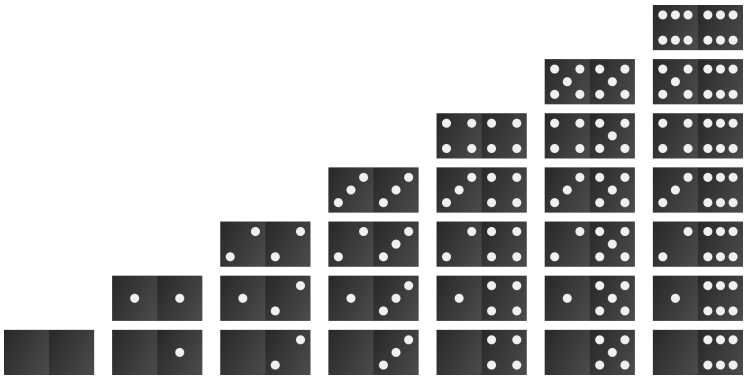
Derek and the dominoes

Derek loves dominoes. He has his own default domino set which consists of 28 square stones. Every stone is divided in two squares. Each square is marked with zero to six dots.



A complete domino set consisting of 28 stones.

In order to play the game, at least two players are needed. Layla — Derek's sister — refuses to play, because she always loses when she plays against her brother. Out of boredom, Derek thought of two games he can play by himself.

Assignment

Every game starts the same way. Derek randomly chooses a number of domino stones and lays them in a sequence one after another. This sequence doesn't necessarily contain all stones of his domino set. In this assignment, we represent a domino stone as a tuple of two integers, that indicate the number of dots that are on both halves of the stone. A sequence of stones is represented as a list or a tuple of domino stones. In a sequence of domino stones, two stones follow each other if the right side of the left stone is equal to the amount of dots on the left side of the right stone.

- Write a function `beginsequence` to which a sequence of stones must be given. The function must print the maximum amount of stones that can follow up after each other at the beginning of the sequence. The function has an optional parameter `turn` to which a Boolean value can be given (default value: `False`). This parameter indicates whether the stone may be turned 180° in order to make it follow the stone before. The first stone of a sequence may never be turned.
- Write a function `dominosequence` to which a sequence of domino stones must be given. The function has an optional parameter `turn` to which a Boolean value can be given (default value: `False`). This parameter indicates whether the stone may be turned 180° in order to make it follow the stone before. The function must print a new sequence of domino stones that is built as follows:
    1. take the first stone of the given sequence and use it as the first stone of a new sequence; this first stone may never be turned
    2. run through the remaining stones of a given sequence from left to right and search the first stone that can be put next to the last stone of a new sequence (possibly after turning it 180° if this is permitted by the parameter `turn`)
    3. depending on whether a stone was found in step 2, there are two possibilities:
        a. if a stone was found, delete it from the given sequence of stones and expand

the new sequence with the new stone on the right-hand side, so that is follows the last stone; after this procedure, repeat it from step 2

b. if no stone was found, the procedure ends and the newly formed sequence must be printed

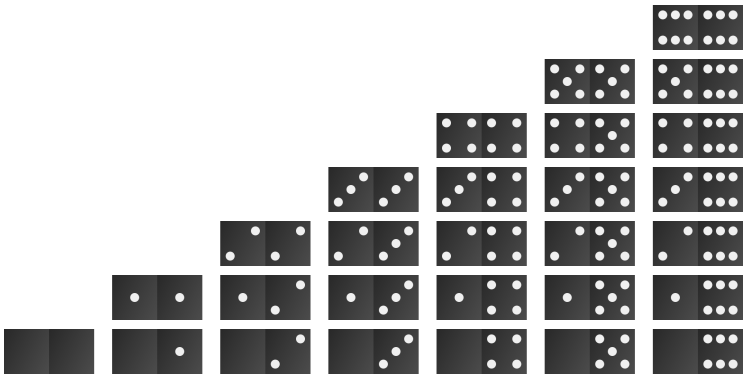
The functions above don't depend on each other. It isn't necessary to call on the function beginsequence when implementing the function dominosequence.

Example

```
>>> sequence = ((3, 0), (0, 1), (1, 2), (0, 2), (0, 0), (3, 3), (1, 1), (2, 3), (3, 1), (2, 2))
>>> beginsequence(reeks)
3
>>> beginsequence (sequence, turn=True)
5

>>> sequence = ((1, 3), (2, 0), (1, 0), (0, 3), (1, 1), (0, 0), (3, 3), (2, 2), (3, 2), (1, 2))
>>> dominosequence(sequence)
[(1, 3), (3, 3), (3, 2), (2, 0), (0, 3)]
>>> dominosequence(sequence, turn=True)
[(1, 3), (3, 0), (0, 2), (2, 2), (2, 3), (3, 3)]
```

Derek is verzot op domino. Hij beschikt dan ook over zijn eigen standaard dominoset, bestaande uit 28 rechthoekige stenen. Elke steen wordt in twee vierkanten verdeeld, waarbij elke helft gemarkeerd is met nul tot zes ogen.



Een volledig dominoset bestaande uit 28 stenen.

Om het dominospel te kunnen spelen zijn er minstens twee spelers nodig. Layla — de zus van Derek — vertikt het echter om nog mee te spelen, omdat ze altijd het onderspit moet delven als ze het tegen haar broer opneemt. Uit verveling heeft Derek dan maar twee spelletje bedacht die hij alleen kan spelen.

Opgave

Elk dominospelletje begint op dezelfde manier. Derek kiest willekeurig een aantal dominostenen en legt die in een reeks achter elkaar. Deze reeks hoeft dus niet noodzakelijk alle stenen van zijn dominoset te bevatten. In deze opgave stellen we een dominosteen voor als een tuple van twee integers, die het aantal ogen aangeven waarmee beide helften van de steen gemarkeerd zijn. Een reeks dominostenen wordt voorgesteld als een lijst of een tuple van dominostenen. In een reeks dominostenen volgen twee stenen elkaar op als het aantal ogen aan de rechterkant van de linker steen gelijk is aan het aantal ogen aan de linkerkant van de rechter steen.

- Schrijf een functie beginreeks waaraan een reeks dominostenen moet doorgegeven worden. De functie moet teruggeven wat het maximum aantal dominostenen is die op elkaar volgen aan het begin van de reeks. De functie heeft een optionele parameter draaien waaraan een Booleaanse waarde kan doorgegeven worden (standaardwaarde: False). Deze parameter geeft aan of een dominosteen 180° mag gedraaid worden om hem te laten volgen op de vorige steen. De eerste steen van de reeks mag nooit gedraaid worden.
- Schrijf een functie dominoreeks waaraan een reeks dominostenen moet doorgegeven worden. De functie heeft een optionele parameter draaien waaraan een Booleaanse waarde kan doorgegeven worden (standaardwaarde: False). Deze parameter geeft aan of een dominosteen 180° mag gedraaid worden om hem te laten volgen op de vorige steen. De functie moet een nieuwe reeks dominostenen teruggeven die op de volgende manier opgebouwd wordt:
    1. neem de eerste steen van de gegeven reeks en gebruik deze als eerste steen van de nieuwe reeks; deze eerste steen mag nooit gedraaid worden
    2. doorloop de resterende stenen van de gegeven reeks van links naar rechts en zoek de eerste steen die aan de rechterkant kan aangelegd worden van de laatste steen van de nieuwe reeks (eventeel na 180° draaien indien dit toegelaten is door de parameter draaien)
    3. afhankelijk van het feit of er in stap 2 een steen gevonden werd, zijn er nu twee mogelijkheden:
        a. indien er een steen gevonden werd, verwijder die dan uit de gegeven reeks dominostenen en breid de nieuwe reeks aan de rechterkant uit met de nieuwe steen, zodat die volgt op de vorige steen; herhaal daarna de procedure vanaf stap 2
        b. indien er geen steen gevonden werd, eindigt de procedure en moet de nieuw gevormde reeks teruggegeven worden

Bovenstaande functies zijn onafhankelijk van elkaar. Het is dus niet nodig om de functie beginreeks aan te roepen bij de implementatie van de functie dominoreeks.

Voorbeeld

```
>>> reeks = ((3, 0), (0, 1), (1, 2), (0, 2), (0, 0), (3, 3), (1, 1), (2, 3), (3, 1), (2, 2))
>>> beginreeks(reeks)
3
>>> beginreeks(reeks, draaien=True)
5

>>> reeks = ((1, 3), (2, 0), (1, 0), (0, 3), (1, 1), (0, 0), (3, 3), (2, 2), (3, 2), (1, 2))
>>> dominoreeks(reeks)
[(1, 3), (3, 3), (3, 2), (2, 0), (0, 3)]
>>> dominoreeks(reeks, draaien=True)
[(1, 3), (3, 0), (0, 2), (2, 2), (2, 3), (3, 3)]
```