ICPC Pacific Northwest Region Practice Contest

November 9, 2019
For the Pacific Northwest Regional Contest, the important versions are as follows:

- Linux: Ubuntu 18.04 LTS (64-bit)
- Desktop: GNOME
- Editors: vi/vim, gvim, emacs, gedit, geany, kate
- Java: OpenJDK 11
- C, C++: gcc 7.4
- Python: Pypy7.7.1 implementing Python 3.6
- Kotlin: 1.3.50
- C#: Mono 7.8.4
- Eclipse 4.13
- IntelliJ 2019.2.2
- Code::Blocks: 17.12-1
- VSCode: 1.39.2

This is similar to World Finals, except at the present time World Finals will not support C#, will be using CPython3 instead of PyPy3, and will not support VSCode.
Problem A
Title Cost

You are head of operations at FlixNet, and you are responsible for transmitting just the title of movies to consumers. All movie titles on FlixNet are a single word and consist only of alphabetic characters.

The aggregate cost for transmitting a particular title is just equal to the number of letters in that title, except that your interweb service preventer has agreed to cap that cost at a given (floating point) value.

Given the title of a movie and the value of that cap, calculate your cost to transmit that movie title to your consumers.

Input

The input is a single line containing the name of the movie \( s \) and the cap on the cost \( c \). The name of the movie will contain between 1 and 30 characters, inclusive, and consist solely of uppercase or lowercase letters. The cost will be a floating point value with \( 0 \leq c \leq 100 \).

Output

Output a single floating point value, which is the minimum of the length of the title and the cap on the cost. Your solution will be judged correct if it is within 1e-8 absolute or relative error of the correct answer.

<table>
<thead>
<tr>
<th>Sample Input 1</th>
<th>Sample Output 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>GoneWithTheWind 13.341</td>
<td>13.341</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sample Input 2</th>
<th>Sample Output 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gigi 93.7</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sample Input 3</th>
<th>Sample Output 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PieHard 3.14159265358979323846</td>
<td>3.14159265358979</td>
</tr>
</tbody>
</table>

Solutions

C++. Note the need to use `setprecision` on `cout`.

```cpp
#include <iostream>
#include <iomanip>
#include <string>
```
int main() {
    std::string s;
    double f;
    std::cin >> s >> f;
    auto r = std::min((double)s.size(), f);
    std::cout << std::setprecision(15) << r << std::endl;
}
Java.

```java
import java.util.*;
public class Practice {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String s = sc.next();
        double f = sc.nextDouble();
        double r = Math.min(s.length(), f);
        System.out.println(r);
    }
}
```

Python. We are only supporting Python 3 this year.

```python
toks = input().split(" ")
f = float(toks[1])
r = min(len(toks[0]), f)
print(r)
```

C. Note the `printf` format specifier.

```c
#include <stdio.h>
#include <string.h>
int main() {
    char s[2048+1] ;
    double f ;
    scanf("%2048s %lg", s, &f);
    double r = (strlen(s) > f ? f : strlen(s));
    printf("%.17lg\n", r);
}
```

C#. Standard floating point output format works fine.

```csharp
using System ;
public class Practice {
    public static void Main(string[] args) {
        String[] input = Console.ReadLine().Split(' ');
        double f = Convert.ToDouble(input[1]);
        double r = (input[0].Length > f ? f : input[0].Length);
        Console.WriteLine(String.Format("{0}", r));
    }
}
```

Kotlin. Standard floating point output format works fine.
import java.util.*
fun main(args:Array<String>) = with(Scanner(System.`in`)) {
    val s = next()
    val f = nextDouble()
    val r = if (s.length > f) f else 0.0 + s.length
    println(String.format("%.17g", r))
}