Here is a summary of a topcoder problem:

- A DNA sequence is a string composed of the characters 'A', 'C', 'G' and 'T', in any order, with repetitions allowed.
- You are given a sequence in the parameter **sequence**, whose size is between 1 and 100,000.
- Return the smallest DNA sequence that is not a substring of **sequence**. If there are multiple answers, return any of them.

Examples

```
# Sequence              A correct answer
- --------------------  ----------------
0 "AGGTCTA"             "AC"  ("AA", "GA" and "TG" will work too, as will others)
1 "AGACGACGGAGAACGA"    "T"
2 "A"                   "C"
3 "AAGATACACCGGCTTCGTG" "CAT"
```

Question 1:	Question 2:
Which type of enumeration can solve this problem. Just put the letter of the answer into the TurningPoint text box: • **A**: Div-Mod Enumeration. • **B**: Power Set Enumeration. • **C**: n-choose-k.	If the length of **sequence** is n, and the length of the answer is m, then what is the running time of your enumeration (not the whole program, just the enumeration)?

Answers to today's clicker questions

Question 1: You want to enumerate all strings of length m composed of the characters 'A', 'C', 'G' and 'T'. This is a Div-Mod enumeration -- the answer is **A**.

Question 2: There are 4^m strings in this enumeration.