

Assignment 5 - Ancient Fax Machine

CmpE 250, Data Structures and Algorithms, Fall 2018

Instructor: A. T. Cemgil
TA: Özlem Şimşek
SA: Burak Çetin, M. Emin AYAR

Due: 27/12/2018, 23:59 Sharp

1 Description

Mahir has a really old fax machine. He is using this fax machine to communicate with his friends and family. But one day this machine starts glitching. As a result all of the space characters in the messages are skipped by the machine. As an example, if a friend sends Mahir the message "the cake is a lie", the fax machine will print it as "thecakeisalie". After some time this starts causing him problems since he cannot know where words start or end for sure. The message "dragonfly" might mean "dragonfly", "dragon fly" or "drag on fly". After struggling with these messages Mahir decides to write a program that can calculate how many ways are there to split a message printed by this fax machine into words.

2 Input/Output

You will be given a message and a list of possible words that can appear in the message. You are asked to print out in how many ways that a given message can be interpreted.

2.1 Input Format

The first line of the input file contains the received message as a string. Second line of the input file is an integer D , specifying the number of words in Mahir's dictionary. Each of the following D lines contain a string, a distinct word in the dictionary. **The dictionary does not have to be in the alphabetical order. A word in the dictionary might not appear in the message.**

2.2 Output Format

Your program should calculate and print out the number of possible splittings of this message into words, using the words the given dictionary. **This number can be really big, so you should calculate it in modulo $10^9 + 7$. One of the biggest prime numbers that can fit into a 32 bit integer.**

Hint: Whenever you perform an arithmetic operation, use *long long int* data type and take the modulo afterwards. This will prevent arithmetic overflow and ensure you get the right answer.

2.3 Example Input 1

```
hellomahir
3
hello
world
mahir
```

2.4 Example Output 2

1

This message can only be interpreted as "hello mahir".

2.5 Example Input 2

```
thisisoverusedcarsalesman
14
ale
ales
car
cars
is
man
over
overused
ruse
sale
sales
salesman
this
used
```

2.6 Example Output 2

6

All possible ways to interpret this message:

```
this is overused car salesman
this is overused cars ales man
this is over used car sales man
this is over used car salesman
this is over used cars ales man
```

2.7 Example Input 3

[illegible]

2.8 Example Output 2

147483634

Since this message can be split on each possible location, the number of possible interpretations of this message is $2^{31} = 2,147,483,648$. 2,147,483,648 modulo 1,000,000,009 is 147,483,634.

3 Grading

Grading of this project is based on the success of your code in test cases. Your final score will be the sum of points from each test case. Each test case will have equal weight. Maximum score is 100.

4 Implementation Details

1. Variable limits are as follows:
 - $1 \leq \text{Length of the received message} \leq 1,000$
 - $1 \leq \text{Number of words in the dictionary} \leq 1,000$
 - $1 \leq \text{Length of each word in the dictionary} \leq 1,000$
 - All of the characters in each string will be a lower case letter.
2. Execution time limit is 1 seconds. **If your code runs more than 1 seconds on a test case you will get 0 points from that test case.** This project is all about optimizing your solution. Before starting with your code, consider the time complexity of your algorithm. If you can realize your algorithm is a slow one before you start coding, it can save you time.
3. Your program will be compiled with **`cmake CMakeLists.txt && make`** command and executed with **`./project5 inputFile outputFile`** command.

5 Warnings

1. Make sure an executable is created after executing **`cmake CMakeLists.txt && make`** commands. Otherwise your code will fail in grading.