

04 59 25

Pautas

Área de
codificaciónEditor | Historial de
compilación y
ejecución

Sumisiones

Formulario de
comentarios

Resultado

Tablero

Graficas

Área de codificación

UNA

segundo

do

re

mi

F

EDITOR EN LÍNEA (C)

Actualización de Stock

+ Descripción del problema

Una bolsa de valores procesa N acciones y una vez cada 20 segundos envía los mejores cinco precios y las cantidades totales de pedido por acción. Si bien el tamaño del mensaje para estos datos de mercado es $<S$ kilobytes por stock, el intercambio debe enviarlo como un marco de tamaño fijo de F KB. Como resultado, el consumo de ancho de banda es $(N * S) \text{ KB} / 20 \text{ seg}$ o $(N * S * 8) / 20 \text{ Kbps}$

El intercambio decide utilizar la compresión y el empaquetado de mensajes en marcos, para reducir el consumo de ancho de banda. Dada una lista de mensajes, su tamaño original y su tamaño comprimido, debe empaquetar estos mensajes en marcos F KB fijos de modo que se minimice el número de marcos. Un mensaje no puede cortar marcos. Puede haber espacio perdido por cuadro. Los mensajes se pueden mapear en cualquier orden.

+ Restricciones

 $C \leq S \leq F$ $N \leq 10000$ $F \leq 1024$

+ Entrada

La primera línea contiene un entero N que denota el número de mensajes

La siguiente línea contiene un entero F que indica el tamaño del marco (en KB)

Las siguientes N líneas contienen 2 enteros delimitados por un espacio que indica el tamaño del mensaje original (S) y el tamaño del mensaje comprimido (C), respectivamente. Ambos están en KB

+ Salida

Imprime el número mínimo de marcos requeridos para enviar todos los mensajes

+ Caso de prueba

+ Explicación

Ejemplo 1

Entrada

6

20

10 8

9 8

6 4

15 10

8 6

5 4

Salida

2

Explicación

 $N = 6, F = 20.$

For packing the messages in the frame we shall consider only the compressed message sizes. Since the total size of the 6 compressed messages is 40 KB and the bin size is 20 bytes, the optimal number of frames required to pack all message without splitting any message across frames, is 2. We need to see if this is possible. One way of packing them is to put the first three messages into one ($8+8+4=20\text{KB}$) into one frame, and the last three into the second frame ($10+6+4=20\text{KB}$). As the optimal compression is possible, this is the minimum number of frames in which it can be packed. Hence the output is 2.

Example 2

Input

7

20

10 8

9 8

6 4

15 10

8 6

5 4

10 9

Output

3

Explanation

 $N=7, F = 20$

For packing the messages in the frame we shall consider only the compressed message sizes. Since the total size of the 7 compressed messages is 49 kB and the bin size is 20 kB, the optimal number of frames required to pack all message without splitting any message across frames, is 3. One way of packing them into frames is to pack the first three into one frame ($8+8+4=20$), the next 3 into the second frame ($10+6+4=20$) and the last in a third frame. As the packing into frames can be achieved in the optimal number of frames, 3 is the output

Upload Solution [Question : C]

☐ I, **Christian** confirm that the answer submitted is my own. ☐ Took help from online sources (attributions)

Choose a
File ...

[CodeVita FAQ's](#)
[About CodeVita](#)
[Privacy Policy](#)
[Careers](#)



© 2018 Tata Consultancy Services Limited. All Rights Reserved.