Software Bugs

`bugs.c | bugs.C | bugs.java | bugs.p`

The biggest problem for all software developers are bugs. You definitely know the situation when a user calls to say ”I've found a bug in your program”. Once you have found and removed the bug, another one appears immediately. It is a hard and never-ending process.

Recently, there appeared a promising open-source initiative called “bug-preprocessor”. The bug-preprocessor is a program able to find all bugs in your source code and mark them, so they are relatively easy to be removed. Your task is to write a program that will remove all marked bugs from the preprocessed source code.

Input Specification

The input contains a text representing the preprocessed source code, an unspecified number of lines of text, some of them may be empty. Bugs are represented by a case-sensitive string “`BUG`”. The text is terminated by the end of file. No line in the input will be longer than 100 characters.

Output Specification

Your program must remove all of the bugs from the input and print a text that does not contain any `BUG` strings. Nothing else than bugs may be removed, not even spaces.

Sample Input

```
print "No bugs here..."

void hello() {
BUGBUG
  printfBUG("Hello, world!\n");
}

wriBUGBUGtelBUGn("Hello B-U-G");
```

Output for Sample Input

```
print "No bugs here..."

void hello() {

  printf("Hello, world!\n");
}

writeln("Hello B-U-G");
```