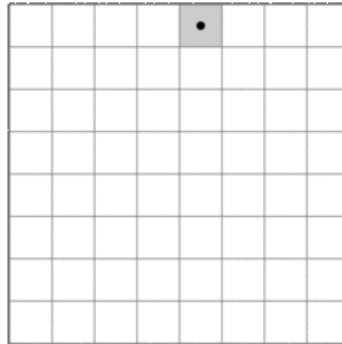


Knights tour

A knight moves in an unusual way on a chessboard. Such an L-shaped movement on a chess board is called a **horse jump**. When it moves, the knight can move to a square that is two squares horizontally and one square vertically, or two squares vertically and one square horizontally. The complete move therefore looks like the letter *L*. A **knight's tour** is a sequence of moves of a knight on a chessboard such that the knight visits every square exactly once. If the knight ends on a square that is one knight's move from the beginning square (so that it could tour the board again immediately, following the same path), the tour is *closed*, otherwise it is *open*. The exact number of open tours on an 8×8 chessboard is still unknown.



Analogous to a knight's tour, other chess pieces might also make a tour on a chessboard. For example, a tower's tour is a series of displacements in which the tour visits all squares exactly ones.

Assignment

First, write three functions `knight`, `rook` and `queen` that each take two integer arguments representing the row number and the column number of the position of an eponymous chess piece on a 8×8 chessboard. Rows and column indexing starts at 0. Each functions must return the list of positions on the chessboard that can be reached by the chess piece in one single move. Each position is represented as a tuple containing a row number and a column number on the chessboard. The list returned by the function must contain the positions in sorted order according increasing row number and then according to increasing column number.

	0	1	2	3	4	5	6	7	
0	1	4	53	18	55	6	43	20	0
1	52	17	2	5	38	19	56	7	1
2	3	64	15	54	31	42	21	44	2
3	16	51	28	39	34	37	8	57	3
4	63	14	35	32	41	30	45	22	4
5	50	27	40	29	36	33	58	9	5
6	13	62	25	48	11	60	23	46	6
7	26	49	12	61	24	47	10	59	7
	0	1	2	3	4	5	6	7	

The sequence of squares visited by a chess piece (i.e. a tour) is represented by labelling the squares of an 8×8 chessboard with the integers 1 to 64. The number of a square indicates its index in the series of movements on the board. Write a function `tour` that returns a Boolean value, indicating whether or not a given series of moves represents an open or a closed tour. The series of moves on a board is passed to the function as an 8×8 grid, represented as a list of lists. The inner lists represent the rows of the chessboard, having integer elements that represent the labels of the squares on each row. The function also has two optional parameters:

- a parameter `piece` that takes a function that computes for a given position on a chessboard which squares can be reached in a single move of a particular chess piece (the functions `knight`, `rook` and `queen` are examples of such functions); if no argument is passed to this parameter, the function `tour` should apply horse jumps
- a parameter `closed` that takes a Boolean value, indicating whether the tour should be closed (value `True`) or if it can also be open (value `False`; the default value)

Chess rules

For those who are not familiar with the rules of chess: a rook may move an arbitrary number of squares horizontally or vertically. A queen may move an arbitrary number of squares horizontally, vertically or diagonally.

Example

```

>>> knight(4, 4)
[(2, 3), (2, 5), (3, 2), (3, 6), (5, 2), (5, 6), (6, 3), (6, 5)]
>>> knight(0, 3)
[(1, 1), (1, 5), (2, 2), (2, 4)]

>>> rook(4, 4)
[(0, 4), (1, 4), (2, 4), (3, 4), (4, 0), (4, 1), (4, 2), (4, 3), (4, 5), (4, 6), (4, 7), (5, 4), (6, 4), (7, 4)]

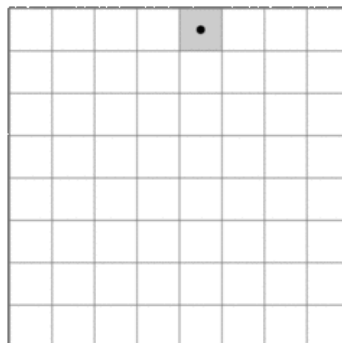
>>> queen(2, 3)
[(0, 1), (0, 3), (0, 5), (1, 2), (1, 3), (1, 4), (2, 0), (2, 1), (2, 2), (2, 4), (2, 5), (2, 6), (2, 7), (3, 2), (3, 3), (3, 4), (4, 1), (4, 3), (4, 5), (5, 0), (5, 3), (5, 6), (6, 3), (6, 7), (7, 3)]

>>> chessboard = [
... [ 3, 22, 49, 56, 5, 20, 47, 58],
... [50, 55, 4, 21, 48, 57, 6, 19],
... [23, 2, 53, 44, 25, 8, 59, 46],
... [54, 51, 24, 1, 60, 45, 18, 7],
... [15, 36, 43, 52, 17, 26, 9, 62],
... [42, 39, 16, 33, 12, 61, 30, 27],
... [35, 14, 37, 40, 29, 32, 63, 10],
... [38, 41, 34, 13, 64, 11, 28, 31]
... ]
>>> tour(chessboard)
True
>>> tour(chessboard, piece=knight, closed=True)
False
>>> tour(chessboard, piece=rook)
False
>>> tour(chessboard, piece=queen)
False

>>> chessboard = [
... [ 1, 2, 3, 4, 5, 6, 7, 8],
... [28, 29, 30, 31, 32, 33, 34, 9],
... [27, 48, 49, 50, 51, 52, 35, 10],
... [26, 47, 60, 61, 62, 53, 36, 11],
... [25, 46, 59, 64, 63, 54, 37, 12],
... [24, 45, 58, 57, 56, 55, 38, 13],
... [23, 44, 43, 42, 41, 40, 39, 14],
... [22, 21, 20, 19, 18, 17, 16, 15]
... ]
>>> tour(chessboard, piece=rook)
True
>>> tour(chessboard, piece=queen)
True
>>> tour(chessboard, piece=queen, closed=True)
False

```

Het paard beweegt op een merkwaardige wijze over het schaakbord. Deze L-vormige manier van bewegen wordt een **paardensprong** genoemd. Om een paardensprong uit te voeren, gaat het paard eerst twee vakjes naar voren of naar achteren en daarna een vakje naar links of naar rechts. Het paard mag ook eerst twee vakjes naar links of naar rechts stappen, en dan een vakje naar voren of naar achteren. Een reeks sprongen van een paard op een schaakbord wordt een **paardenronde** genoemd als het paard daarbij elk vakje op het schaakbord juist éénmaal aandoet. Als het paard eindigt op een vakje dat één sprong verwijderd is van het vakje van waar de reeks begonnen is, dan zegt men dat de paardenronde **gesloten** is. In het andere geval wordt het een **open** paardenronde genoemd.



Analoog aan een paardenronde, kunnen ook andere schaakstukken gebruikt worden om een ronde uit te voeren over alle vakjes van het schaakbord. Zo is een **torenronde** bijvoorbeeld een reeks zetten van een toren op een schaakbord, waarbij elk vakje juist éénmaal bezocht wordt.

Opgave

Schrijf eerst en vooral drie functies `paard`, `toren` en `koningin` waaraan telkens het rij- en kolomnummer van de positie van een gelijknamig schaakstuk op een 8×8 schaakbord moeten doorgegeven worden. Rijen en kolommen worden hierbij steeds genummerd

vanaf nul. Elk van deze functies moet de lijst van posities op het schaakbord teruggeven die het schaakstuk in één zet kan bereiken. Elke positie wordt voorgesteld als een tuple met een rij- en kolomnummer op het schaakbord. De posities moeten in volgorde opgelijst worden volgens stijgend rijnummer en vervolgens volgens stijgend kolomnummer.

	0	1	2	3	4	5	6	7	
0	1	4	53	18	55	6	43	20	0
1	52	17	2	5	38	19	56	7	1
2	3	64	15	54	31	42	21	44	2
3	16	51	28	39	34	37	8	57	3
4	63	14	35	32	41	30	45	22	4
5	50	27	40	29	36	33	58	9	5
6	13	62	25	48	11	60	23	46	6
7	26	49	12	61	24	47	10	59	7

We stellen een ronde voor door de vierkanten van een 8×8 schaakbord te vullen met de getallen van 1 tot en met 64. Het getal op elk vierkant duidt het volgnummer aan van het vakje in een reeks van zetten op het bord. Schrijf een functie `ronde` die een Booleaanse waarde teruggeeft die aangeeft of een gegeven reeks zetten al dan niet een open of gesloten ronde vormt. De reeks zetten op het bord wordt aan de functie doorgegeven als een 8×8 rooster, onder de vorm van een lijst van lijsten. Hierbij stellen de binnenste lijsten de rijen van het schaakbord voor, waarvan de elementen de getallen voorstellen waarmee de vierkanten op de rij genummerd zijn. De functie heeft ook nog twee optionele parameters:

- een parameter `schaakstuk` waaraan een functie kan doorgegeven worden die voor een gegeven positie op het schaakbord aangeeft welke vierkanten in één zet kunnen bereikt worden door een schaakstuk (de functies `paard`, `toren` en `koningin` zijn voorbeelden van dergelijke functies); als er geen argument wordt doorgegeven voor deze parameter, dan moet de functie `ronde` werken met paardensprongen
- een parameter `gesloten` waaraan een Booleaanse waarde kan doorgegeven worden; hiermee wordt aangegeven of de ronde gesloten moet zijn (waarde `True`) of dat die eventueel open mag zijn (waarde `False`, de standaardwaarde)

Schaakregels

Voor zij die niet zo vertrouwd zijn met de regels van het schaakspel geven we nog mee dat een toren bij elke zet mag bewegen over een willekeurig aantal velden in horizontale of verticale richting. Een koningin mag bij elke zet bewegen over een willekeurig aantal velden in horizontale, verticale of diagonale richting.

Voorbeeld

```
>>> paard(4, 4)
[(2, 3), (2, 5), (3, 2), (3, 6), (5, 2), (5, 6), (6, 3), (6, 5)]
>>> paard(0, 3)
[(1, 1), (1, 5), (2, 2), (2, 4)]

>>> toren(4, 4)
[(0, 4), (1, 4), (2, 4), (3, 4), (4, 0), (4, 1), (4, 2), (4, 3), (4, 5), (4, 6), (4, 7), (5, 4), (6, 4), (7, 4)]

>>> koningin(2, 3)
[(0, 1), (0, 3), (0, 5), (1, 2), (1, 3), (1, 4), (2, 0), (2, 1), (2, 2), (2, 4), (2, 5), (2, 6), (2, 7), (3, 2), (3, 3), (3, 4), (4, 1), (4, 3), (4, 5), (5, 0), (5, 3), (5, 6), (6, 3), (6, 7), (7, 3)]

>>> schaakbord = [
... [ 3, 22, 49, 56, 5, 20, 47, 58],
... [50, 55, 4, 21, 48, 57, 6, 19],
... [23, 2, 53, 44, 25, 8, 59, 46],
... [54, 51, 24, 1, 60, 45, 18, 7],
... [15, 36, 43, 52, 17, 26, 9, 62],
... [42, 39, 16, 33, 12, 61, 30, 27],
... [35, 14, 37, 40, 29, 32, 63, 10],
... [38, 41, 34, 13, 64, 11, 28, 31]
... ]
>>> ronde(schaakbord)
True
>>> ronde(schaakbord, schaakstuk=paard, gesloten=True)
False
>>> ronde(schaakbord, schaakstuk=toren)
False
>>> ronde(schaakbord, schaakstuk=koningin)
False

>>> schaakbord = [
... [ 1, 2, 3, 4, 5, 6, 7, 8],
... [28, 29, 30, 31, 32, 33, 34, 9],
```

```
... [27, 48, 49, 50, 51, 52, 35, 10],
... [26, 47, 60, 61, 62, 53, 36, 11],
... [25, 46, 59, 64, 63, 54, 37, 12],
... [24, 45, 58, 57, 56, 55, 38, 13],
... [23, 44, 43, 42, 41, 40, 39, 14],
... [22, 21, 20, 19, 18, 17, 16, 15]
... ]
>>> ronde(schaakbord, schaakstuk=toren)
True
>>> ronde(schaakbord, schaakstuk=koningin)
True
>>> ronde(schaakbord, schaakstuk=koningin, gesloten=True)
False
```