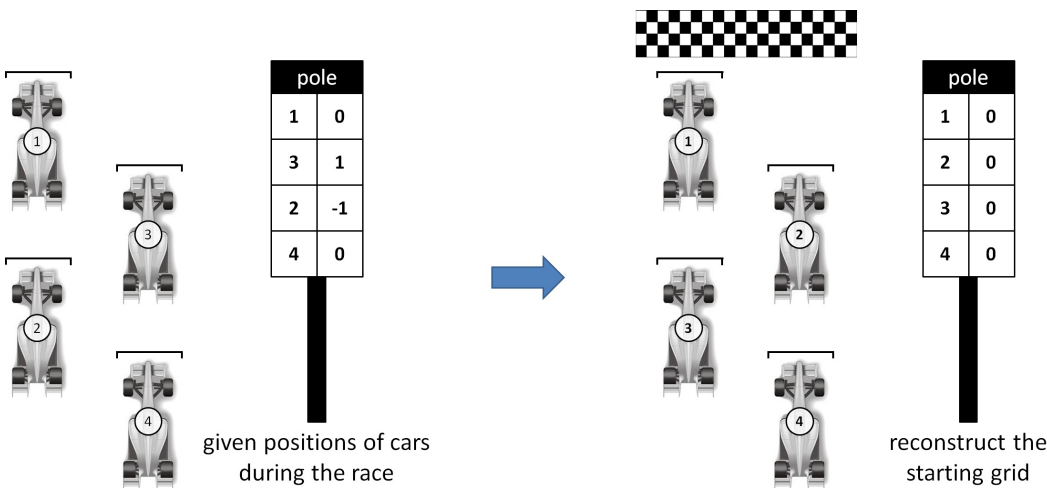


# Pole position

In Formula One races there is always a high pole next the finish line of the track. Before a race starts, the pole is used to display the starting grid (what car starts at what position). The number of the first car in the grid (the *pole position*) is displayed at the top of the pole, the number of the car in the second place is shown below the first one, and so on. During the race, the pole is used to display the current position of each car: the car that is winning the race has its number displayed at the top of the pole, followed by the car that is in second place, and so on.

Besides showing the current position of a car, the pole is also used to display the number of positions the cars have won or lost relative to its position on the starting grid. This is done by showing an integer next to the car number. A positive value  $p$  next to a car number on the pole means that the car has won  $p$  positions relative to its position on the starting grid. A negative value  $-p$  means that the car has lost  $p$  positions. A zero next to a car number on the pole means that the car neither won nor lost any positions (the car is at the same position it started the race).



Given the information displayed on the pole during a Formula One race, you are asked to reconstruct the positions of the cars on the starting grid. While doing this you should take into account that the pole system might not be working properly, causing it to display inconsistent information that makes it impossible to compute valid starting positions for each of the cars. Your program should be able to detect such defects of the pole.

## Assignment

Write a function `startgrid` that takes a list of tuples, where each tuple  $(w, p)$  describes the position of a car during the race.  $w \in \mathbb{Z}$  indicates the car number (each car has been assigned a unique number), and  $p \in \mathbb{Z}$  indicates the number of positions the car has won or lost relative to its position on the starting grid. The order of the tuples in the list determines the current position of the cars during the race. The function should return a list that gives the positions of the cars on the starting grid. This list should contain the car numbers, ordered from the first position to the last position on the starting grid. The function should return an empty list if it is not possible to reconstruct a valid position on the starting grid for any of the cars.

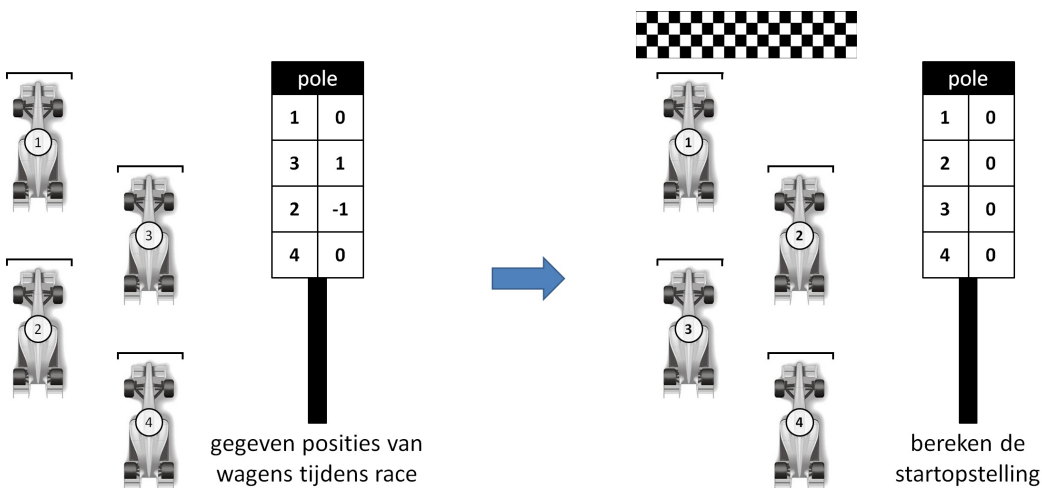
## Example

Click the icons next to the examples to see a graphical representation of the position of the cars during the race and the information displayed on the pole at that moment. Use this information to reconstruct the position of the cars on the starting grid.

```
>>> startgrid([(1, 0), (3, 1), (2, -1), (4, 0)]) 🏁
[1, 2, 3, 4]
>>> startgrid([(2, 2), (8, 0), (5, -2), (7, 1), (1, 1), (9, 1), (3, -3)]) 🏁
[5, 8, 2, 3, 7, 1, 9]
>>> startgrid([(22, 1), (9, 1), (13, 0), (21, -2)]) 🏁
[]
>>> startgrid([(19, 1), (9, -99), (17, 0)]) 🏁
[]
```

Bij formule-1 wedstrijden staat er altijd een hoge paal (in het Engels *pole*) naast de eindstreep. Voor de start van de wedstrijd toont deze paal de startposities (welke wagen vertrekt in welke positie). Het nummer van de eerste wagen staat helemaal bovenaan (op de *pole position*), het nummer van de tweede wagen staat eronder, enzoverder. Tijdens de wedstrijd wordt de paal gebruikt om de huidige positie van elke wagen te tonen: (het nummer van) de leidende wagen staat dan helemaal bovenaan, gevolgd door de wagen op de tweede plaats, enzoverder.

Naast de huidige positie van elke wagen, toont de paal ook het aantal posities dat elke wagen gewonnen of verloren heeft, relatief ten opzichte van zijn startpositie. Dit wordt voorgesteld door een geheel getal  $\$p\$$ , waarbij een negatieve waarde  $\$-p\$$  betekent dat de wagen  $\$p\$$  posities heeft verloren sinds de start van de wedstrijd, en een positieve waarde  $\$p\$$  betekent dat de wagen  $\$p\$$  posities gewonnen heeft. Indien  $\$p=0\$$  dan betekent dit dat de wagen in dezelfde positie rijdt als aan de start van de wedstrijd.



Gevraagd wordt om, gegeven de informatie van de paal tijdens een formule-1 wedstrijd, de startposities van alle wagens te berekenen. Opgelet: de paal kan soms defect zijn, waardoor er foute informatie getoond wordt en het niet mogelijk is om een geldige startpositie voor alle wagens te berekenen. Je programma moet defecten aan de paal kunnen detecteren.


### Opgave


Schrijf een functie `startposities` waaraan een lijst van tuples wordt doorgegeven. Elke tuple  $\$(w, p)\$$  in de lijst beschrijft de positie van een wagen in de wedstrijd.  $\$w\$$  stelt het nummer van de wagen voor (alle wagens in een wedstrijd hebben een verschillend nummer), en  $\$p\$$  stelt het aantal posities voor dat deze wagen gewonnen of verloren heeft ten opzichte van de start van de


wedstrijd. De wagens worden opgelijst in volgorde van hun positie tijdens de wedstrijd. De functie moet de gereconstrueerde startposities terug, als een lijst van de nummers van de wagens, van eerste naar laatste positie. Indien het niet mogelijk is om voor een of meerdere wagens een geldige startpositie te reconstrueren, moet de functie een lege lijst teruggeven.


## Voorbeeld

Klik op de icoontjes naast de voorbeelden om een grafische voorstelling te zien van de positie van de wagens tijdens de wedstrijd en de stand van de pole op dat ogenblik, en de daarmee corresponderende positie van de wagens bij aanvang van de wedstrijd.

```
>>> startposities([(1, 0), (3, 1), (2, -1), (4, 0)])   
[1, 2, 3, 4]
```

```
>>> startposities([(2, 2), (8, 0), (5, -2), (7, 1), (1, 1), (9, 1), (3, -3)])   
[5, 8, 2, 3, 7, 1, 9]
```

```
>>> startposities([(22, 1), (9, 1), (13, 0), (21, -2)])   
[]
```

```
>>> startposities([(19, 1), (9, -99), (17, 0)])   
[]
```