

$$1^a \quad 6 \% 2 = 0$$

$$p = 1$$

$$2^a \quad 5 \% 2 \neq 0$$

$$\text{numParejas} = 1$$

$$3^a \quad 4 \% 2 = 0$$

$$p = 2$$

$$4^a \quad 3 \% 2 \neq 0$$

$$\text{numP} = 3$$

## Parejas de impares y pares

Dado un vector  $v$  de números enteros positivos distintos, queremos calcular cuantas parejas  $(x, y)$  podemos formar tal que  $x$  aparezca antes que  $y$  en el vector,  $x$  sea impar e  $y$  sea par.

Por ejemplo, dado el siguiente vector:

[1, 2, 3, 4, 5, 6]

podemos formar 6 parejas que cumplen las condiciones anteriores:

(1, 2) (1, 4) (1, 6) (3, 4) (3, 6) (5, 6)

Especifica e implementa una función eficiente que, dado un vector de números enteros positivos distintos, calcule cuántas parejas se pueden formar que cumplan las condiciones anteriormente indicadas. Escribe como comentarios en el código el invariante y la cota de los bucles. Finalmente, justifica el coste del algoritmo en el caso peor.

### Entrada

El primer número de la entrada indica cuántos casos de entrada vendrán a continuación, cada uno en una línea distinta.

Cada caso de prueba comienza con el número  $n$  de elementos del vector ( $0 < n \leq 50.000$ ) y a continuación aparecen dichos elementos. Se garantiza que todos los elementos  $x$  del vector son distintos y  $0 < x < 10^6$ .

### Salida

El programa debe imprimir el número de parejas  $(x, y)$  que se pueden crear en cada vector de forma que  $x$  aparezca antes que  $y$  en el vector,  $x$  sea impar e  $y$  sea par. Se garantiza que el número de parejas es inferior a  $10^9$ .

Cada respuesta se imprimirá en una línea distinta.

### Entrada de ejemplo

```
9
3 1 2 3
3 2 4 6
3 1 4 6
5 1 2 3 4 5
5 1 3 5 2 4
5 2 4 1 3 5
5 2 4 2 5 6
10 1 2 3 4 5 6 7 8 9 10
10 10 9 8 7 6 5 4 3 2 1
```

tom.  
vector

Vemos el impar encontrado como  $x$  hace parejas con los pares encontrados previamente.

$\text{if } (v[i] \% 2 == 0)$   
par + r

else

$\text{numParejas} += \text{par};$

Se ve que 1 impar (x) hace pareja con (par) pares  
Aquí hay 3

### Salida de ejemplo

```
1
0
2
3
6
0
1
15
10
```

### Nota

Este ejercicio debe verse en el contexto de la asignatura de Fundamentos de Algoritmia (FAL), FDI-UCM. Por tanto *no* vale cualquier solución, sino sólo aquellas que utilicen los conceptos de la asignatura. Es muy posible que se den aclaraciones adicionales en clase a este respecto.