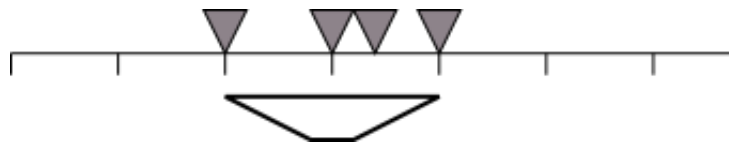


## Elefántok

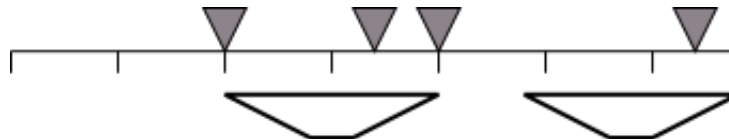
Egy mutatóványban  $N$  elefánt szerepel, egy sorban állnak, egy helyen egymás mögött akár több is állhat. A mutatóvány során egy lépésben egy elefánt mozdul el a helyéről.

Az elefántokról fényképek készülnek, egy fényképezőgép  $L$  hosszú intervallumban levő elefántokat tud egyszerre fényképezni, tehát több fényképezőgépre is szükség lehet, hogy egy adott pillanatban minden elefánt rajta legyen legalább egy fényképen.

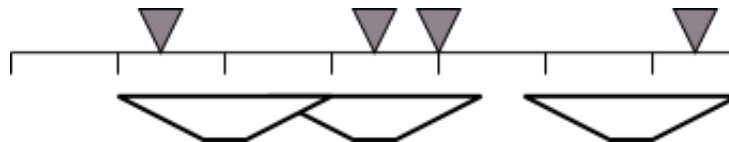
Ha  $L=10$  és az elefántok a 10, 15, 17 és 20 pozícióban vannak, akkor egyetlen fényképezőgép elég. (Az elefántokat háromszögekkel, a fényképezőgépeket trapézzal jelöljük.)



A következő lépésben a 15 pozíción levő elefánt a 32-re lép. Ezután már két fényképezőgép kell.



A 10 pozíción levő elefánt a 7-re lép. Ezután már három fényképezőgép kell.



Írj interaktív programot, amely minden lépés után kiszámítja, hogy minimálisan hány fényképezőgép kell!

## Feladat

Írd meg a következő eljárásokat:

♣ `init(N,L,X)`:

♣  $N$  – az elefántok száma. Az elefántokat  $0$ -tól  $N-1$ -ig sorszámozzuk.

♣  $L$  – egy fényképezőgéppel lefényképezhető intervallum hossza ( $0 \leq L \leq 1\,000\,000\,000$ ).

♣  $X$  – az elefántok kezdeti pozícióját megadó egydimenziós tömb. Az  $i$ . elefánt kezdetben az  $X[i]$  pozíción van ( $0 \leq i < N$ )  $X$  rendezett, azaz  $0 \leq X[0] \leq \dots \leq X[N-1] \leq 1\,000\,000\,000$ .

Ezt az eljárást hívják meg először, egyszer. Nincs visszaadott értéke.

♣ `update(i,y)`:

♣  $i$  – a lépésben mozduló elefánt eredeti sorrendbeli sorszáma.

♣  $y$  – a pozíció, ahova az  $i$ . elefánt elmozdul ( $0 \leq y \leq 1\,000\,000\,000$ ).

Ezt a függvényt sokszor hívhatják. Minden hívásra az értéke a lépés után szükséges fényképezőgépek minimális száma.

## Példa

$N=4$ ,  $L=10$ , az elefántok kezdeti pozíciója

10  
 $X=$  15  
17  
20

Először az init eljárást hívják meg a fenti paraméterekkel. Ezután az update függvényt az alábbi paraméterekkel hívják, a helyes visszatérési értéket a táblázat tartalmazza:

lépés	hívás	érték
1	update(2,16)	1
2	update(1,25)	2
3	update(3,35)	2
4	update(0,38)	2
5	update(2,0)	3

## Tesztek

### 1. eset (10 pont)

- ♣  $N = 2$ .
- ♣ Soha nincs egy helyen több elefánt.
- ♣ Az update függvényt legfeljebb **100**-szor hívják.

### 2. eset (16 pont)

- ♣  $1 \leq N \leq 100$ .
- ♣ Soha nincs egy helyen több elefánt.
- ♣ Az update függvényt legfeljebb **100**-szor hívják.

### 3. eset (24 pont)

- ♣  $1 \leq N \leq 50\,000$ .
- ♣ Soha nincs egy helyen több elefánt.
- ♣ Az update függvényt legfeljebb **50 000**-szer hívják.

### 4. eset (47 pont)

- ♣  $1 \leq N \leq 70\,000$ .
- ♣ Több elefánt is lehet azonos pozíción.
- ♣ Az update függvényt legfeljebb **70 000**-szer hívják.

### 5. eset (3 pont)

- ♣  $1 \leq N \leq 150\,000$ .
- ♣ Több elefánt is lehet azonos pozíción.
- ♣ Az update függvényt legfeljebb **150 000**-szer hívják.

---

## Határok

- ⤴ Időlimit: 9 másodperc

**Megjegyzés:** A C++ Standard Library használata lassúvá teheti a programod, ezért nem biztos, hogy maximális pontszámot tudsz elérni.

- ⤴ Memórialimit: 256 MB

**Megjegyzés:** A verem méretre nincs külön korlát.

## Interfész (API)

- ⤴ A megoldás könyvtára: elephants/

- ⤴ A megvalósítandó modul: elephants.c vagy elephants.cpp vagy elephants.pas

- ⤴ Saját interfész: elephants.h vagy elephants.pas

- ⤴ Minta értékelő: grader.c vagy grader.cpp vagy grader.pas

- ⤴ Minta bemenetek: grader.in.1, grader.in.2, ...

**Megjegyzés:** A minta értékelő a bemenetet a következő formában olvassa:

- ⤴ 1. sor: **N**, **L** és **M**, ahol **M** a lépések száma.

- ⤴ 2...**N**+1. sor: a kezdeti pozíciók; a **k**+2. sorban van az **X[k]** ( $0 \leq k < N$ ).

- ⤴ **N**+2...**N**+**M**+1. sor: az **M** lépés leírása; az **N**+1+j. sorban van az **i[j]**, az **y[j]** és az **s[j]**, egy-egy szóközzel elválasztva, ami azt jelenti, hogy a **j**. lépésben az **i[j]**. elefánt elmozdul az **y[j]** pozícióra, és ezután **s[j]** a helyes megoldás ( $1 \leq j \leq M$ ).

- ⤴ A minta bemenetre elvárt kimenetek: grader.expect.1, grader.expect.2, ..., amelyekben a „Correct.” szövegnek kell lenni.