

## Díjazás

Egy programozási versenyen az első három helyezett összesen  $E$  euró jutalmat kap. A szervezők még nem döntötték el, hogyan osszák el a három dobogós között ezt az összeget. Úgy szeretnék szétosztani a nyereményt, hogy az első többet kapjon, mint a második és a második többet kapjon, mint a harmadik, továbbá mindhárman kapjanak legalább 1 eurót.

Írj programot, ami meghatározza az összes lehetséges díjazást! A díjazásokat fordított ábécé szerinti sorrendben kell felsorolni. Ez azt jelenti, hogy az  $A = (a_1, a_2, a_3)$  (ahol  $a_1, a_2, a_3$  rendre az első, második és harmadik helyezett díja) akkor és csak akkor van előbb, mint a  $B = (b_1, b_2, b_3)$  díjazás, ha  $a_1 > b_1$ , vagy  $a_1 = b_1$  és  $a_2 > b_2$ , vagy  $a_1 = b_1$  és  $a_2 = b_2$  és  $a_3 > b_3$ . Például  $E = 11$  esetén a  $(6, 4, 1)$  díjazás megelőzi a  $(6, 3, 2)$  díjazást a felsorolásban.

### Bemenet

A standard bemenet első sora egy egész számot tartalmaz, a jutalomként kiosztandó  $E$  összeget.

### Kimenet

A standard kimenetre soronként egy-egy lehetséges díjazást kell kiírni, fordított ábécé szerinti sorrendben! Minden sorban elsőként az első, másodiknak a második és harmadiknak a harmadik helyezett versenyző díja álljon.

### Példa

Bemenet	Kimenet
11	8 2 1 7 3 1 6 4 1 6 3 2 5 4 2

### Korlátok

$$6 \leq E \leq 1000$$

A lehetséges díjazások száma legfeljebb 100 000

**Időlimit:** 0.5 mp.

**Memórialimit:** 64 MB

### Pontozás

A megoldásodat több különböző tesztesetre lefuttatjuk. A teszteseteknek önálló pontértéke van és a megoldás pontszáma a megoldott tesztesetek pontszámainak összege. A feladat összpontszáma a legtöbb pontot elért megoldás pontszáma.

A pontszám 10%-a szerezhető olyan tesztekre, ahol  $E \leq 10$ .

A pontszám további 10%-a szerezhető olyan tesztekre, ahol  $E \leq 50$ .

A pontszám további 20%-a szerezhető olyan tesztekre, ahol  $E \leq 100$ .

A pontszám további 20%-a szerezhető olyan tesztekre, ahol  $E \leq 500$ .

## Lapozgatás

Hanga képeskönyvet nézeget. A könyvnek  $M$  oldala van, melyek aljára az  $1, \dots, M$  számokat nyomtatták. A könyv kezdetben a  $K$ -adik oldalon van kinyitva.

Hanga szeretné megnézni az  $A_1, A_2, \dots, A_N$  oldalak mindegyikét, ebben a sorrendben. Ehhez minden lépésben az aktuális oldaltól kiindulva, egyesével lapozva megnézi az összes oldalt egészen addig az oldalig, amit meg akart nézni. Az egyetlen kivétel, ha a megtekinteni kívánt oldal olyan, amit korábban már megnézett: ekkor lapozás helyett marad a mostani oldalon.

Például legyen  $K = 5$  és  $A = [3, 4, 6]$ . Most a könyv az 5-ödik oldalon van kinyitva.

- Hanga következőnek a 3-adik oldalt szeretné megnézni: ehhez sorban megnézi az 5-ödik, 4-edik és 3-adik oldalakat.
- Következőnek a 4-edik oldalt szeretné megnézni, de ezt már korábban megnézte, így marad a 3-adik oldalon.
- Utolsónak a 6-odik oldalt szeretné megnézni, ezt még nem látta, így egyenként megnézi a következő oldalakat: 3, 4, 5, 6.

Írj programot, ami kiszámítja, hogy hányszor kezdett el lapozni Hanga, és ezek alatt összesen hányat lapozott a könyvben! A fenti példában kétszer kezdett el lapozni, és először kétszer lapozott ( $5 \rightarrow 4$  és  $4 \rightarrow 3$ ), majd végül háromszor ( $3 \rightarrow 4$ ,  $4 \rightarrow 5$  és  $5 \rightarrow 6$ ), így a lapozások száma 5.

### Bemenet

A standard bemenet első sorában a könyv oldalainak  $M$  száma, a kezdeti oldal  $K$  sorszáma és a megtekinteni kívánt oldalak  $N$  száma található.

A második sor  $N$  pozitív egész számot tartalmaz, a megtekinteni kívánt oldalak  $A_i$  sorszámait.

### Kimenet

A standard kimenetre egy sort kell írni két egész értékkel: az első szám megadja, hogy hányszor kezdett el Hanga lapozni a könyvben, a második pedig, hogy összesen hány oldalt lapozott.

### Példa

Bemenet	Kimenet
10 5 3	2 5
3 4 6	
Bemenet	Kimenet
30 1 8	4 23
10 13 7 9 16 2 15 24	

### Korlátok

$$10 \leq M \leq 10^9$$

$$1 \leq K \leq M$$

$$1 \leq N \leq 200\,000$$

$$1 \leq K, A_i \leq M \text{ minden } i = 1 \dots N\text{-re}$$

Az  $A_i$  számok és a  $K$  érték mind különbözőek

**Időlimit:** 0.5 mp.

**Memórialimit:** 256 MB

### **Pontozás**

A megoldásodat több különböző tesztesetre lefuttatjuk. A teszteseteknek önálló pontértéke van és a megoldás pontszáma a megoldott tesztesetek pontszámainak összege. A feladat összpontszáma a legtöbb pontot elért megoldás pontszáma.

A pontszám 12%-a szerezhető olyan tesztekre, ahol  $N = 1$ .

A pontszám további 24%-a szerezhető olyan tesztekre, ahol  $M \leq 1000$  és  $N \leq 1000$ .

A pontszám további 28%-a szerezhető olyan tesztekre, ahol  $K = 1$ .

Minden tesztesetben **részpontokat** lehet szerezni: a teszteset pontszámának 50%-a jár, ha az első kiírt szám helyes. Ügyelj arra, hogy a megoldásod minden esetben pontosan két egész értéket írjon ki, különben előfordulhat, hogy az értékelő formai hiba miatt nem adja meg a részpontokat.

## Két ajándék

Az iskolai programozási verseny győztese egy asztalra sorban lerakott díjak közül választhat kettőt. Összesen  $N$  díj van az asztalon, melyeket sorrendben 1-től  $N$ -ig számozunk. Az  $i$ -edik díj értéke  $E_i$ . A díjakat úgy kell kiválasztania, hogy a nagyobb sorszámú kiválasztott díj értéke nagyobb legyen a kisebb sorszámúénál. Formálisan, a győztesnek meg kell jelölnie két sorszámot,  $p$ -t és  $q$ -t úgy, hogy  $p < q$  és  $E_p < E_q$  teljesüljön.

A versenyző természetesen arra törekszik, hogy a két díj összértéke a lehető legnagyobb legyen. Melyik két díjat kell ehhez kiválasztania?

### Bemenet

A standard bemenet első sora egy egész számot tartalmaz, a díjak  $N$  számát. A második sor  $N$  pozitív egész számot tartalmaz, a díjak  $E_i$  értékeit.

### Kimenet

A standard kimenetre a két kiválasztott díj sorszámát kell kiírni! Ha nem választható ki két díj a feltételeknek megfelelően, akkor két 0 számot kell kiírni!

Több megoldás esetén bármelyik megadható.

### Példa

Bemenet

10  
1 9 6 2 7 2 8 4 6 3

Kimenet

5 7

### Korlátok

$$2 \leq N \leq 50\,000$$

$$1 \leq E_i \leq 100\,000\,000 \text{ minden } i = 1 \dots N\text{-re}$$

**Időlimit:** 0.5 mp.

**Memórialimit:** 64 MB

### Pontozás

A megoldásodat több különböző tesztesetre lefuttatjuk. A teszteseteknek önálló pontértéke van és a megoldás pontszáma a megoldott tesztesetek pontszámainak összege. A feladat összpontszáma a legtöbb pontot elért megoldás pontszáma.

A pontszám 10%-a szerzhető olyan tesztekre, ahol  $N \leq 100$  és pontosan kétféle értékű díj van.

A pontszám további 10%-a szerzhető olyan tesztekre, ahol  $N \leq 100$  és pontosan háromféle értékű díj van.

A pontszám további 20%-a szerzhető olyan tesztekre, ahol  $N \leq 100$  és  $E_i \leq 10\,000$ .

A pontszám további 10%-a szerzhető olyan tesztekre, ahol  $N \leq 10\,000$  és  $E_i \leq 100$ .

A pontszám további 10%-a szerzhető olyan tesztekre, ahol  $N \leq 10\,000$ .

A pontszám további 10%-a szerzhető olyan tesztekre, ahol  $N \leq 30\,000$ .

## Szerencsés számok

Áron szereti a hárommal osztható számokat. Úgy véli, hogy ezek a számok szerencsét hoznak, ezért a hárommal osztható természetes számokat **szerencsés számoknak** nevezi.

Egy nap úgy döntött, hogy meglepi a barátait egy játékos feladattal: írják le egy papírra a lehető legtöbb szerencsés számot úgy, hogy a 0 számjegyből legfeljebb  $d_0$  darabot, az 1 számjegyből legfeljebb  $d_1$  darabot, ..., a 9 számjegyből pedig legfeljebb  $d_9$  darabot írnak le összesen a papírra. Ugyanaz a szerencsés szám akárhányszor leírható.

Segíts Áron barátainak: írd meg a programot, ami megadja, hogy legfeljebb hány szerencsés számot lehetséges leírni a rendelkezésre álló számjegyek felhasználásával!

### Bemenet

A standard bemenet első és egyetlen sorában 10 nemnegatív egész található, az egyes számjegyekből felhasználható  $d_0, d_1, \dots, d_9$  mennyiségek.

### Kimenet

A standard kimenetre egy sort kell írni, mely a papírra leírható szerencsés számok maximális darabszámát tartalmazza.

### Példa

Bemenet	Kimenet
0 1 0 0 0 0 0 1 0 0	0
Bemenet	Kimenet
0 2 1 0 0 0 0 1 0 0	1
Bemenet	Kimenet
3 1 1 2 0 5 1 1 3 1	11

Az első példában nem tudunk szerencsés számot írni a papírra.

A második példában a számjegyekből a 12, 21, 27, 72, 117, 171, 711 szerencsés számok készíthetők el, melyek közül legfeljebb egy írható fel a papírra.

### Korlátok

$$0 \leq d_i \leq 1\,000\,000 \text{ minden } i = 0 \dots 9\text{-re}$$

**Időlimit:** 0.5 mp.

**Memórialimit:** 256 MB

### Pontozás

A megoldásodat több különböző tesztesetre lefuttatjuk. A teszteseteknek önálló pontértéke van és a megoldás pontszáma a megoldott tesztesetek pontszámainak összege. A feladat összpontszáma a legtöbb pontot elért megoldás pontszáma.

A pontszám 12%-a szerezhető olyan tesztekre, ahol csak a 0 és 1 számjegyek használhatók.

A pontszám további 10%-a szerezhető olyan tesztekre, ahol csak a 0, 1 és 2 számjegyek használhatók és  $d_0 = d_1 = d_2$ .

Az **összes** pont 66%-a szerezhető olyan tesztekre, ahol  $d_i \leq 10$  minden  $i = 0 \dots 9$ -re.