

# Programa-Me

“First, solve the problem. Then, write the code” John Johnson.

## Programa-Me 2022 Final Nacional Problemas

Concurso online *multisitio*



Ejercicios realizados por



Coordinado desde la **Facultad de Informática (U.C.M.)**  
28 de abril de 2022



28 de abril de 2022  
<http://www.programa-me.com>

## Listado de problemas

A. Arca de Noé	3
B. Bandurria Hero	5
C. Cincuecento	7
D. División exacta	9
E. El telegrama más corto	11
F. Familia numerosa	13
G. Gorroneando helado	15
H. Hipnosis	17
I. Incidencia acumulada	19
J. Julia robó pan	21
K. Kilos de basura	23

Autores de los problemas:

- Marco Antonio Gómez Martín (Universidad Complutense de Madrid)
- Pedro Pablo Gómez Martín (Universidad Complutense de Madrid)

**Revisores:**

- Alberto Verdejo López (Universidad Complutense de Madrid)





# Arca de Noé

Con el cielo amenazando lluvia, el caos alrededor del arca de Noé está llegando a unos límites inquietantes. Se ha ido embarcando a animales de muchas especies y el lío es tremendo. Ya no se sabe si se han dejado subir a individuos de la misma especie y sexo, si hay una pareja de cada especie, si hay animales sin pareja... ¡El reinicio del mundo tras el diluvio corre grave peligro!



Afortunadamente Noé es una persona organizada y ha ido apuntando religiosamente en una lista cada animal que subía a su arca. Además, en su antiguo idioma los nombres de los animales macho acaban con -o, los nombres de los animales hembra terminan en -a, y tienen el resto de las letras exactamente iguales entre sí.

## Entrada

El programa deberá procesar múltiples casos de prueba leídos de la entrada estándar. Cada uno comienza con un número  $1 \leq N \leq 10.000$  indicando cuántos animales se han embarcado en el arca hasta ahora. A continuación aparecen  $N$  palabras, de como mucho 20 letras minúsculas, indicando el nombre de cada uno.

La entrada termina con un 0.

## Salida

Por cada caso de prueba el programa escribirá el número de parejas de animales que están completas (hay un ejemplar macho y otro hembra), cuántas parejas están incompletas (falta el ejemplar macho o hembra) y cuántos animales sobran por haber ya algún otro de la misma especie y sexo. Los tres valores se escribirán en la misma línea separados por un espacio.

## Entrada de ejemplo

```
4
gato perra gata perro
4
gato perra gato perro
6
coneja perro perra gata mono perro
0
```

## Salida de ejemplo

```
2 0 0
1 1 1
1 3 1
```



## ● B

# Bandurria Hero

Albert Una es aficionado a los juegos de ritmo y orgulloso poseedor de reliquias como Gitar Hero, Rock Band o Guitar Master, con todos los cachivaches de las distintas plataformas en las que se publicaron.

Pero su verdadera afición es la bandurria. Poder tocar en su consola de videojuegos canciones clásicas de la tuna como “Clavelitos”, “Cielito lindo” o “De colores” le haría recordar otros paisajes y otros tiempos... y conseguir muchos más puntos que con cualquier otro juego.

De modo que, ni corto ni perezoso, está aprendiendo a programar para poderse hacer su propio “Bandurria Hero”. De momento, está atascado con el cálculo de la puntuación por culpa de los *combos*. Quiere que si se consigue dar correctamente una nota se sumen 10 puntos. Si a continuación se acierta con la siguiente nota, esta se premie con 20 puntos, la siguiente con 30 y así sucesivamente, incrementando la valoración de 10 en 10, siempre que no se falle. En cuanto se pulse incorrectamente una cuerda o se salte una nota, el *combo* se termina y la siguiente nota correcta volverá a ser valorada con 10 puntos.



### Entrada

El programa deberá leer, de la entrada estándar, un primer número indicando cuántos casos de prueba vendrán a continuación.

Cada caso de prueba consiste en una línea compuesta de una secuencia de “0” (vocal o mayúscula) y “.” indicando, respectivamente, una nota correcta o un fallo. Todos los casos tienen al menos un carácter y ninguno tiene más de 10.000.

### Salida

Por cada caso de prueba se indicará la puntuación que se debe dar al jugador teniendo en cuenta la valoración de los *combos*, donde la primera nota de una serie consecutiva de aciertos se valora con 10 puntos y las siguientes notas de la serie proporcionan 10 puntos más que la anterior.

### Entrada de ejemplo

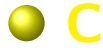
```
4
0.0.
00...
00.00..
.0000...
```

### Salida de ejemplo

```
20
30
60
100
```







# Cinquecento

El *cinquecento* es un periodo del arte europeo (principalmente italiano) enclavado en pleno renacimiento. Aunque su nombre esconde el número cinco, en realidad ¡pertenece al siglo XVI! *Cinquecento* es, abreviadamente, “años [mil] quinientos”, en italiano, y es que el siglo XVI comprendió los años desde el 1501 al 1600, igual que el siglo XXI empezó en el 2001, con un 20 en sus dos primeros dígitos y no un 21.



Dado un año, ¿de qué siglo es?

## Entrada

La entrada comienza con un número indicando cuántos casos de prueba deberán ser procesados. Cada uno es un número entre 1 y 10.000.

## Salida

Por cada caso de prueba el programa escribirá el siglo al que pertenece el año de la entrada. Por simplicidad, se escribirá con números normales (arábigos) en lugar de números romanos.

## Entrada de ejemplo

```
3
1582
2000
2101
```

## Salida de ejemplo

```
16
20
22
```



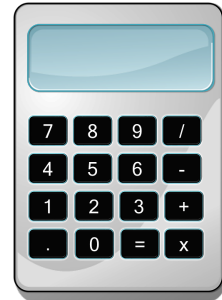


# División exacta

¡Las calculadoras mienten!! Si pruebas a dividir 2 entre 3 recibirás algo como 0.6666667 que, claramente, no está bien. La respuesta correcta no acaba con un 7, sino que está compuesta de una secuencia infinita de “seises”. Se dice que el número es *periódico* y se escribe poniendo una línea encima del dígito que se repite continuamente,  $0.\overline{6}$ .

A veces no se repite solamente un dígito, sino varios. Ocurre por ejemplo con algo tan simple como  $2/7$ , cuyo resultado es  $0.\overline{285714}$ . También puede ocurrir que no se repitan todos los dígitos, sino que haya algunos únicos al principio, y luego ya otros que se repitan, como en  $1/6$  ( $0.1\overline{6}$ ).

Dados dos números, ¿puedes calcular su división exacta?



## Entrada

Cada caso de prueba está compuesto por el numerador y el denominador de una fracción. Se garantiza que ambos estarán entre 1 y 10.000 y que el numerador será siempre menor que el denominador, de modo que la división será menor que 1.

La entrada termina con dos ceros.

## Salida

Por cada caso de prueba el programa escribirá la división completa de los dos números. Si la división tiene una parte periódica, esta se escribirá entre dos barras verticales.

## Entrada de ejemplo

```
2 3
2 7
1 6
4 8
0 0
```

## Salida de ejemplo

```
0.|6|
0.|285714|
0.1|6|
0.5
```





## El telegrama más corto

Aunque hoy su uso está en horas bajas, el sistema de telégrafo fue el medio principal de comunicación durante el final del siglo XIX y la primera mitad del XX<sup>1</sup>. Para enviar letras y símbolos, se utilizaba *código Morse*, que codificaba cada elemento como una sucesión de *puntos* y *rayas*<sup>2</sup>. Al menos esa es la creencia popular porque, en realidad, ambos son meros “pitidos” enviados por un cable con la característica de que el pitido asociado a una raya es tres veces más largo que el de un punto. Se conocen como puntos y rayas porque, en el receptor, esos pitidos ocasionaban perforaciones en una cinta de papel en movimiento, lo que originaba zonas rasgadas cortas (puntos) y largas (rayas).

Letra	Código	Letra	Código
A	.-	N	-.
B	-...	O	---
C	-.-.	P	.--.
D	-..	Q	--.-
E	.	R	.-.
F	..-.	S	...
G	--.	T	-
H	....	U	..-
I	..	V	...-
J	.---	W	.-.
K	-. -	X	-.-.
L	.-..	Y	-.--
M	--	Z	--..
!	-.-.--	?	..--..

El sistema era tan artesanal que el servicio se pagaba por número de palabras enviadas. Eso creó un lenguaje propio del telegrama, evitando palabras superfluas e incorporando abreviaturas particulares, como ocurriría mucho tiempo después con otras tecnologías como los SMS o Twitter.

Dice la leyenda que el telegrama más corto de la historia lo envió Victor Hugo. Pese a ser amante del subjuntivo, las frases subordinadas y los libros largos, cuando llegó el momento de preguntar por telegrama a su editor cómo estaban yendo las ventas de su recién publicado libro “Los miserables”, decidió dejar su rimbombante prosa para otra ocasión y envió un simple “?”.

Para saber si realmente Víctor Hugo envió el telegrama más corto de la historia hay que recurrir a la *foto finish*. Su editor, a la vista de las buenas ventas del libro, le respondió con un simple “!”.

### Entrada

La entrada comienza con un número indicando cuántos casos de prueba deberán ser procesados.

A continuación aparece una línea por cada caso de prueba, conteniendo una frase de no más de 80 letras mayúsculas, signos de admiración e interrogación y espacios. Se garantiza que no aparecen dos espacios seguidos, y que la línea no empieza ni termina en espacio.

### Salida

Por cada caso de prueba el programa escribirá la duración *en puntos* del telegrama. Según la publicación de 1922 del estándar del código Morse, una raya dura tres puntos, entre símbolo y símbolo (raya o punto) de una misma letra hay que esperar un punto, la separación temporal entre letras son tres puntos, y la separación entre palabras cinco.

<sup>1</sup>Hoy en día, en muchos países el servicio de telégrafos, directamente ya no existe. En España se mantiene por su valor legal al dejar constancia de su envío y de su contenido, igual que los *burofaxes*.

<sup>2</sup>Esto ya tampoco es cierto. Hoy los “telegramas” son documentos PDF firmados digitalmente que viajan por redes de datos.

### Entrada de ejemplo

```
4
?
!
SI
YA NACIO
```

### Salida de ejemplo

```
15
19
11
73
```



## Familia numerosa

En la familia de Al von Dighas son un montón de hermanos. Cuando en familias tan numerosas se pone la comida en el centro para compartir, suele surgir una carrera subconsciente para comer más. Es muy habitual que haya adultos que comen muy deprisa y se deba a haber vivido una infancia con muchos hermanos.

Pero la familia von Dighas no es así. Son ordenados y reparten las cosas de la forma más equitativa posible. Hoy hay albóndigas para comer y Al, que tenía que irse pronto, ha comido solo. Acostumbrado a repartir, ha contado cuántas albóndigas había en la cazuela, ha dividido el total entre el número de hermanos y se ha comido las que le correspondían.

Cuando el resto se ha sentado a la mesa, ha contado las albóndigas restantes y se han hecho una pregunta: ¿cuántas había al principio?



### Entrada

La primera línea de la entrada contiene un número con la cantidad de casos de prueba que el programa deberá procesar.

Cada caso de prueba está compuesto por dos números,  $c$  y  $n$ , indicando el número de comensales (incluyendo al propio Al) y la cantidad de albóndigas que quedaban después de que Al se comiera las que les correspondían. Al menos hay dos comensales y ninguno de los dos números es mayor que 100.000.

### Salida

Por cada caso de prueba el programa escribirá dos números separados por espacio indicando la menor y mayor cantidad posible de albóndigas que había al principio. Al ha cogido las que le correspondían sin pasarse. Eso significa que si la división no era exacta, ha cogido algo menos de lo que le tocaba para no tener que partir una albóndiga.

### Entrada de ejemplo

```
3
2 2
3 3
4 12
```

### Salida de ejemplo

```
3 4
4 4
15 16
```







# Gorroneando helado



Heladio, Laura y Domingo son tres devoradores de helados. Les gustan todos los sabores, desde clásicos como vainilla o chocolate, hasta cosas esotéricas como *nubes* o palomitas de maíz, sin olvidarse de sabores “de nicho” como turrón o ron con pasas.



Hoy están asistiendo a una celebración donde hay *Viennetta*, esas barras de helado de nata con láminas de chocolate para cortar en raciones. Los anfitriones han puesto a disposición de los asistentes la barra más grande del mercado y la han enriquecido con fresas y frutos rojos, creando una combinación difícil de olvidar.

Heladio, Laura y Domingo tienen la intención secreta de comerse la gigante barra de *Viennetta* entre los tres pero, dado que los frutos rojos les gustan todavía más que el propio helado, han llegado a un acuerdo. Partirán, cuando nadie mire, la barra de *Viennetta* en tres grandes raciones de modo que el número de frutos rojos que quede en cada una sea lo más parecido posible al de las demás. Eso significa que la resta entre el número de la ración más poblada y el de la menos debe ser lo menor posible.

## Entrada

Cada caso de prueba comienza con un número  $3 \leq n \leq 200.000$  con el número de porciones de la barra de helado. A continuación aparecen  $n$  números entre 1 y 10.000 con la cantidad de frutos rojos en cada una.

La entrada termina con un 0.

## Salida

Por cada caso de prueba el programa decidirá cuántas porciones debe comerse cada amigo para que los tres coman el número de frutos rojos lo más parecido posible. Recuerda que la barra de *Viennetta* se partirá en tres grandes raciones, por lo que las porciones que cada uno comerá serán consecutivas en la entrada.

El programa escribirá, para cada caso, la diferencia entre el número de frutos rojos que come la persona a la que más le corresponden y la que menos come con el corte más equitativo posible del helado. Recuerda que lo importante no es el número de porciones que toma cada uno (que podría ser muy distinto), sino el de frutos rojos.

## Entrada de ejemplo

```
3
1 1 1
4
1 1 2 1
6
1 2 3 1 2 3
6
3 1 1 1 1 2
0
```

## Salida de ejemplo

```
0
1
2
0
```



# ● H

## Hipnosis

La semana pasada mi grupo de amigos y yo fuimos de público a un programa de variedades de la tele. En un determinado momento, salió un hipnotizador que pidió un voluntario. Luis, al que llamamos *sleepy*, estaba aburrido medio dormido, y entre varios le levantamos el brazo y le intentamos poner de pie para que le cogieran. ¡¡Y resultó!! Creo que hicimos un favor al hipnotizador, porque el voluntario le dio la mitad del trabajo hecho.



Usando sus técnicas de persuasión, el hipnotizador se las arregló para que Luis se olvidara del número 6. Luego le pidió que se contara los dedos de las manos. Como había olvidado el 6, tras el número 5 saltaba al 7 y, claro, al terminar la cuenta ¡le salían 11 dedos! Al descubrir que tenía un dedo de más, recontaba y recontaba con pavor no entendiendo de dónde le había salido el dedo sobrante.

Tras el espectáculo, busqué rápido al hipnotizador para felicitarle y le pregunté qué habría pasado si le hubiera hecho contar todos los dedos, incluidos los de los pies. Me dijo que se habría también saltado el número 16.

Pensando después, supuse que también se habría saltado los números del 60 al 69 o todos los números entre 6.000 y 6.999.

¡Esto de la hipnosis es increíble! Nos reímos un montón. Todos salvo *sleepy*, claro, al que ahora llamamos *eleven*.

### Entrada

El programa deberá leer, de la entrada estándar, múltiples casos de prueba. Cada uno contiene dos números  $1 \leq N \leq 10^9$  y  $1 \leq D \leq 9$ .

La entrada termina con dos ceros, que no deben procesarse.

### Salida

Por cada caso de prueba, el programa escribirá el número al que habría llegado Luis contando si el hipnotizador le hubiera pedido que olvidara el dígito  $D$  y que contara un conjunto de  $N$  cosas.

### Entrada de ejemplo

```
10 6
23 2
987654321 1
0 0
```

### Salida de ejemplo

```
11
36
3595502030
```





# Incidencia acumulada

La *incidencia acumulada* es un número que se utiliza para conocer el nivel de contagio de una determinada enfermedad entre la población. Es el número de nuevos enfermos por cada 100.000 habitantes en un tiempo determinado. Así, por ejemplo, la incidencia acumulada a 7 días de una determinada enfermedad es el número de nuevos casos que han ocurrido en los últimos 7 días por cada 100.000 habitantes. Se calcula sumando cuanta gente ha enfermado en los 7 días previos, dividiendo el valor por la población total y multiplicándolo por 100.000.



Cedar Creek es una pequeña población californiana con exactamente 100.000 habitantes a la que ha llegado un nuevo virus, el *motaba*, debido a un mono traído ilegalmente desde Zaire. Las autoridades sanitarias han ido apuntando los nuevos casos diarios, pero no han calculado la incidencia acumulada.

## Entrada

Cada caso de prueba comienza con dos números  $2 \leq i \leq n \leq 500.000$  que indican, respectivamente, la longitud en días de la incidencia acumulada que se quiere calcular, y el número de días de los que se conoce el número de casos nuevos. A continuación aparecen  $n$  números separados por espacio, con la cantidad de casos nuevos diarios. Una persona infectada puede recuperarse rápido, recaer posteriormente ¡y contar en la incidencia acumulada más de una vez!

La entrada termina con dos ceros.

## Salida

Por cada caso de prueba se escribirá la máxima incidencia acumulada a  $i$  días que se ha sufrido en Cedar Creek.

## Entrada de ejemplo

```
3 3
1 3 5
2 4
1 3 5 2
0 0
```

## Salida de ejemplo

```
9
8
```





# Julia robó pan

Hay una vieja *cancioncilla* que los conductores de autobús escuchan una y otra vez cuando les toca llevar de excursión a colegiales. En ella se establece una especie de conversación entre una persona y el resto del grupo:

- [Grupo] Julia robó pan en la fiesta de San Juan.
- [Julia] ¿Quién? ¿Yo?
- [Grupo] ¡Sí, tú!
- [Julia] ¡Yo no fui!
- [Grupo] ¿Entonces quién?
- [Julia] Mmmmmm... ¡Andrés!



En ese momento el acusado pasa a ser Andrés; el grupo canta de nuevo el “Andrés robó pan en la fiesta de San Juan.”, Andrés es el que se defiende, acusa a otro y así sucesivamente.

A la larga, claro, llega un momento en que una persona es acusada por segunda vez. La regla no escrita es que en ese caso, cuando le toca decir el nombre del siguiente, no puede decir al mismo que dijo la primera vez; si todos hicieran lo mismo se terminaría en un ciclo que se repetiría indefinidamente.

En la excursión de hoy, no obstante, esa regla no se cumple. Todos los excursionistas tienen un “*mejor amigo*” que es al que acusarán si les llega el turno. Eso significa que, da igual quién sea el primer sospechoso de robar pan, el juego terminará en un ciclo sin fin. Dada la lista de mejores amigos, ¿cuántos de tales ciclos hay?

## Entrada

La entrada estará compuesta por distintos casos de prueba cada uno en dos líneas.

Cada caso de prueba representa una excursión y comienza con el número  $N$  de colegiales (hasta 10.000). A continuación viene una línea con  $N$  números, uno por viajero. El número  $i$ -ésimo representa el mejor amigo de la persona  $i$  (las personas están numeradas entre 1 y  $N$ ). Se garantiza que ningún excursionista se acusa a sí mismo.

## Salida

Por cada caso de prueba se escribirá una línea con un único entero indicando el número de finales posibles distintos, es decir en cuántos ciclos diferentes se puede terminar.

## Entrada de ejemplo

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

## Salida de ejemplo

```
1
2
1
```





# ● K

## Kilos de basura

Todas la noches igual. Cuando el cubo de basura se llena y toca salir a la calle para llevarla al contenedor terminamos discutiendo. Ninguno queremos ir.

El cubo tiene ya unos añitos, así que dentro de poco tendremos que cambiarlo. Y ya hacía falta, que seguimos con un único cubo por lo que hasta ahora no separábamos los envases. Nuestro objetivo es comprar un cubo de esos que admiten dos bolsas, una de envases y otra para el resto de basura, que nos garantice que el número de veces que tendremos que salir a la calle a vaciarlo no supere un umbral.

Para lograrlo llevamos tiempo analizando nuestros hábitos de consumo y de generación de residuos para escoger el cubo más apropiado. Durante estos últimos meses hemos ido apuntando todo lo que hemos ido tirando y tenemos una lista con lo que ocupa cada cosa y si iría a la bolsa de envases o a la otra.

El siguiente paso es decidir la capacidad que tiene que tener el cubo. Hemos visto que todos los que hay en el mercado con dos zonas tienen la zona de los envases con el doble de capacidad que la del resto de residuos (suponemos que será porque los envases en general ocupan más pero pesan menos). Para que no nos ocupe mucho en casa lo queremos lo más pequeño posible.



### Entrada

La entrada está compuesta por distintos casos de prueba.

Cada caso de prueba comienza con una línea con dos números,  $K$  y  $N$ . El primero (entre 1 y 1.000) indica el número máximo de veces que estamos dispuestos a sacar la basura. El segundo (entre 1 y 200.000) indica la cantidad de datos recogidos.

A continuación aparecen  $N$  líneas con la información de las cosas que hemos ido tirando. Cada una comienza con un número indicando su tamaño (hasta 10.000) seguido de un carácter que puede ser E o R indicando que se tira un envase o un residuo genérico.

Tras el último caso de prueba viene una línea con dos ceros que no debe procesarse.

### Salida

Por cada caso de prueba se escribirá la capacidad mínima de la bolsa de residuos genéricos que tiene que tener el cubo para que no necesitemos sacar la basura más de  $K$  veces para deshacernos de los  $N$  residuos. Hay que tener en cuenta que la capacidad se da siempre como un número entero y que salimos a tirar la basura cuando al ir al cubo a dejar algo, no entra; en ese momento salimos al contenedor, tiramos las dos bolsas y a la vuelta ponemos bolsas nuevas y “estrenamos” la que corresponda con aquello que no entró. El cubo comienza vacío y al final debe quedar vacío también.

### Entrada de ejemplo

```
2 3
10 E
7 R
9 R
2 4
4 R
10 E
4 R
8 E
0 0
```

## Salida de ejemplo

9
5