{First}(\beta_n) = First(\beta_1\ \beta_2 \ldots\ \beta_n) = \varepsilon$。

Please according to the rules, calculate the First set of Grammar.

Input Format

Each line is a Nonterminal in begin, and follow the rule separate by a blank, then end by ‘\n’.
Difference rules will separate by ‘|’.
When each line input finish, it will input “END_OF_GRAMMAR” to mean it’s end.
Nonterminal and Terminal are one letter.

Allowed token is:

- One uppercase letter ”A-Z” is onterminal.
- One lower case letter “a-z” is Terminal.
- ‘;’ is empty string as ‘ε’.
- ‘$’ is ‘EOF’.

※The all cases are legitimate.

※The all cases are not recursive.

Output Format

Order Nonterminal and First Set by ASCII from big to small.
Output each line Nonterminal in begin, and follow the First Set by a blank, then end by ‘\n’.
E.g. First Set of A is “abc;”, then print “A ;abc”. Print “END_OF_FIRST” at last line, then end by ‘\n’.

Sample Input

```
S ABC
A a|Cb|;
B C|dA|;
C e|f|;
END_OF_GRAMMAR
```

Sample Output

```
A ;abef
B ;def
C ;ef
S ;abdef
END_OF_FIRST
```

Sample Input

```
S AC$
C c|;
A aBCd|BQ
B bB|;
Q q|;
END_OF_GRAMMAR
```

Sample Output

```
A ;abq
B ;b
C ;c
Q ;q
S $abcq
END_OF_FIRST
```

Sample Input

```
S aBDh
B cC
C bC|;
D EF
E g|;
F f|;
END_OF_GRAMMAR
```

Sample Output

```
B c
C ;b
D ;fg
E ;g
F ;f
S a
END_OF_FIRST
```

Sample Input

```
S AaAb|BbBa
A ;
B ;
END_OF_GRAMMAR
```

Sample Output

```
A ;
B ;
S ab
END_OF_FIRST
```