

## Lo que cuesta sumar

*Johnny Calculín* sabe sumar números mentalmente a gran velocidad, pero el esfuerzo que le supone realizar una suma depende del valor de los sumandos. En concreto, sumar  $a$  más  $b$  le supone un esfuerzo igual a  $a + b$  (independientemente de los valores concretos de  $a$  y  $b$ ). Sencillo.

Cuando tiene que sumar más de dos números la cosa se complica, ya que el orden en que va realizando las sumas afectan al esfuerzo total empleado. Por ejemplo, si tiene que sumar 1, 2 y 3, puede sumar  $1 + 2$  con un esfuerzo igual a 3, y después sumar  $3 + 3$  con un esfuerzo igual a 6, lo que supone un esfuerzo total igual a 9. En cambio, si suma primero  $2 + 3$  con un esfuerzo igual a 5 y después  $5 + 1$  con un esfuerzo igual a 6, el esfuerzo total asciende a 11. Obviamente el resultado de la suma es siempre 6, por las propiedades conmutativa y asociativa de la suma.

Johnny se está preparando para un concurso de sumas y quiere averiguar cómo debería ir sumando los números para necesitar el mínimo esfuerzo. ¿Puedes ayudarle?

$$\begin{array}{r} 8 \\ + 4 \\ \hline ? \end{array}$$

### Entrada

La entrada está compuesta por diversos casos de prueba, ocupando cada uno de ellos dos líneas: la primera línea contiene un entero  $N$  (entre 1 y 100.000) que representa el número de sumandos, y la segunda contiene esos  $N$  sumandos, números enteros entre 1 y 1.000.000.

La entrada termina con un caso donde  $N$  es 0 que no debe procesarse.

### Salida

Para cada caso de prueba se debe escribir una línea con el esfuerzo mínimo necesario para sumar los números de la entrada.

### Entrada de ejemplo

```
3
1 2 3
4
3 1 4 2
4
30 40 50 60
1
5
0
```

### Salida de ejemplo

```
9
19
360
0
```

**Autor:** Alberto Verdejo.

## Unidad Curiosa de Monitorización

La *Unidad Curiosa de Monitorización* (UCM) se encarga de leer los datos proporcionados por una serie de sensores y enviar con cierta periodicidad los datos obtenidos y procesados a los usuarios que se han registrado previamente.

La UCM admite que los usuarios se registren proporcionando un *Identificador*, un número entre  $10^3$  y  $10^9$  que identifica de forma única al usuario, y un *Periodo*, el intervalo de tiempo que transcurrirá entre dos envíos consecutivos de información a ese usuario (entre 1 y 5.000). Es decir, cuando hayan pasado *Periodo* segundos desde que el usuario se registró, este recibirá la información de la UCM por primera vez; y después recibirá la información cada *Periodo* segundos.



Acaban de registrarse varios usuarios, todos con identificadores distintos. ¿Podrías decir a quiénes irán dirigidos los  $K$  primeros envíos de información? Si dos o más usuarios tienen que recibir la información al mismo tiempo, los envíos se realizan en orden creciente de sus identificadores de usuario.

### Entrada

La entrada consta de una serie de casos de prueba. Para cada uno, primero aparece el número de usuarios que se acaban de registrar (no más de 25.000). A continuación, aparecen los datos de las órdenes de registro de cada uno de ellos (el identificador de usuario y la periodicidad con la que desea recibir información). Y por último aparece el número  $K$  de envíos cuyo destinatario se desea conocer ( $1 \leq K \leq 100.000$ ). La entrada termina cuando el número de usuarios sea 0.

### Salida

Para cada caso se escribirán los identificadores de los usuarios que recibirán los  $K$  primeros envíos, uno por línea.

Después de cada caso se escribirá ---.

### Entrada de ejemplo

```
2
1234 200
9000 300
5
0
```

### Salida de ejemplo

```
1234
9000
1234
1234
9000
---
```

**Autor:** Alberto Verdejo.

## Ordenando a los pacientes en urgencias

En el Hospital ACR (*Aquí Curamos Rápido*) se han puesto a mejorar las Urgencias para que los enfermos que llegan con dolencias más graves sean atendidos antes que los demás. Para eso, han comprado una UCM (*Unidad de Catalogación Médica*) que es capaz de valorar instantáneamente el estado de un paciente con un número entero entre 0 y 1.000.000, donde 0 indica que su dolencia es menor (quizá incluso inexistente y sea un mero hipocondríaco) y 1.000.000 indica que el enfermo está casi caminando hacia la luz.



Por desgracia, la afluencia de enfermos es tal que incluso así es muy complicado saber rápidamente quién debería ser el próximo en ser atendido. No hacen más que entrar pacientes nuevos a la vez que los más graves son atendidos, y no es fácil mantenerlos ordenados. Cuando un médico queda libre, debe ser atendido el enfermo a la espera con una valoración más grave. Si hay dos pacientes evaluados con la misma gravedad, deberá ser atendido el que más tiempo lleve esperando.

Para ayudar en la tarea de decidir quién es el próximo, desde ACR se ha hecho un llamamiento para buscar ayuda entre los mejores programadores. ¿Eres tú uno de ellos?

### Entrada

La entrada está formada por diversos casos de prueba. Cada caso comienza con una línea indicando el número  $n$  de eventos que ocurrirán (como mucho 200.000), y a continuación aparecen  $n$  líneas cada una con un evento. Un evento puede ser la llegada de un paciente nuevo, o la atención por parte de un médico que ha quedado libre del paciente más urgente. Los ingresos de pacientes nuevos se indican de la forma **I nombre gravedad**, donde **nombre** es una cadena de como mucho 20 caracteres (sin espacios) y **gravedad** es un número entre 0 y 1.000.000 con su estado (0 leve, 1.000.000 muy grave). Los eventos en los que se atiende al siguiente paciente se indican con el carácter **A**. Se garantiza que no habrá nunca eventos de tipo **A** si no quedan pacientes esperando.

La entrada termina cuando el número de eventos es 0.

### Salida

Para cada evento de tipo **A** de cada caso de prueba se escribirá el nombre del paciente que es atendido en ese momento. Se atiende primero al paciente más grave y, en caso de igualdad entre dos o más pacientes, se elegirá entre ellos al que más tiempo lleve esperando.

Al finalizar el tratamiento de cada caso se escribirá una línea más con tres guiones (---).

### Entrada de ejemplo

```
9
I Alberto 4000
I Pepe 3000
A
I Rosa 2000
I Laura 5000
A
I Sara 3000
A
A
0
```

## Salida de ejemplo

```
Alberto  
Laura  
Pepe  
Sara  
---
```

**Autores:** Alberto Verdejo y Pedro Pablo Gómez Martín.

## Reina del súper

Hace tiempo, Ismael se enamoró perdidamente de una cajera del supermercado. No se cansa de espiarla entre la sección de ofertas y menaje del hogar. Tiene la tarjeta echando humo, porque aunque tenga la nevera llena, todas las tardes baja a comprar cualquier cosa con tal de volver a verla. Y da igual cuánta gente haya; él siempre se pone a esperar en la fila de su caja.



Pero al llegar hoy al súper se ha llevado una desagradable sorpresa. Al cerebritito de turno que gestiona el supermercado le ha dado por instaurar la “fila única”. Ahora en lugar de haber filas independientes para cada caja, hay una única fila para todas y cuando una caja se queda vacía, el primero que ocupa la fila va a ella.

El gerente del súper seguramente piense que ahora los clientes quedarán más satisfechos con el servicio proporcionado porque saben que nunca esperarán de más, pero a Ismael le han destrozado la vida. Ya no tiene la garantía de que le atienda su reina del súper particular cuando le toca pagar.

Solo le queda una alternativa. Cuando se aproxima a la fila única, puede estimar cuánto tardará cada cliente en pasar por la caja en base a la cantidad de productos en el carrito y averiguar entonces en qué caja le tocará para ver si es la de su amada o no.

### Entrada

La entrada estará compuesta por distintos casos de prueba, cada uno de ellos representando el estado del supermercado uno de los días en los que Ismael va a comprar.

Cada caso de prueba consta de dos líneas. La primera contiene el número  $N$  de cajas abiertas en ese momento y el número  $C$  de clientes esperando. A continuación viene una línea con  $C$  números positivos que indican el número de segundos que tardará cada cliente en ser atendido. El primer número se corresponde con el tiempo de la primera persona de la fila única.

Al último caso de prueba le sigue uno con 0 cajas y 0 clientes que no debe procesarse.

### Salida

Para cada caso de prueba se escribirá una línea con el número de la caja en la que será atendido Ismael si se coloca detrás de los  $C$  clientes esperando.

Ten en cuenta que las cajas están numeradas de la 1 a la  $N$  y que en caso de quedar dos cajas libres a la vez, el primer cliente irá a la caja con menor número.

### Entrada de ejemplo

```
2 2
10 5
2 2
5 10
3 2
5 10
0 0
```

### Salida de ejemplo

```
2
1
3
```

**Autores:** Marco Antonio Gómez Martín, Pedro Pablo Gómez Martín y Alberto Verdejo.

## Volando drones

Desde hace algún tiempo, quedamos la pandilla para ir a volar drones los sábados. Cada dron necesita dos pilas para poder volar: una de 9 V y otra de 1.5 V. En el club tenemos dos cajas para guardar las pilas, en una tenemos las de 9 V y en otra las de 1.5 V. Cada sábado cogemos las que están más cargadas y las colocamos en los drones. Para aprovechar al máximo el tiempo de vuelo, colocamos siempre las dos pilas más cargadas de cada tipo juntas, ya que los drones solo vuelan mientras las dos pilas tienen carga; después las dos siguientes más cargadas las ponemos en el siguiente dron; y así mientras queden pilas con carga de los dos tipos. Una vez colocadas las pilas, echamos los drones a volar. Cuando todos ellos acaban en el suelo por agotamiento de alguna de sus pilas, volvemos al club y guardamos en las cajas las pilas que todavía no están totalmente gastadas.



Por ejemplo, si a un dron le pusimos una pila de 9 V que permitía volar 5 horas y una pila de 1.5 V que permitía volar 2, el dron habrá volado 2 horas y al volver al club guardaremos la pila de 9 V a la que le quedarán 3 horas de vuelo. La pila de 1.5 V estará agotada y la echaremos al cubo de reciclaje.

Queremos saber cuántas horas de vuelo realizarán entre todos los drones cada sábado que podamos salir a volar, antes de que se agoten las pilas que hay ahora mismo en las cajas. Las pilas las tenemos que colocar en el club, por lo que cada dron solo puede volar una vez cada sábado.

### Entrada

La entrada consta de una serie de casos de prueba. Cada caso comienza con una línea en que se indica el número  $N$  de drones que tenemos ( $1 \leq N \leq 1.000$ ), el número  $A$  de pilas de 9 V y el número  $B$  de pilas de 1.5 V ( $1 \leq A, B \leq 200.000$ ). En la línea siguiente se indica el número de horas de vuelo que permite la carga de cada una de las pilas de 9 V y en la tercera línea el número de horas de vuelo que permite cada una de las pilas de 1.5 V.

### Salida

Para cada caso de prueba se escribirá en una línea el número de horas de vuelo que realizarán los drones cada sábado, mientras se pueda salir a volar algún dron.

Se garantiza que esos números nunca serán mayores que  $10^9$ .

### Entrada de ejemplo

```
2 4 2
5 12 7 15
10 10
2 4 3
5 12 7 15
20 20 2
3 3 3
25 15 10
20 20 5
1 4 6
5 9 2 6
7 3 3 1 6 4
```

### Salida de ejemplo

```
20
27 12
40 5
7 6 4 2 2 1
```

**Autores:** Isabel Pita y Alberto Verdejo.

# 25

## Pájaros en vuelo

Es muy conocido que algunas especies de aves vuelan creando una formación en V. La razón es simple: de esa forma las aves que van detrás aprovechan el rebufo de las que van delante. Esto implica que el pájaro que hace más esfuerzo es aquél que va primero ocupando el vértice de la V, por lo que es importante elegir bien quién es.



Algunas especies lo que hacen es colocarse en orden de edad de izquierda a derecha, de forma que el pájaro que queda en el centro es el que no es ni demasiado joven ni demasiado viejo. De esta forma evitan que el que abre camino esté poco desarrollado o demasiado cansado para tirar del grupo. Cuando a la bandada

se van uniendo más pájaros, ocupan su lugar y, si es necesario, el pájaro que abre el camino cambia.

Por ejemplo, imaginemos que hay una bandada en vuelo con pájaros de edades 10, 20 y 30 meses. En ese caso, el que ocupará el primer lugar es el de 20 meses. Si ahora llegan dos pájaros nuevos, uno de 25 meses y otro de 35, ambos ocuparán su sitio y el primero pasará a ser el de 25 meses. Si posteriormente llega un abuelo de 40 meses con su nieto de 5, ocuparán los extremos y el primer puesto no variará.

Lo que queremos es simular la formación de una de estas bandadas de pájaros que comienza con un único ejemplar y a la que se van añadiendo nuevas parejas. Cada vez que se añade una pareja queremos determinar la edad del ave que ocupa la punta de la bandada.

### Entrada

La entrada estará compuesta por una serie de casos de prueba, cada uno con la información de creación de una bandada de pájaros. Cada caso consta de dos líneas: la primera contiene la edad del primer pájaro que echa a volar seguido del número de parejas que se irán uniendo después (al menos una y hasta 100.000 parejas); en la línea siguiente aparecen las edades de cada pareja.

Se garantiza que las edades de cada uno de los pájaros que entran en la bandada son distintas. Para eso éstas vienen expresadas en segundos, y nunca será un número mayor a 100.000.000.

La entrada termina con una línea con dos ceros.

### Salida

Para cada caso de prueba se escribirá una línea que tendrá tantos números como parejas se incorporan a la bandada, indicando la edad del pájaro que ocupa la primera posición tras la incorporación de cada pareja.

### Entrada de ejemplo

```
30 3
10 20 35 25 5 40
0 0
```

### Salida de ejemplo

```
20 25 25
```

**Autor:** Marco Antonio Gómez Martín.