Realizar un programa que solucione el siguiente problema:

Wilson trabaja para una compañía de mudanzas. Su principal tarea es cargar elementos del hogar en un camión. Wilson tiene una bolsa que él usa para mover estos elementos. El coloca un puñado de elementos en la bolsa, luego los mueve al camión y los lleva al destino.

Wilson tiene una reputación como trabajar perezoso. Julie es la supervisora de Wilson y ella está encargada de que Wilson no haga trampa. Ella quiere que Wilson lleve al menos 50 libras de elementos en su bolsa cada vez que el vaya al camión.

Por suerte para Wilson, su bolsa es opaca. Cuando el lleva una bolsa llena de elementos, Julie puede descifrar cuántos elementos hay en la bolsa (basada en la altura de la pila de elementos en la bolsa) y también el peso del elemento que está en la parte superior de la pila. Sin embargo ella no puede descifrar cuánto pesan los demás elementos de la bolsa. Ella asume que cada elemento en la bolsa pesa al menos tanto como el elemento que está en la parte superior, porque seguramente Wilson, como es tan perezoso, no sería tan tonto como para colocar elementos más pesados encima de elementos menos pesados. Julie es lamentablemente ignorante de la extensiva falta de dedicación de Wilson en sus tareas y sus suposiciones son frecuentemente incorrectas.

Hoy hay **N** elementos para ser movidos y Wilson, a quien se le paga por hora, quiere maximizar el número de viajes para colocar todos los elementos en el camión ¿cuál es número de viajes máximo que Wilson puede hacer sin ser reprendido por Julie?

Cabe notar que Julie no es consciente de que elementos van a ser movidos hoy y ella no mantiene la cuenta de cuantos elementos Wilson ha movido cuando examina cada bolsa. Ella simplemente asume que cada bolsa llena de elementos contiene al menos un peso total de **k \* w** donde **k** es el número de elementos en la bolsa y **w** es el peso del elemento que está en la parte superior.

**Entrada.**

La entrada comienza con un entero **T**, el número de días que Wilson “trabaja” en su trabajo. Para cada día hay una línea que contiene el entero **N**, después hay **N** líneas, donde cada línea subsiguiente contiene el peso del item **Wi**

**Salida**

Para cada día i, imprimir una línea que contenga “Case #i:” seguido de el máximo número de viajes que Wilson puede hacer ese día.

**Restricciones.**

1 ≤ **T** ≤ 500

1 ≤ **N** ≤ 100

1 ≤ **Wi** ≤ 100

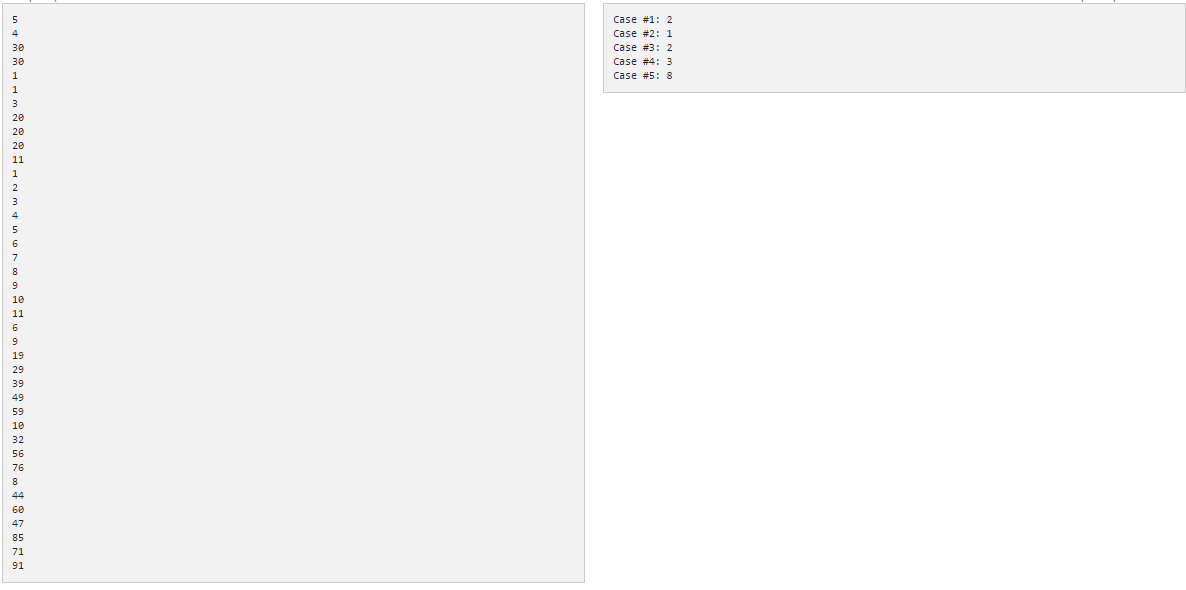
En cada día, es garantizado que el peso total de todos los elementos es al menos 50 libras.

**Explicación del ejemplo**

En el primer caso Wilson puede hacer dos viajes, colocando un elementos que pesa 30 encima de un elemento que pesa 1, haciendo parecer que contiene al menos 60 libras en cada viaje.

En el segundo caso, Wilson necesita poner todos los elementos en la bolsa de una vez y solo puede hacer un viaje.

En el tercer caso una posible solución es poner los elementos con peso par en la bolsa para el primer viaje y los elementos impares para el segundo viaje, asegurando poner los elementos más pesados en la parte superior.



Nota: Asegure que el programa siga el formato otorgado.

Los archivos lazy\_loading\_example\_input y lazy\_loading\_example\_output son un ejemplo de una posible entrada y una posible salida para el programa.

**NOTA: Debe ser una aplicación web que tenga una pantalla en la que sus entradas sean la cédula del participante y el archivo a procesar, las salida debe ser el archivo con el proceso terminado, se debe crear una tabla donde se guarde una traza de cada ejecución con la cédula del ejecutor y la fecha de ejecución.**

**Realizar el proyecto con la mejor elección tecnológica y mejores prácticas de desarrollo.**