B. 01 Game

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Alica and Bob are playing a game.

Initially they have a binary string s consisting of only characters `0` and `1`.

Alice and Bob make alternating moves: Alice makes the first move, Bob makes the second move, Alice makes the third one, and so on. During each move, the current player must choose two **different adjacent** characters of string s and delete them. For example, if $s = 1011001$ then the following moves are possible:

1. delete s_1 and s_2: $\mathbf{10}11001 \rightarrow 11001$;
2. delete s_2 and s_3: $1\mathbf{01}1001 \rightarrow 11001$;
3. delete s_4 and s_5: $1011\mathbf{10}1 \rightarrow 10101$;
4. delete s_6 and s_7: $101100\mathbf{1} \rightarrow 10110$.

If a player can't make any move, they lose. Both players play optimally. You have to determine if Alice can win.

Input

First line contains one integer t ($1 \leq t \leq 1000$) — the number of test cases.

Only line of each test case contains one string s ($1 \leq |s| \leq 100$), consisting of only characters `0` and `1`.

Output

For each test case print answer in the single line.

If Alice can win print `DA` (YES in Russian) in any register. Otherwise print `NET` (NO in Russian) in any register.

Example

input

```
3
01
1111
0011
```

output

```
DA
NET
NET
```

Note

In the first test case after Alice's move string s become empty and Bob can not make any move.

In the second test case Alice can not make any move initially.

In the third test case after Alice's move string s turn into 01. Then, after Bob's move string s become empty and Alice can not make any move.