

RapidLight

El problema

La empresa RapidLight se dedica al transporte de mercancías por carretera. La empresa dispone de vehículos con distintas capacidades de depósito, por lo que está interesada en disponer de un programa que, dado un plan de transporte, les permita determinar la capacidad mínima del vehículo necesaria para llevarlo a cabo. Más concretamente, un plan de transporte consta de:

- Una serie de etapas que deben completarse. Cada etapa requiere, para su realización, un determinado volumen de combustible. Asimismo, al final de cada etapa será posible, si fuera necesario, volver a llenar el depósito del vehículo.
- Un número máximo de repostajes permitido, ya que cada repostaje supone invertir tiempo, y uno de los objetivos de RapidLight es transportar las mercancías en el menor tiempo posible.

Se pide desarrollar un algoritmo “divide y vencerás” que determine la capacidad mínima del vehículo necesaria para ejecutar un plan de transporte, dados: (i) la lista de consumos requeridos por las etapas que componen el viaje; y (ii) el número máximo de repostajes permitidos en el camino. Aparte de desarrollar el algoritmo, deberás determinar razonadamente su complejidad.

Ten en cuenta que el camión comenzará su andadura con el depósito lleno y le basta con llegar con la gasolina justa a su destino.

Programa de prueba

El algoritmo se utilizará desde un programa de prueba que lea, por la entrada estándar, planes de transporte, y, para cada plan leído, invoque al algoritmo, y escriba la capacidad mínima necesaria del vehículo para llevar a cabo dicho plan. Cada plan de viaje consta de dos líneas: En la primera aparece el número de etapas del viaje y el número máximo de repostajes permitido. En la segunda aparecen, en orden, el consumo de combustible requerido por cada etapa. La entrada termina con una línea que contiene -1.

A continuación, se muestra un ejemplo de entrada/salida:

Entrada	Salida
2 1	6
6 4	14
3 0	9
3 6 5	
3 1	
3 6 5	
-1	

Trabajo a desarrollar

Tu trabajo consiste en:

- Diseñar y analizar el algoritmo, rellenando los huecos correspondientes, entre comentarios, que preceden a la función `min_capacidad`
- Codificar el algoritmo. El punto de entrada es la función `min_capacidad`. Aparte, podrás definir las funciones auxiliares que consideres oportuno.
- Entregar `dv.cpp` a través del juez en línea de la asignatura.

Importante:

- **Sólo puntuarán aquellas entregas que superen los casos del juez (el resto de entregas puntuarán 0).**
- **Debes poner tu nombre y apellidos en el comentario habilitado para ello en la plantilla (aquellas entregas no identificadas con nombre y apellidos puntuarán 0)**
- No modificar el código proporcionado. Únicamente deben responderse a los distintos apartados, en el interior de los comentarios, e implementar la función `min_capacidad` (sí pueden incluirse funciones auxiliares, si se considera necesario).