

# Pub game

A well-known pub game goes like this: take a hexagonal beverage coaster and put the numbers between 1 and 12 on each of the twelve corners and midpoints of the six sides, in such a way that the sum of the three numbers on each side is the same.

The image below gives a possible solution with sum equal to 19. There are many possible solutions, that don't necessarily result in the same sum. The example solution can be represented as the list of integers 1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10. Such a list representation always starts with a number that is put at a corner point, and reads the numbers at all the corners and midpoints in succession. It does not matter what corner point is taken as the starting point, nor whether we read clockwise or counterclockwise.

The list of integers 1, 7, 12, 3, 5, 9, 6, 4, 10, 2, 8, 11 represents another possible solution with sum equal to 20.



In addition, the game does not necessarily need to be played with hexagonal beverage coasters. Each beverage coaster with  $n$  ( $n \geq 3$ ) sides can be used. The general description of the game then becomes: put all numbers between 1 and  $2n$  on each of the  $2n$  corners and midpoints of the  $n$  sides, such that the sum of the three numbers on each side is the same. The list of integers 4, 3, 5, 1, 6, 2 represents a solution for triangular beverage coasters, the list of integers 5, 2, 6, 3, 4, 8, 1, 7 is a solution for square beverage coasters and the list of integers 7, 8, 1, 10, 5, 2, 9, 4, 3, 6 is a possible solution for pentagonal beverage coasters.

## Assignment

Write a function `valid_solution` that can be used to determine whether or not a given list of integers represents a valid solution for the pub game described above. The function takes a list of integers as its first argument. The function can take an optional second integer argument  $s$ . The given list of numbers is a valid solution if all of the following conditions are met:

- the number of elements  $n$  in the given list of integers is even and greater than or equal to six
- each integer between 1 and  $n$  must occur exactly once in the list
- the sum of any three numbers that are on the same side is equal to the other sums
- if a value  $s$  is passed to the optional parameter, the sum of the three numbers on the same side must be equal to  $s$

The function must return the value `True` if the integers in the given list meet all the above conditions. Otherwise the value `False` must be returned.

## Example

```
>>> valid_solution([1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10])
True
>>> valid_solution([1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10], s=19)
True
>>> valid_solution([1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10], s=20)
False
>>> valid_solution([1, 7, 12, 3, 5, 9, 6, 4, 10, 2, 8, 11], s=20)
True
>>> valid_solution([4, 3, 5, 1, 6, 2])
True
>>> valid_solution([5, 2, 6, 3, 4, 8, 1, 7], 13)
True
>>> valid_solution([7, 8, 1, 10, 5, 2, 9, 4, 3, 6], 16)
True
```

Een bekend caféspelletje gaat als volgt: neem een zeshoekig bierkaartje, en plaats de getallen tussen 1 en 12 op de twaalf hoek- en middelpunten van de zes zijden, zodat de som van de drie getallen op elke zijde dezelfde is.

Onderstaande figuur geeft een mogelijke oplossing met som gelijk aan 19. Er zijn namelijk heel wat verschillende oplossingen mogelijk, die niet steeds dezelfde som opleveren. De voorbeeldoplossing kan voorgesteld worden door de reeks getallen 1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10. Hierbij moet het eerste getal altijd op een hoekpunt geplaatst worden en moeten de getallen in volgorde op de volgende hoek- en middelpunten uitgeschreven worden. Welk hoekpunt als startpunt wordt genomen, en of de getallen in wijzerzin dan wel tegenwijzerzin worden uitgeschreven maakt geen verschil uit. De reeks getallen 1, 7, 12, 3, 5, 9, 6, 4, 10, 2, 8, 11 stelt dan een andere oplossing voor met som gelijk aan 20.



Bovendien hoeft het spelletje niet noodzakelijk met zeshoekige bierkaartjes gespeeld te worden. Elk bierkaartje met  $n$  ( $n \geq 3$ ) zijden kan gebruikt worden. De algemene omschrijving van het spelletje wordt dan: plaats de getallen tussen 1 en  $2n$  op de  $2n$  hoek- en middelpunten van de  $n$  zijden, zodat de som van de drie getallen op elke zijde dezelfde is. De reeks getallen 4, 3, 5, 1, 6, 2 stelt dan een oplossing voor met driehoekige bierkaartjes, de reeks getallen 5, 2, 6, 3, 4, 8, 1, 7 is een oplossing voor vierkante bierkaartjes en 7, 8, 1, 10, 5, 2, 9, 4, 3, 6 stelt een mogelijke oplossing met vijfhoekige bierkaartjes voor.

## Opgave

Schrijf een functie `geldige_oplossing` waarmee kan bepaald worden of een gegeven reeks getallen een geldige oplossing voorstelt van het hierboven beschreven caféspeletje. Aan deze functie moet verplicht een lijst van natuurlijke getallen als argument doorgegeven worden. Optioneel kan als tweede argument ook nog een bijkomend natuurlijk getal  $s$  doorgegeven worden. De opgegeven lijst getallen vormt een geldige oplossing wanneer aan al de volgende voorwaarden voldaan is:

- het aantal getallen  $n$  in de opgegeven lijst moet even zijn en groter of gelijk aan zes
- elk van de getallen tussen 1 en  $n$  moet juist één keer in de lijst voorkomen
- de som van elke drie getallen op eenzelfde zijde moet gelijk zijn
- indien een waarde  $s$  werd doorgegeven voor de optionele parameter, dan moet de som van de drie getallen op elke zijde gelijk zijn aan  $s$

De functie moet de waarde `True` teruggeven indien de getallen uit de opgegeven lijst voldoen aan alle voorwaarden. Anders moet de waarde `False` teruggegeven worden.

## Voorbeeld

```
>>> geldige_oplossing([1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10])
True
>>> geldige_oplossing([1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10], s=19)
True
```

```
>>> geldige_oplossing([1, 6, 12, 2, 5, 11, 3, 9, 7, 4, 8, 10], s=20)
False
>>> geldige_oplossing([1, 7, 12, 3, 5, 9, 6, 4, 10, 2, 8, 11], s=20)
True
>>> geldige_oplossing([4, 3, 5, 1, 6, 2])
True
>>> geldige_oplossing([5, 2, 6, 3, 4, 8, 1, 7], 13)
True
>>> geldige_oplossing([7, 8, 1, 10, 5, 2, 9, 4, 3, 6], 16)
True
```