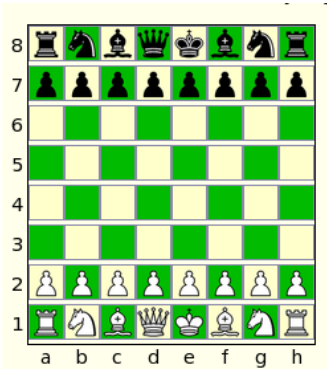


# Forsyth-Edwards notation

The *Forsyth-Edwards notation* (FEN) is a standard notation with which a certain game situation of a chess game can be described. FEN contains all information that is needed to continue a chess game from a certain game situation. The system was developed by David Forsyth — journalist with a Scottish paper — who already popularized it in the 19th century. Steven J. Edwards later expanded the FEN so that it could also be used with computers.



Positioning at the start of a chess game from the white player's point of view. On the bottom row, from left to right, we have: tower (R), knight (N), bishop (B), queen (D) and king (K). On the second row we find the pawns.

In the first part of FEN (we leave out a few parts of the game situation, for example who's turn it is) the placement of the pieces on the chess board is described from the white player's point of view. The rows are described from top to bottom, and the filling of the fields in each row is described from left to right. According to the [Standard Algebraic notation](#) (SAN) every chess piece is described with one letter: K = king, Q = queen, R = rook, B = bishop, N = knight, P = pawn. White pieces are indicated with uppercase letters (KQRBNP) and black pieces with a lowercase letter (kqrbnp). Empty fields are indicated with numbers 1 to 8 that indicated the number of consecutive empty fields. A slash (/) is used to separate the rows.

## Assignment

- Write a function `fen2grid` which can be used to convert the positioning of a chess board in FEN to a string representation of the format of a  $8 \times 8$  grid. In this last description, the rows are separated by newlines (`\n`) and every empty field is indicated with one certain ASCII character. The pieces are indicated with the same letters as with FEN. To this function, a string with the FEN description should be given as a first argument. As a second, optional, argument, one can also give the ASCII character that is used to indicate the empty fields with (standard, an asterisk (\*) is used for this). The function must print the string with the grid representation as a result.
- Write a function `grid2fen` that does the exact opposite of `fen2grid`, and thus converts a description in the format of a grid to a description in FEN. Give the function a string with grid description as a first argument. As a second, optional argument, a character indicating the way empty fields are indicated in the grid description. The asterisk (\*) is again used as a standard value. The function must print the string with FEN description.

## Example

```

>>> print(fen2grid('rnbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR'))
rnbqkbnr
pppppppp
*****
*****
*****
*****
PPPPPPPP
RNBQKBNR
>>> print(fen2grid('rnbqkbnr/pppppppp/8/8/4P3/8/PPP1PPP/RNBQKBNR', '.'))
rnbqkbnr
pppppppp
.....
.....
....P...
.....
PPP.PPP
RNBQKBNR
>>> print(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPP1PPP/RNBQKBNR', '+'))
rnbqkbnr
pp+ppppp
+++++++
++p++++
++++P+++
+++++++
PPP+PPP
RNBQKBNR

>>> grid = "rnbqkbnr
... pppppppp
... *****
... *****
... *****
... *****
... *****
... P P P P P P P P
... R N B Q K B N R"
>>> print(grid2fen(grid))
rnbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR

>>> print(grid2fen(fen2grid('rnbqkbnr/pppppppp/8/8/4P3/8/PPP1PPP/RNBQKBNR')))
rnbqkbnr/pppppppp/8/8/4P3/8/PPP1PPP/RNBQKBNR
>>> print(grid2fen(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPP1PPP/RNBQKBNR', '.'), '.'))
rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPP1PPP/RNBQKBNR
>>> print(grid2fen(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/5N2/PPP1PPP/RNBQKB1R', '+'), '+'))
rnbqkbnr/pp1ppppp/8/2p5/4P3/5N2/PPP1PPP/RNBQKB1R

```

De *Forsyth-Edwards notatie* (FEN) is een standaardnotatie waarmee een bepaalde spelsituatie van een schaakspel kan beschreven worden. FEN bevat alle informatie die nodig is om een schaakspel vanaf een bepaalde spelsituatie te kunnen hervatten. Het systeem werd ontwikkeld door David Forsyth — journalist van een Schotse krant — die het reeds in 19e eeuw populariseerde. Steven J. Edwards breidde FEN later uit zodat het ook door computers kon gebruikt worden.



Beginopstelling van een schaakspel vanuit het gezichtspunt van speler wit. Op de onderste rij staan vanaf links toren (R), paard (N), loper (B), dame (D) en koning (K). Op de tweede rij staan de pionnen.

In het eerste deel van FEN (we laten hier gemakshalve een aantal onderdelen van een spelsituatie vallen, zoals bijvoorbeeld wie als volgende aan zet is) wordt de plaatsing van de stukken op het schaakbord beschreven vanuit het gezichtspunt van speler wit. De rijen worden beschreven van boven naar onder, en de bezetting van de velden op elke rij wordt beschreven van links naar rechts. Volgens de [Standaard Algebraïsche notatie](#) (SAN) wordt elk schaakstuk omschreven met één enkele letter: K = koning, Q = dame/koningin, R = toren/kasteel, B = loper/raadsheer, N = paard, P = pion. Witte stukken worden aangeduid met een hoofdletter (KQRBNP) en zwarte stukken met een kleine letter (kqrbnp). Lege velden worden aangeduid met de cijfers 1 tot en met 8, die het aantal opeenvolgende lege velden aanduiden. Een schuine streep (/) wordt gebruikt om de rijen van elkaar te scheiden.

## Opgave

- Schrijf een functie `fen2grid` waarmee de opstelling van een schaakbord in FEN kan omgezet worden naar een stringvoorstelling in de vorm van een  $8 \times 8$  rooster. Bij deze laatste omschrijving worden de rijen van elkaar gescheiden door newlines (`\n`) en wordt elk leeg veld aangeduid met één bepaald ASCII karakter. De stukken worden met dezelfde letters aangeduid als bij FEN. Aan deze functie moet als eerste argument een string met de FEN omschrijving doorgegeven worden. Als tweede optionele argument kan men aan de functie ook nog het ASCII karakter doorgeven waarmee lege velden aangeduid worden (standaard wordt hiervoor een sterretje (\*) gebruikt). De functie moet de string met de roostervoorstelling als resultaat teruggeven.
- Schrijf een functie `grid2fen` die precies het omgekeerde doet als de functie `fen2grid`, dus een omschrijving in de vorm van een rooster omzetten in een omschrijving in FEN. Als eerste argument moet een string met de roosteromschrijving doorgegeven worden, en als tweede optionele argument een karakter dat aangeeft hoe lege velden voorgesteld worden in de roosteromschrijving. Het sterretje (\*) wordt hierbij opnieuw als standaardwaarde gebruikt. De functie moet als resultaat de string met de FEN omschrijving teruggeven.

## Voorbeeld

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

\*\*\*\*\*

PPPPPPPP

RNBQKBNR

```
>>> print(fen2grid('rnbqkbnr/pppppppp/8/8/4P3/8/PPPP1PPP/RNBQKBNR', '.'))
```

rnbqkbnr

pppppppp

.....

.....

....P...

.....

PPPP.PPP

RNBQKBNR

```
>>> print(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPPP1PPP/RNBQKBNR', '+'))
```

rnbqkbnr

pp+ppppp

+++++++

++p++++

++++P+++

+++++++

PPPP+PPP

RNBQKBNR

```
>>> rooster = "rnbqkbnr
```

```
... pppppppp
```

```
... *****
```

```
... *****
```

```
... *****
```

```
... *****
```

```
... *****
```

```
... P P P P P P P P
```

```
... R N B Q K B N R"
```

```
>>> print(grid2fen(rooster))
```

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

```
>>> print(grid2fen(fen2grid('rnbqkbnr/pppppppp/8/8/4P3/8/PPPP1PPP/RNBQKBNR')))
```

rnbqkbnr/pppppppp/8/8/4P3/8/PPPP1PPP/RNBQKBNR

```
>>> print(grid2fen(fen2grid('rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPPP1PPP/RNBQKBNR', '.'), '.'))
```

rnbqkbnr/pp1ppppp/8/2p5/4P3/8/PPPP1PPP/RNBQKBNR

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

rnbqkbnr/pp1ppppp/8/2p5/4P3/5N2/PPPP1PPP/RNBQKB1R