

Uitwerking opdracht

Opdracht : *Fen notatie*
Weeknummer : *5*
Studentnummer : *s1060679*
Naam student : *Dion van den Berg*
Specialisatie : *MEDT*
Pogingnummer : *1*

1. Vraagstelling

Zet een string of grid om in de fen notatie

2. Specificatie

Invoer

De gebruiker voert een string in bij de fen2grid functie of een string met enters in bij de grid2fen functie.

Uitvoer

De gebruiker krijgt een grid te zien of een fen notatie string

Verband tussen in- en uitvoer

De uitvoer wordt gemaakt door de ingevoerde string.

Beperkingen

Te lange strings worden afgekapt

Voorbeelden (testscenario's)

Test 1

Invoer:

```
print(fen2grid('rnbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR'))
```

Uitvoer:

```
rnbqkbnr
pppppppp
*****
*****
*****
*****
PPPPPPPP
RNBQKBNR
```

Test 2

Invoer:

```
rooster = '''rnbqkbnr
... pppppppp
... *****
... *****
... *****
... *****
... *****
... Pppppppp
... RNBQKBNR'''
>>> print(grid2fen(rooster))
```

Uitvoer:

```
rnbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR
```

3. Ontwerp

Hoe maak ik het rooster ?

Door het te printen met een \n erin na 8 karakters

Hoe ga ik alles omzetten?

Door het te splitsen op elke enter

Hoe laat ik alles uiteindelijk zien ?

in regels netjes onderverdeeld onder elkaar met \n als enters

4. Pseudocode

Functie zet om naar grid (string, tekenje)

Splits de sting op de / → lijst

Loop door de lijst

Loop per lijstitem door de characters.

*als er een nummer staat zet het tekenje neer * het nummer*

anders print de letter

zet aan het einde vande 8 karakters een \n voor enter

functie grid2fen (stiring, tekenje)

maak er een lijst van en vul op per \n

tel het aantal tekenjes op en schrijf het cijfer in de string

zet de letters in de string

per 8 letters zet een / neer

5. Code

```
def fen2grid(fen: str, emptysign='*') -> str:
    # je krijgt een string binnen die een 8x8 grid voor moet stellen
    string = ''
    rows = fen.split('/')
    expectedRows = 0
    # loop door elke letter heen
    for x in rows:
        for y in x:
            if y.isdigit():
                # is het een nummer? zet er het tekenje neer van emptysign
                # keer het aantal dat er staat.
                string += (emptysign * int(y))

            else:
                string += y
                expectedRows += 1
                # zodra je de 8 hebt bereikt zet een enter neer.
                if expectedRows != len(rows):
                    string += '\n'
    return string

def grid2fen(grid: str, emptysign='*') -> str:
    # je krijgt een grid binnen met enters erin.
    string = ''
    teller = 0
    # splits het grid op bij elke enter
    rows = grid.split('\n')
    expectedRows = 0
    # loop door de rijen heen en zet het netjes achter elkaar.
    for x in rows:
        for y in x:
            # als er een emptysign tekenje staat zet een teller omhoog.
            if y == emptysign:
                teller += 1

            if not y == emptysign:
                if teller != 0:
                    # zet het nummer van de teller neer in de string
                    string += str(teller)
                    string += y
                    teller = 0

            # als na de loop de teller geen 0 is. zet hem achter de string en
            # reset hem.
            if teller != 0:
                string += str(teller)
                teller = 0
                expectedRows += 1
            if expectedRows != len(rows):
                string += '/'
    return string
```

```

# laat het voorbeeld zien.
print(fen2grid('rnbqkbnr/pppppppp/8/8/8/8/PPPPPPPP/RNBQKBNR') + '\n')
print(fen2grid('rnbqkbnr/pppppppp/8/8/4P3/8/PPPP1PPP/RNBQKBNR', '.') +
'\n')
print(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPPP1PPP/RNBQKBNR', '+') +
'\n')

rooster = '''rnbqkbnr
pppppppp
*****
*****
*****
*****
PPPPPPPP
RNBQKBNR'''
print(grid2fen(rooster))

print(grid2fen(fen2grid('rnbqkbnr/pppppppp/8/8/4P3/8/PPPP1PPP/RNBQKBNR')))
print(grid2fen(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPPP1PPP/RNBQKBNR',
'.'), '.'))
print(grid2fen(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/5N2/PPPP1PPP/RNBQKB1R',
'+'), '+'))

```

6. Test

Test 1

```

print(fen2grid('rnbqkbnr/pppppppp/8/8/8/8/PPPPPPPP/RNBQKBNR'))
rnbqkbnr
pppppppp
*****
*****
*****
*****
PPPPPPPP
RNBQKBNR

```

Test 2

```

>>> rooster = '''rnbqkbnr
... pppppppp
... *****
... *****
... *****
... *****
... Pppppppp
... RNBQKBNR'''
>>> print(grid2fen(rooster))
rnbqkbnr/pppppppp/8/8/8/8/PPPPPPPP/RNBQKBNR

```