

# Ecologisch vrachtvervoer



## Opgave

Een vrachtwagen wordt in zijn depot geladen met goederen die moeten geleverd worden. Er is van tevoren beslist welk parcours de vrachtwagen moet afleggen, maar niet de richting waarin dat parcours wordt afgelegd: die mag door de chauffeur gekozen worden. Vermits de chauffeur een premie krijgt voor brandstofbesparing, en vermits rijden met een zwaardere vrachtwagen evenredig meer brandstof verbruikt, wil hij zo licht mogelijk rijden, gemiddeld gezien over het hele parcours natuurlijk. Jij moet hem helpen en een programma schrijven dat berekent welke van de twee richtingen de meest zuinige is.

Een voorbeeld: stel dat de chauffeur vertrekt met 100kg lading en volgend schema moet afwerken:

- van depot naar eerste stopplaats: 30km rijden, dan 50kg leveren.
- van eerste naar tweede stopplaats: 20km rijden, dan 30kg leveren.
- van tweede naar derde stopplaats: 10km rijden, dan 20kg leveren.
- van derde stopplaats terug naar depot: 40km rijden.

Het laatste deel van het traject (terug naar het depot) wordt steeds leeg gereden. Er wordt enkel gekeken naar de *extra* kost die de lading meebrengt. Leeg rijden kost dus niets (extra kost is 0).

De chauffeur berekent zijn extra brandstofverbruik op basis van het aantal kg vervoerde goederen over elk traject. Indien hij dit voorbeeldparcours in de gegeven volgorde zou afleggen, is zijn extra verbruik evenredig met  $100 \times 30 + 50 \times 20 + 20 \times 10 + 0 \times 40 = 4200$ . In de omgekeerde richting wordt dit:  $100 \times 40 + 80 \times 10 + 50 \times 20 + 0 \times 30 = 5800$ .

In dit voorbeeld is de originele richting dus de meest zuinige.

## Invoer

De eerste regel van de invoer is een geheel getal  $n$  ( $1 \leq n \leq 1000$ ) dat het aantal opgaven voorstelt. Daarna volgen  $n$  opgaven. Elke opgave bestaat uit 4 opeenvolgende regels:

1. Regel 1 bevat het initiële gewicht  $w$  (in kg) van de lading van de vrachtwagen, uitgedrukt als een geheel getal ( $0 \leq w \leq 1000$ ).

2. Regel 2 bevat een geheel getal  $m$  ( $2 \leq m \leq 100$ ) dat de lengte van het af te leggen parcours aangeeft (het aantal stukken traject die moeten worden afgelegd). Er is dus steeds minstens 1 tussenstop.
3. Regel 3 bevat de afstanden (in km) tussen elke stop in het parcours, uitgedrukt als  $m$  opeenvolgende gehele getallen  $a_i$  ( $1 \leq a_i \leq 100$ ).
4. Regel 4 bevat het aantal kg dat er op elke stop geleverd moet worden. Dit is een rij van  $m - 1$  opeenvolgende gehele getallen  $b_i$  ( $0 \leq b_i \leq 100$ ) met  $\sum b_i = w$ , want het laatste deel wordt steeds leeg afgelegd.

## Uitvoer

Voor elke opgave bereken je welke richting de zuinigste is. Per geval wordt er eerst het totale gewicht  $w$  uitgeschreven. Als de opgegeven richting de zuinigste is print je na een spatie **goed**, anders **omgekeerd**, gevolgd door een newline. Als beide richtingen gelijkwaardig zijn print je **goed**.

## Voorbeeld

### Invoer

```
2
100
4
30 20 10 40
50 30 20
100
4
40 10 20 30
20 30 50
```

### Uitvoer

```
100 goed
100 omgekeerd
```