

## El viajante avaricioso

Amancio Ortega ha conseguido un nuevo trabajo como comercial. Está un poco descontento con el puesto, porque le han dicho que su sueldo no va a depender de las ventas (que es lo que se le da bien), sino que le van a pagar por cada kilómetro que haga (1€ por cada kilómetro, tampoco nos pasemos). Como Amancio necesita dinero para emprender, va a intentar siempre hacer **el máximo de kilómetros posible**. El único requisito que debe cumplir es comenzar en el centro de distribución del producto (nodo 0) y volver a él al final de la ruta. Podemos asumir que desde un punto de venta se puede llegar a todos los demás, pero no podemos visitar dos veces el mismo punto de venta (salvo el origen, al que hay que volver siempre).



Como Amancio nos ha prometido acciones de su empresa cuando consiga lanzarla, vamos a echarle un cable programando un algoritmo que, dados los puntos de venta a los que puede acudir y las distancias entre ellos, le devuelva la máxima ganancia que va a conseguir.

### Entrada

La primera línea contiene un entero  $N$ , que indican el número de puntos de venta (incluido el origen).

Las siguientes  $N$  líneas contienen  $N$  enteros que indican la distancia del punto  $N_i$  al punto  $N_j$  (la distancia de  $N_i$  a  $N_i$  siempre será 0).

### Salida

Por cada caso de prueba se debe imprimir el máximo beneficio que obtendrá el Amancio dado el mapa de puntos de venta.

Ejemplo de entrada	Ejemplo de salida
4 0 10 15 20 10 0 35 25 15 35 0 30 20 25 30 0	95

### Límites

- $4 \leq N \leq 100$
- $10 \leq N_i \leq 1000$