You have to program several functions. In each case, few lines of code are enough. \textit{Warning:} do not use the \texttt{split} string method.

1. Write a function \texttt{um\_count(s)} that given an string \textit{s} returns the number of times character \texttt{u} is followed by character \texttt{m} in string \textit{s}.

2. Write an integer function \texttt{word\_count(s)} that returns the number of words in string \textit{s}. We assume all characters of \textit{s} are letters and spaces.

3. Write a function \texttt{kth\_word(s, k)} that given a string \textit{s} and an integer \textit{k} \(\geq 1\) returns the \textit{kth} word in string \textit{s}. If \textit{s} has less than \textit{k} words it returns the empty string. We assume all characters of \textit{s} are letters and spaces.

4. Write a function \texttt{suc\_word(s)} that given a string \textit{s} and returns the first word in string \textit{s} that has some uppercase letter. If all the letters in \textit{s} are lowercase it returns the empty string. We assume all characters of \textit{s} are letters and spaces.

5. Write a function \texttt{drawA(n)} that given an odd integer \textit{n} \(\geq 3\) prints a letter \texttt{A} of size \textit{n} formed with symbol \texttt{*}.

\textbf{Scoring}

Every function counts 20 points.

\textbf{Sample session}

\begin{verbatim}
>>> um_count("Qui invenit amicum invenit thesauruM")
1
>>> word_count("Alea iacta est")
3
>>> kth_word("Alea iacta est", 3)
est
>>> suc_word("qui invenit amicum invenit thesauruM")
amCum
>>> drawA(5)
  * *
  *****
  * *
  * *
\end{verbatim}

\textbf{Problem information}

Author : Jorge Castro
Generation : 2016-09-20 09:24:03

http://www.jutge.org