**Felvételi előkészítő**

**(Elemi algoritmusok; Számjegyek és oszthatóság)**

1. Határozzuk meg egy adott ***n*** természetes számnál kisebb számok közül azt, amelyiknek a legtöbb valódi osztója van. A valódi osztó nagyobb mint 1, és kisebb, mint maga a szám! Írjuk ki a számot és valódi osztóinak darabszámát.

**Példa**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 1000 | 840 30 |

1. Ismert, hogy ha egy adott ***n*** természetes számot ismételten alávetünk a következő feldolgo­zásnak, eljutunk az 1-es számhoz: ha ***n*** páros, akkor elosztjuk 2-vel, ha ***n*** páratlan, megszo­rozzuk 3-mal és az eredményhez hozzáadunk 1-et. Azoknak a lépéseknek a számát, amelyek egy adott ***n***-ből az 1-hez vezetnek, *karakterisztikának* nevezzük. Írjunk programot, amely meghatározza azt az ***n*** számot, amely az [***a***, ***b***] intervallumhoz tartozik és amelynek a karak­terisztikája a legnagyobb.

**Példa**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 1 4 | 3 |

1. Adott a következő sorozat, amelynek minden elemét – az elsőt kivéve – az előző elem segít­ségével generáljuk: 1, 11, 21, 1211, 111221, ...

A generálási szabály a következő:

* megszámoljuk, balról jobbra haladva az előző érték számjegyeit
* az új értéket úgy kapjuk meg, hogy beírjuk a régi érték összes számjegyeinek előfor­dulási számát és az illető számjegyet.

Határozzuk meg az ***n***-edik (***n*** ≤ 20) elemét a sorozatnak!

**Példa**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 6 | 312211 |

1. Döntsük el egy adott számról, hogy völgyszám-e vagy hegyszám. Egy völgyszám számjegyei csökkenő sorrendben követik egymást egy bizonyos elemmel bezárólag, azután *szigorúan*növekvő so­rozatot alkotnak. A hegyszám számjegyei *szigorúan*növekvő sorrendben követik egymást egy bizonyos elemmel bezárólag, azután csökkenő sorozatot alkotnak.

Írjunk ki egy megfelelő üzenetet aszerint, hogy az adott szám völgyszám-e vagy hegyszám. Ha a szám nem völgyszám és nem hegyszám, vágjuk le a szám *első* néhány számjegyét amíg az így kapott szám völgyszámmá vagy hegyszámmá válik. Írjuk ki az adott szám azon részét, amely völgyszám vagy hegyszám, ha létezik ilyen, vagy megfelelő üzene­tet, ha nem.

**Példák**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 13752 | hegyszam |
| 85369 | volgyszam |
| 913752 | 13752 |
| 1234 | nincs benne hegyszam vagy volgyszam |

1. Írjuk ki n darab természetes szám legnagyobb közös osztóját!

**Példa**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 3  16 24 32 | 8 |

1. Írjunk algoritmust, amely megadja a Fibonacci-sorozat egy adott számnál kisebb elemeinek számát! A Fibonacci-sorozat nulladik elemét nem kell figyelembe venni.

**Példa**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 54 | 9 darab kisebb elem van a Fibonacci-sorozatban, mint 54 |

1. Írjunk algoritmust, amely egy adott, 6-nál nagyobb *páros* számot felír két különböző páratlan prímszám összegeként (Goldbach-sejtés).

**Példák**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 10  24 | 10 = 3 + 7  24 = 5 + 19 |

1. Egy sorozat ***r*** léptékű számtani haladvány, ha bármely elemét – kivéve az elsőt – megkapjuk, ha az előtte levőhöz hozzáadunk ***r***-t. Például, a (12, 14, 16, 18, 20) sorozat egy 2 léptékű számtani haladvány.

Adott legtöbb 3 ≤ ***n*** ≤ 105 természetes szám, ahol 0 ≤ ***szám*** ≤ 103. Döntsük el, hogy létezik-e egy ***r*** természetes szám, amely annak a számtani haladványnak a léptéke, amelyet úgy kapnánk, ha az adott sorozat ***különböző***értékeit átrendeznénk. Ha nem létezik ilyen ***r*** szám, írjunk ki megfelelő üzenetet, különben írjuk ki a lépték értékét. **Oldjuk meg a feladatot rendezés nélkül!**

**Példa**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 11  180 30 80 280 130 330 230 30 30 330 80 | 50 |

1. Egy pontosan *kétjegyű* ***x*** természetes számot az ***y*** természetes szám „alszám”-ának nevezzük, ha ***x*** számjegyei megjelennek, eredeti sorrendjükben, egymás után az ***y*** számban. Például a 41 alszáma 14121-nek, 413-nak és 41-nek, de nem alszáma 143-nak és 431-nek, 77 alszáma 77757-nek, ahol kétszer fordul elő.

Legyen az ***n*** elemű ***a***, természetes számokat tároló sorozat (0 ≤ ***n*** ≤ 106, 0 < ***ai*** ≤ 29, ***i*** = 1, 2, …, ***n***).

Írjatok programot, amely meghatározza azt a ***k*** elemű ***b*** sorozatot, amely az ***a*** sorozat elemeinek azokat az alszámait tartalmazza, amelyek a *legtöbbször* for­dulnak elő. A ***b*** sorozat elemeinek sorrendje tetszőleges.

**Példák**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 5  733325, 698787, 127898, 613373, 212612 | 2  12 33 |

Megjegyzés: a 12 és 33 alszámok, mindkettő háromszor fordul elő az ***a*** sorozatban.

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 2  5907 23 | 3  59 90 23 |

1. Egy természetes számot *majdnem prím*nek nevezünk, ha egyenlő *két különböző prímszám szorzatával.* Például, a 15 *majdnem prím*, mivel egyenlő a 3 és 5 prímszámok szorzatával.

Legyen egy ***n*** természetes szám(1 ≤ ***n*** ≤ 100 000) és egy ***n*** elemű ***x*** sorozat, amelyben az ele­mek 1-nél szigorúan nagyobb és 30 000-nél kisebb természetes számok.

Írjatok programot, amely meghatározza az adott sorozat leghosszabb tömbszakaszát, amely csak *majdnem prím*eket tartalmaz, és kiírja az illető tömbszakasz kezdőindexét (***balMax***) és végsőindexét (***jobbMax***). Ha több ilyen tömbszakasz létezik, a *legelső* leghosszabb tömbsza­kasz kezdőindexét és végsőindexét kell kiírnotok. Ha a sorozatban nem létezik egyetlen *majdnem prím* sem, ***balMax*** és ***jobbMax*** értéke egyenlő lesz -1-gyel.

**Példák**

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 8  24 34 35 11 8 77 35 26 | 6 8 |

(***x***6 = 77 = 7 \* 11, ***x***7 = 35 = 5 \* 7, ***x***8 = 26 = 2 \* 13).

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 3  24 11 8 | -1 -1 |

## A példákban a sorozatot 1-től kezdődően indexeltük.