Schwerer Theorietag in Sarnen

Swiss Olympiad in Informatics

13. Februar 2019

Anleitung

Alle 4 Aufgaben werden auf die Gruppenmitglieder aufgeteilt. (Uniform zufällig.)

Bewertung

Die Präsentation sollte folgende Punkte enthalten:

1. Beschreibung der Idee und Erklärung des Algorithmus.
2. Beweis, wieso der Algorithmus korrekt ist.
3. Analyse der asymptotischen Laufzeit und des asymptotischen Speicherverbrauchs.

Die Bewertung beinhaltet hauptsächlich die Laufzeit und Korrektheit des Algorithmus. Die Qualität der Präsentation wird ebenfalls berücksichtigt.

Du darfst vorgeschriebenen Code zeigen, aber keine sonstigen Notizen. Zur Erklärung steht dir ein leeres Blatt Papier zur Verfügung.

Pro Aufgabe solltest du nicht länger als 10 Minuten benötigen.

Technische Hinweise

Es steht nicht, nach welchen Kriterien du das Programm optimieren musst. Es sollte aber in allen Fällen klar sein, welcher Parameter gemeint ist (z.B. Zahl n oder Länge des Eingabestrings).

Du kannst annehmen, dass arithmetische Operationen mit Ganzzahlen in konstanter Zeit berechnet werden können, unabhängig von ihrer Grösse. Die Darstellung einer Zahl benötigt $O(1)$ Speicher.

Aufsteigend

Es sei $D_{n,k}$ eine Liste aller *aufsteigenden* Folgen von n ganzen Zahlen mit Elementen zwischen 1 und k. Zum Beispiel:

$$D_{3,4} = (1,2,3),\ (1,2,4),\ (1,3,4),\ (2,3,4)$$
$$D_{2,5} = (1,2),\ (1,3),\ (1,4),\ (1,5),\ (2,3),\ (2,4),\ (2,5),\ (3,4),\ (3,5),\ (4,5)$$

In jeder List $D_{n,k}$ haben die einzelnen Folgen eine festgelegte Reihenfolge: die *lexikographische* Ordnung. Diese Ordnung wird bei beiden der obigen Beispiele verwendet.

Formal ist die lexikographische Ordnung wie folgt definiert:

Die Folge $a_0, \ldots, a_{n-1}$ ist *lexikographisch kleiner* als die Folge $b_0, \ldots, b_{n-1}$ falls es einen index $j \geq 0$ gibt sodass $a_j < b_j$ und wir für alle $i < j$ haben dass $a_i = b_i$.

Zum Beispiel, $(3,7,12)$ ist lexikographisch kleiner als $(3,8,9)$. Also würde in $D_{3,20}$ die Folge $(3,7,12)$ früher auftreten als $(3,8,9)$. Die Folge $(3,7,12)$ ist auch kleiner als $(4,5,6)$.

Task

Du bekommst n, k, und eine Folge S, welche eine der Folgen in $D_{n,k}$ ist. Deine Aufgabe ist es, den Index von S in $D_{n,k}$ zu finden. (Die erste Folge in $D_{n,k}$ hat den Index 1.)

Zum Beispiel, in $D_{3,2}$ ist der Index von $(1,2,4)$ gleich 2 und der Index von $(2,3,4)$ ist 4.

Beispiel

$n = 5, k = 8, S = [1,2,3,4,8]$ (Antwort: 4; die ersten vier Folgen in $D_{5,8}$ sind $(1,2,3,4,5)$, $(1,2,3,4,6)$, $(1,2,3,4,7)$, und **(1,2,3,4,8)**).

Zyklische Verschiebung

Die i-te *zyklische Verschiebung* der Sequenz $A = (a_0, \ldots, a_{n-1})$ ist die Sequenz, die entsteht, wenn man die Elemente von A um i Positionen nach links verschiebt und den Anfang mit dem Ende verbindet. Formell, für $0 \le i < n$ ist die i-te zyklische Verschiebung definiert als $C(A, i) = (a_i, \ldots, a_{n-1}, a_0, a_1, \ldots, a_{i-1})$.

Zum Beispiel für $A = (3, 4, 5, 6)$ gibt es vier zyklische Verschiebungen:

$$\begin{aligned} C(A,0) &= (3,4,5,6) \\ C(A,1) &= (4,5,6,3) \\ C(A,2) &= (5,6,3,4) \\ C(A,3) &= (6,3,4,5) \end{aligned}$$

Beachte, dass für einige Sequenzen verschiedene zyklische Verschiebungen dem selben Resultat entsprechen können. Zum Beispiel, falls $B = (1, 2, 1, 2)$ dann $C(B, 0) = C(B, 2) = (1, 2, 1, 2)$ und $C(B, 1) = C(B, 3) = (2, 1, 2, 1)$.

Die zyklische Verschiebung einer Sequenz A kann *lexikographisch geordnet* werden. Das kleinste Element in dieser Ordnung nennen wir die *lexikographisch kleiste zyklische Verschiebung*.

Genauer: die lexikographische Ordnung ist definiert als:

Die Sequenz $X = (x_0, \ldots, x_{n-1})$ ist *lexikographisch kleiner* als die Sequenz $Y = (y_0, \ldots, y_{n-1})$ falls es einen Index $j \ge 0$ gibt, sodass $x_j < y_j$ und für alle $i < j$ gilt $x_i = y_i$. Wir schreiben dies als $X \prec Y$.

Im Besipiel oben gilt $C(A, 0) \prec C(A, 1) \prec C(A, 2) \prec C(A, 3)$ und $C(B, 0) = C(B, 2) \prec C(B, 1) = C(B, 3)$. Für die Sequenz $C = (3, 1, 3, 7)$ ist die lexikographische Ordnung ihrer zyklischen Verschiebungen $C(C, 1) \prec C(C, 0) \prec C(C, 2) \prec C(C, 3)$. Beachte, dass $C(C, 0) = (3, 1, 3, 7) \prec (3, 7, 3, 1) = C(C, 2)$.

Also ist die lexikographisch kleinste Verschiebung von A die Verschiebung um 0, für C die Verschiebung m 1 und für B gibt es mehrere Verschiebungen, welche der kleinsten lexikographischen Verschiebung entsprechen: 0 und 2.

Aufgabe

Du kennst n ($1 \le n \le 100\,000$), die Länge einer versteckten Sequenz $A = (a_0, a_1, \ldots a_{n-1})$.

Deine Aufgabe ist es zu bestimmen welche zyklischen Verschiebungen die kleinsten sind. Allerdings kennst du die Sequenz nicht. Um die Informationen zu erhalten, die du benötigst, musst du Fragen zur Sequenz stellen. Für jede Frage wählst du zwei Indizes i und j ($0 \le i, j < n$) und rufst die Funktion CMP(i, j) auf. Diese gibt entweder <, = oder > zurück, je nach dem ob $a_i < a_j$, $a_i = a_j$ resp. $a_i > a_j$.

Finde heraus, für welches k die zyklischen Verschiebungen $C(A, k)$ lexicographisch minimal ist. Falls es mehr als eine Lösung gibt, kannst du eine beliebige auswählen.

Minimiere in erster Linie die asymptotische Anzahl Aufrufe von CMP, in zweiter Linie Laufzeit/Speicher.

Verbundenheit

Maus Lara Gut hat viele Hobbies, sie liebt Singen und Tanzen, aber ihre wahre Leidenschaft ist Sport, also konnte sie die Olympischen Spiele in Sochi nicht verpassen. An der Olympiade angekommen, war sie sofort vom Olympischen Logo beeindruckt: fünf Ringe, die sogar miteinander verbunden sind. Sie realisierte, dass sie Verbundenheit mag, und deshalb entschied sie, ein neues Logo zu entwerfen, auf dem alles miteinander Verbunden ist. Sie begann mit Segmented auf einer Linie. Nachdem sie ein paar Stück gemalt hatte, fragte sie sich, wieviele Segmente sie von ihrer Menge auswählen kann, sodass alle Paare von gewählten Segmenten verbunden sind.

Lara glaubt dass zwei Segmente verbunden sind, wenn (und nur wenn) die Länge ihres Schnittes grösser als Null und kleiner als die Länge jedes der Segmente ist. Das bedeutet, dass die Segmente überlappen müssen, aber keines der beiden sollte im anderen enthalten sein. Segmente mit genau einem gemeinsamen Punkt werden als nicht verbunden angesehen.

Aus der gesamten Menge von n Segmenten, die Lara gemalt hat, gibt sie dir eine Liste von k Segmenten, die sie besonders mag. Für jedes dieser Segmente sollst du für Sie die maximale Menge von Segmenten entdecken welche dieses Segment enthält und in welcher alle Paare von Segmenten verbunden sind.

Es ist garantiert dass keine zwei Segmente dasselbe linke Ende oder dasselbe rechte Ende haben.

Bitte hilf Lara ihr Problem so schnell wie möglich zu lösen, sodass sie weiter Medaillen für die Schweiz gewinnen gehen kann.

Beispiel

$n = 3, s = [(1, 4), (3, 6), (2, 5)], x = [2, 1, 3]$ (Antwort: 3, 3, 3)

Morpher

In dieser Aufgabe verwenden wir ein Alphabet, welches aus den ersten n Grossbuchstaben des englischen Alphabets besteht. Ein *Morpher* ist eine Funktion f, welche auf den Buchstaben unseres Alphabetes definiert ist. Für jeden Buchstaben gibt f einen String zurück, der aus mindestens zwei Buchstaben besteht.

Ein Beispiel für einen Morpher für $n = 3$: $f(\texttt{A}) = \texttt{ABC}$, $f(\texttt{B}) = \texttt{AB}$, $f(\texttt{C}) = \texttt{BC}$.

Sobald wir einen Morpher definiert haben, können wir ihn verwenden um Strings zu transformieren. Das Abbild eines Strings erhalten wir indem wir die Bilder seiner Buchstaben aneinander hängen. Wenn wir zum Beispiel den oben definierten Morpher verwenden, kommt $f(\texttt{BAB}) = f(\texttt{B}) + f(\texttt{A}) + f(\texttt{B}) = \texttt{ABABCAB}$ heraus.

Da die Ausgabe das selbe Alphabet benutzt wie die Eingabe, ist es möglich den selben Morpher auf einem gegebenen String mehrmals hintereinander anzuwenden. Zum Beispiel: $f(f(\texttt{C})) = f(\texttt{BC}) = \texttt{ABBC}$.

Wir geben dir einen Morpher f, einen String s und eine Ganzzahl k und einen Index p. Stell dir vor, dass wir mit dem String s beginnen und dann den Morpher f genau k mal hintereinander angewendet haben. Gib den p-ten Buchstaben (in 0-basierter Nummierung) des resultierenden Strings aus.

Falls der resultierende String weniger als p Zeichen besitzt, gib stattdessen “-” aus.

Beispiele

- $n = 3, k = 1, p = 3, s = \texttt{ABC}, m = [\texttt{ABC}, \texttt{AB}, \texttt{BC}]$ (Antwort: A).
- $n = 3, k = 1, p = 7, s = \texttt{ABC}, m = [\texttt{ABC}, \texttt{AB}, \texttt{BC}]$ (Antwort: -).