

Webfejlesztő programozási feladat

Adott egy számokból álló háromszög, például:

```
      1
     2 1
    4 1 8
   2 2 3 4
```

Függőleges út: Irányítatlan (fentről lefelé vagy lentől fölfelé ugyanaz), az adott elem alatt közvetlenül álló két elemből az egyiket választva végighaladunk a háromszögön függőleges irányban. Például:

1 -> 1 -> 8 -> 4, 1 -> 1 -> 1 -> 2

Út összege: Az utat kiadó elemek összege. Például a fenti első útra:

$\text{Szumma}(1 \rightarrow 1 \rightarrow 8 \rightarrow 4) = 1 + 1 + 8 + 4 = 14$

1. feladat

Hány lehetséges út van egy N magas háromszögben?

2. feladat

Írjon parancssori programot a maximális összegű út összegének megtalálása. A programnak "nagyon nagy" (pl. 50 magas) háromszögekre is hamar (1 sec alatt, körül) le kell tudnia futni egy mai átlagos gépen. A program a standard bemeneten kapja a háromszöget, soronként a sorokat (sorhatároló karakter a `\n`), az elemek egyetlen szóközzel vannak elválasztva. A program kimenete egyetlen szám legyen, a maximális összegű út összege.

Összehasonlításképpen, a Ruby (tehát interpretált) referencia implementációnk véletlenszerű, 50 magas háromszögekre 1 másodpercnél kevesebb idő alatt szolgáltat eredményt 2.53 GHz-es Core2Duo processzoron.

Webfejlesztő programozási feladat

Írj PHP nyelven egy függvényt, mely a paraméterben kapott stringből felismeri a leghosszabb ismétlődő stringrészletet és ezzel tér vissza.

Például:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed imperdiet orci ut erat elementum mollis. Morbi quis dolor ipsum, sit amet sagittis urna. Sed ullamcorper lacinia elit nec mollis. Vestibulum nec magna velit, ac porttitor nisl. Nulla in venenatis risus. Phasellus a nisl nulla, id condimentum nibh. Donec eget eros id eros cursus viverra. Maecenas et adipiscing velit. Proin porta placerat vulputate. Duis varius erat eget lorem pellentesque rhoncus. Mauris vehicula interdum mauris, in sodales metus aliquam ac. Etiam ac porttitor quam. Vivamus venenatis quam molestie mi ultrices a laoreet enim faucibus.

Ebben a leghosszabb ismétlődő rész-string: '**ac porttitor**'

Törekedj a minél elegánsabb, átgondoltabb, hibátűrőbb és hatékonyabb megoldásra, fel kell készülni nagyon hosszú stringek kezelésére is.

Webfejlesztő programozási feladat

Kerek prímnek nevezzük az olyan prímet, mely számjegyeinek összes "forgatásából" adódó szám is prím.

Például: **719** --> 197 --> 971 --> 719 (körbeértünk), tehát 719 összes forgatása: 197, 971, amik szintén prímek, tehát 719 kerek prím.

Feladat: Hány kerek prím van 1 millió alatt?

Amit várunk: egy kész implementáció, mely kiszámolja és kiírja az 1 millió alatti kerek prímek számát. Mellékelve vagy a kódban megjegyzésként szeretnénk magyarázó szöveget is látni.

Webfejlesztő programozási feladat

Adott egy számokból álló tömb. Az **egyensúlyi index** egy olyan elem indexe, melyre teljesül hogy a nála kisebb indexű elemek összege megegyezik a nála nagyobb indexű elemek összegével.

Például: `[-7, 1, 5, 2, -4, 3, 0]` --> egyensúlyi indexek: **3, 6**

Feladat: Egy `e_index(tomb) {}` függvény készítése, mely visszatér egy egyensúlyi index-szel, vagy ha nem létezik ilyen, `-1`-gyel. A tömb "nagyon nagy" is lehet, hatékony megoldásra van szükség.

Segítség: létezik $O(n)$ megoldás.

A feladat megoldását PHP nyelven várjuk.