

Soluciones Curso 13/3

¡Esperamos que os hayan gustado los problemas!

Problema

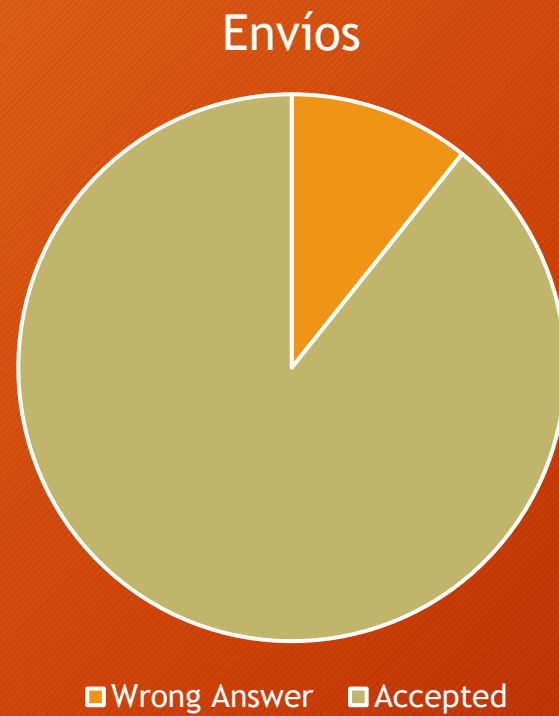
B

Ordenando con Bubble Sort

Autor: Jakub Jan Luczyn

Ordenando con Bubble Sort

B



Total:

28 envíos

25 Accepted

3 Wrong Answer

Ordenando con Bubble Sort

B

El problema más fácil del concurso.

Implementamos ordenación por el método de la burbuja y contamos cuantas veces tenemos que cambiar dos elementos.

El método de la burbuja consiste en recorrer el array N veces, y si un elemento es mayor que el anterior los intercambiamos. De esta forma, comprobamos todo el array buscando el mayor elemento y lo mueves hasta el final.

Problema

C

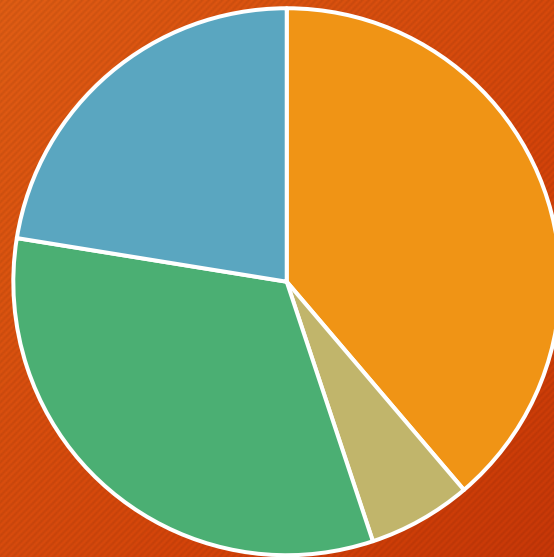
Tim el Hambriento

Autor: José Francisco Rodríguez Peña

Tim el Hambriento

C

Envíos



Wrong Answer RunTime Error Accepted TimeLimit Exceeded

Total:

49 envíos

16 Accepted

19 Wrong Answer

3 RunTime Error

11 Timelimit Exceeded

Tim el Hambriento

C

Tim quiere saciarse comiendo en el restaurante, y quiere hacerlo con un límite de platos.

Como Tim siempre va a preferir comerse los platos que más le sacien, podemos ordenar los platos de mayor a menor e ir cogiendo tantos platos como necesite Tim para saciarse.

Buscar cada vez el plato más grande => TLE ($O(n^2)$)

... ¡Pero se puede mejorar aún más!

Tim el Hambriento

C

Al leer los platos podemos sumar el total de lo que nos va a saciar el restaurante. Si desde el principio, Tim pide más de lo que nos va a dar en total el restaurante... ¡Ni siquiera hace falta ordenar los datos ni recorrer el vector!

En ese caso imprimimos “Sinpa” y seguimos con el siguiente restaurante.

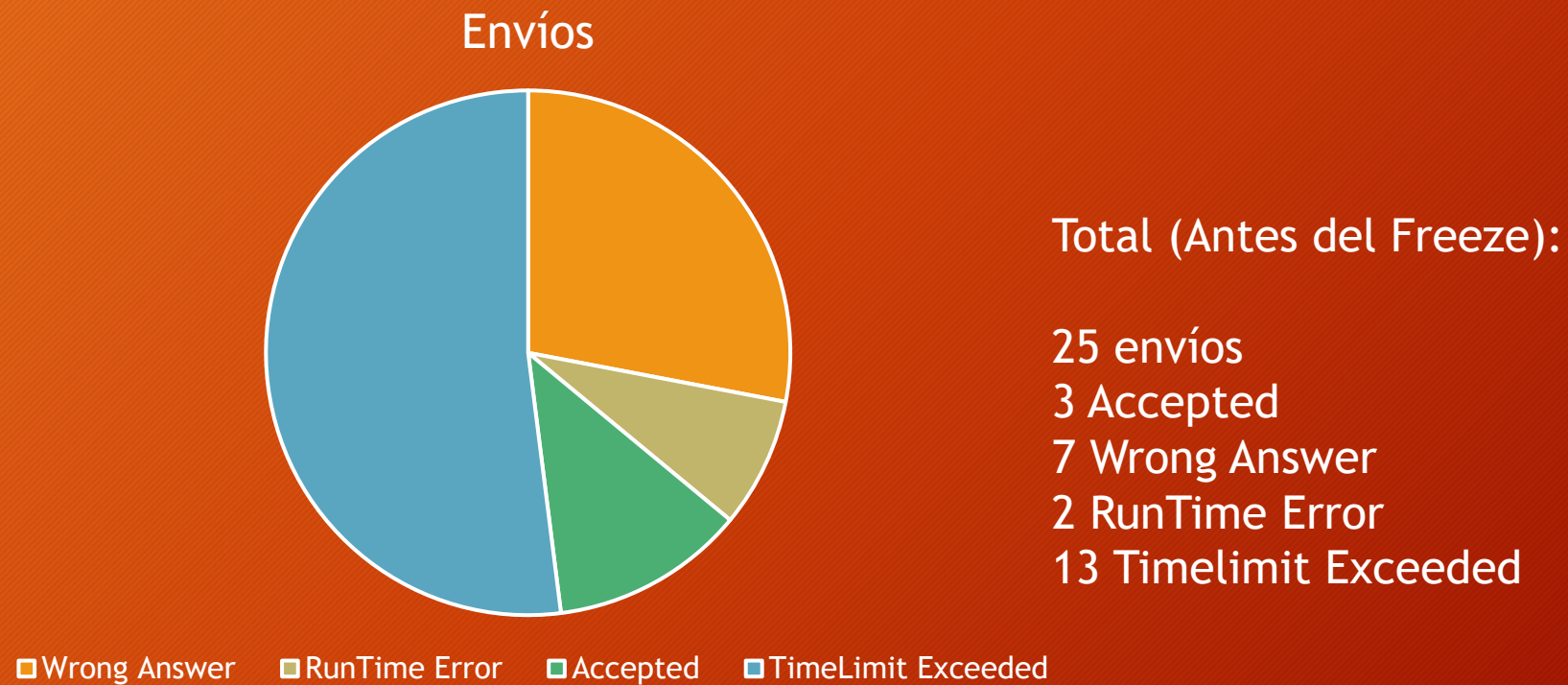
Problema

A

Pánico en el Supermercado
Autor: Iván Martín de San Lázaro

Pánico en el Supermercado

A



Pánico en el supermercado

A

En este ejercicio había dos complicaciones:

La primera, leer el enunciado. Aunque los productos vengan en pack... Todos se dividen desde el principio. No hace falta en ningún caso guardar nada más. Podemos crear un vector de productos, y si viene un producto que ocupa 10 y puede ser dividido 2 veces, añadimos dos productos de tamaño 5 al vector.

Con esto, acabamos con un vector de productos, y luego leemos otro de huecos o espacios en la despensa.

Pánico en el supermercado

A

Y la segunda complicación...

Si hay menos espacios que productos en la despensa... ¡No se debe procesar el caso! Directamente regalamos cosas a la vecina y pasamos al siguiente. No hacerlo => TLE

El otro mayor problema que habéis tenido es que igualabais tamaños a productos... ¡Un producto tamaño 6 cabe en un hueco tamaño 10!

La solución es ordenar de mayor a menor tanto productos como huecos e ir colocando productos.

Problema

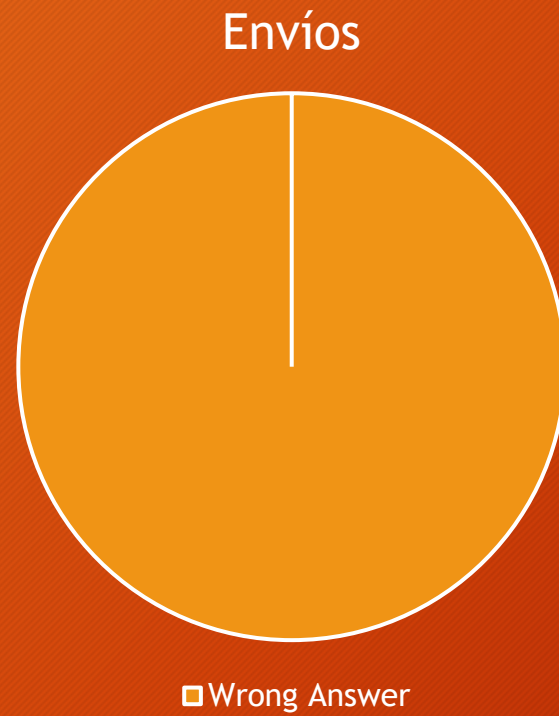
D

Becario Precario Binario

Autor: Jakub Jan Luczyn

Becario Precario Binario

D



Total (Antes del Freeze):

2 envíos

2 Wrong Answer

Becario Precario Binario

D

La solución es una mezcla de búsqueda binaria con greedy.

A este tipo de problemas se le dice que es “Buscar la respuesta en el espacio de soluciones”.

Se debe hacer búsqueda sobre el tamaño de la memoria USB, empezando en 16MB hasta la suma de todos los valores de la entrada.

Para comprobar si un tamaño es válido se puede simplemente simular el número de viajes requerido con el tamaño dado. Si supera el número de viajes máximo, no es viable.

Problema

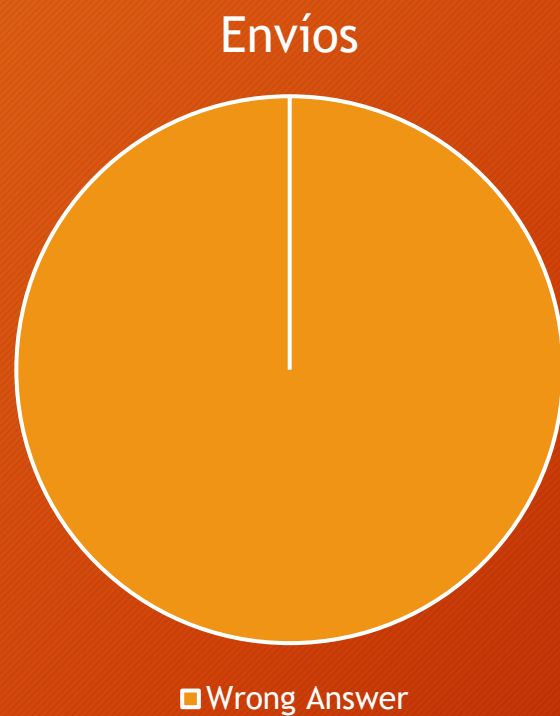
E

Elecciones en la URJC

Autor: Iván Martín de San Lázaro

Elecciones en la URJC

E



Total (Antes del Freeze):

1 envío

1 Wrong Answer

Elecciones en la URJC

E

Este problema tenía muchas cosas que tener en cuenta. Se resumen en 2:

- Como analizar los resultados de cada campus
- Como saber quién ha ganado

Elecciones en la URJC

E

Pensemos en los resultados de cada campus. Observemos el ejemplo

- Opción 1: Se conocen los resultados de todos los candidatos

(Por ejemplo: Vicálvaro)

En este caso, apuntamos que cada candidato tiene dichos votos

Candidato	Votos
Héctor	9
Álvaro	5
María	2
Pablo	4

Elecciones en la URJC

E

Pensemos en los resultados de cada campus. Observemos el ejemplo

- Opción 2: Se conocen los resultados de todos los candidatos menos de uno

(Por ejemplo: Argüelles)

El enunciado explica que cada candidato solo sale una vez

Así que concluimos que el que falte se lleva todos los votos

Candidato	Votos
Héctor	34
Álvaro	15
María	12
Pablo	9

Elecciones en la URJC

E

Pensemos en los resultados de cada campus. Observemos el ejemplo

- Opción 3: Se conocen los resultados de todos los candidatos menos de dos o más

(Por ejemplo: Móstoles)

No tenemos certeza de quién se lleva esos puntos

Así que los anotamos como votos posibles

Candidato	Votos	Posibles
Héctor	64	0
Álvaro	15	50
María	32	0
Pablo	9	50

Elecciones en la URJC

E

Sabiendo esto, contemos los resultados. Parece obvio que Álvaro ha ganado, tiene 65 votos... ¿Pero los tiene de verdad? ¡No! Es una suposición.

De los 50 votos que faltan, se los podría haber llevado Álvaro o Pablo.

Sumando los 50 votos, o bien Álvaro o bien Héctor Podrían haber ganado

Candidato	Votos	Posibles
Héctor	64	0
Álvaro	15	50
María	32	0
Pablo	9	50

Elecciones en la URJC

E

Tras analizar los resultados de cada campus, cada candidato tiene dos cuentas, los votos certificados y los que podría haber conseguido en su mejor posición.

Para saber si un candidato opta a la victoria o no, sumamos sus puntos seguros con sus posibles. Si con ello supera el máximo de los votos seguros de los demás candidatos, podría haberlos superado y es candidato a la victoria.