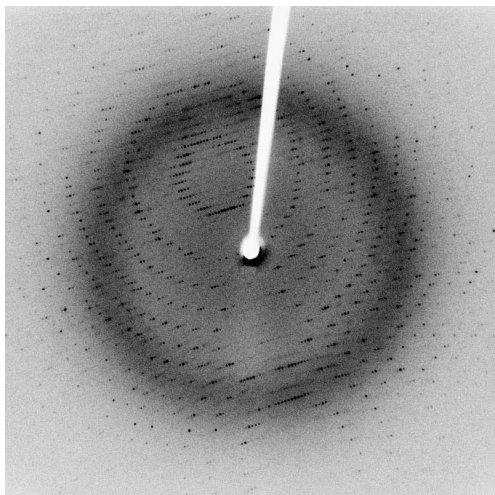


X-ray diffraction

X-ray diffraction is a tool used for identifying the atomic and molecular structure of a crystal, in which the crystalline atoms cause a beam of incident X-rays to diffract into many specific directions. By measuring the angles and intensities of these diffracted beams, a crystallographer can produce a three-dimensional picture of the density of electrons within the crystal. From this electron density, the mean positions of the atoms in the crystal can be determined, as well as their chemical bonds, their disorder and various other information.



X-ray diffraction pattern formed when X-rays are focused on a crystalline material, in this case a protein. Each dot, called a reflection, forms from the coherent interference of scattered X-rays passing through the crystal.

Since many materials can form crystals — such as salts, metals, minerals, semiconductors, as well as various inorganic, organic and biological molecules — X-ray crystallography has been fundamental in the development of many scientific fields. In its first decades of use, this method determined the size of atoms, the lengths and types of chemical bonds, and the atomic-scale differences among various materials, especially minerals and alloys. The method also revealed the structure and function of many biological molecules, including vitamins, drugs, proteins and nucleic acids such as DNA. X-ray crystallography is still the chief method for characterizing the atomic structure of new materials and in discerning materials that appear similar by other experiments. X-ray crystal structures can also account for unusual electronic or elastic properties of a material, shed light on chemical interactions and processes, or serve as the basis for designing pharmaceuticals against diseases.

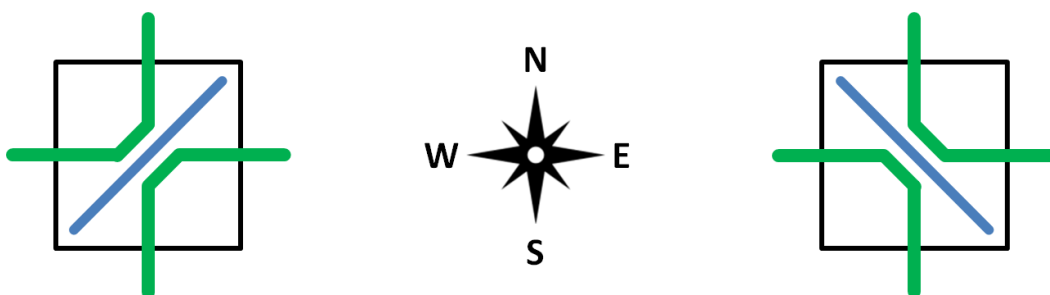
In a single-crystal X-ray diffraction measurement, a crystal is mounted on a goniometer. The goniometer is used to position the crystal at selected orientations. The crystal is bombarded with a finely focused monochromatic beam of X-rays, producing a diffraction pattern of regularly spaced spots known as reflections. The two-dimensional images taken at different rotations are converted into a three-dimensional model of the density of electrons within the crystal using the mathematical method of Fourier transforms, combined with chemical data known for the sample. Poor resolution (fuzziness) or even errors may result if the crystals are too small, or not uniform enough in their internal makeup.

Assignment

We simulate the scattering of X-rays that hits the atoms of a planar molecule, by subdividing the

rectangular area occupied by the molecule into m rows and n columns ($m, n \in \mathbb{N}_0$). Each atom of the molecule occupies one of the cells of this rectangular grid, and cells never contain more than one atom.

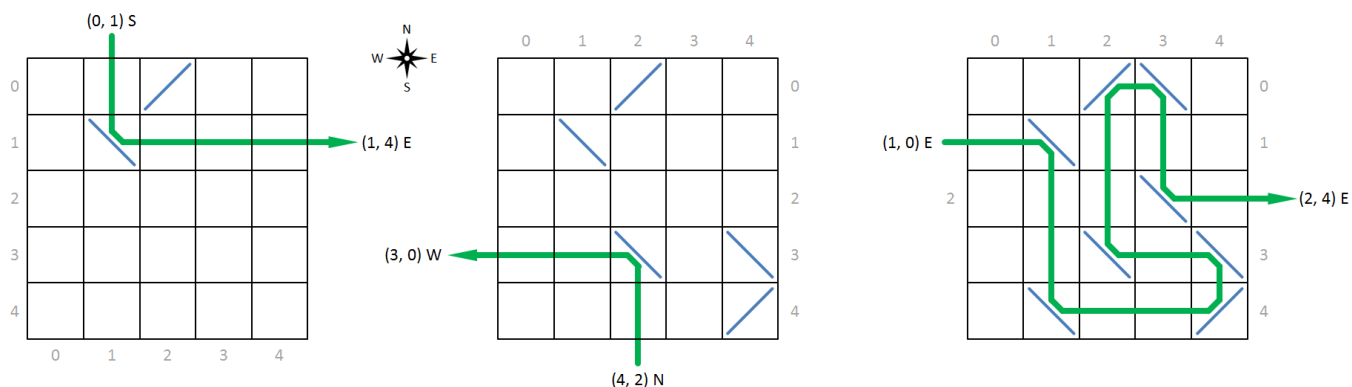
We always bombard the molecules with X-rays that are oriented either to the north, east, south or west. Based on the way they deflect incoming X-rays, atoms fall in one of two categories. Atoms having **forward deflection** (left in figure below) bend X-rays directed towards the east to the north (and *vice versa*) and bend X-rays directed toward the south to the west. Atoms having **backward deflection** (right in figure below) bend X-rays directed toward the east to the south (and *vice versa*) and bend X-rays directed toward the north to the west (and *vice versa*).



Based on the way they deflect incoming X-rays, atoms fall in one of two categories. Atoms having **forward deflection** (left in figure below) bend X-rays directed towards the east to the north (and *vice versa*) and bend X-rays directed toward the south to the west. Atoms having **backward deflection** (right in figure below) bend X-rays directed toward the east to the south (and *vice versa*) and bend X-rays directed toward the north to the west (and *vice versa*).

In order to reference specific cells in the grid, we increasingly index the rows from north to south and the columns from west to east, with indexing always starting at zero (gray indices in the figure below). This allows us to represent the position of a cell in the grid as a tuple (r, c) , with r the row index and c the column index of the cell in the grid. We describe an X-ray which is incident on a molecule by the position of the cell where the beam is incident on the molecule, together with the direction to which the beam is directed. Your task is to determine for a given incident X-ray where and in which direction it will leave the molecule.

Below, we show some examples of X-rays that are incident on a molecule and are deflected by its molecules. The molecule to the left, for example has an atom with forward deflection at position $(0, 2)$ and an atom with backward deflection at position $(1, 1)$. As indicated by the green line, an X-ray directed to the south that hits the molecule at position $(0, 1)$, is bent to the east by the atom at position $(1, 1)$ and leaves the molecule in that direction at position $(1, 4)$.



Examples of X-rays that are incident on a molecule and are deflected by its molecules. The molecule to the left, for example has an atom with forward deflection at position $(0, 2)$ and an

atom with backward deflection at position (1, 1). As indicated by the green line, an X-ray directed to the south that hits the molecule at position (0, 1), is bent to the east by the atom at position (1, 1) and leaves the molecule in that direction at position (1, 4).

Your task is to write a class `Molecule` whose objects represent a planar molecule. The atoms of this molecule are organized in a rectangular $m \times n$ grid. Each cell of the grid may contain an atom having either forward or backward deflection, that behaves as described above. The class must support at least the following methods:

- An initialization method that takes the number of rows $m \in \mathbb{N}_0$ and the number of columns $n \in \mathbb{N}_0$ of the rectangular grid. After initialization of an object, the grid does not contain any atoms yet, but during the lifetime of the object atoms having forward or backward deflection may be placed on the grid by calling the methods `atom` and `atoms`.
- A method `atom` that can be used to place an atom on the grid. The method takes the position of the grid cell where the atom must be placed. In case the given position falls outside the grid or if this position is already occupied by another atom, the method must raise an `AssertionError` with the message `invalid position`. By default, the method places an atom having forward deflection. However, the method also has a second optional parameter `forward` that takes a Boolean value (default value: `True`). In case the value `False` is passed to this parameter, the method must place an atom having backward deflection.
- A method `atoms` that can be used to place one or more atoms having the same type of deflection on the grid. The positions of these atoms must be passed to the method as a collection (a list, tuple or set). These positions must meet the same conditions as the position passed to the method `atom`. If this is not the case, the method must raise an `AssertionError` with the message `invalid position`. The method also has a second optional parameter `forward` that must be interpreted in the same way as for the method `atom`, and indicates the type of deflection of the atoms that must be placed on the grid.
- A method `__str__` that returns a string representation of the atoms that have been placed on the grid. This string representation contains m lines that each contain n space-separated symbols. The c -th symbol at the r -th line (lines and symbols are numbered starting from zero) indicates if the cell at position (r, c) is occupied by an atom, and the type of deflection of this atom. Atoms having forward deflection are represented by a slash (`/`) and atoms having backward deflection are represented by a backward slag (`\`). Cells that are not occupied by an atom are indicated by a dot (`.`).
- A method `deflection` that takes two arguments representing the coordinates of an incident X-ray: *i*) the position of the cell where the beam hits the molecule and *ii*) the direction in which the beam is directed. This direction is represented by a single letter: N for north, S for south, W for west and E for east. The method must return a list of positions in the grid that indicate the order of the cells visited by the X-ray as it passes through the molecule. In passing the molecule, the beam is deflected by its atoms as described in the introduction of this assignment.
- A method `diffraction` that takes the same two arguments (with the same meaning) as the method `deflection`. The method must return a tuple containing two elements: *i*) the position where the beam leaves the molecule, and *ii*) the direction in which the beam leaves the molecule.

Note: Strings in Python use the backslash character as an escape symbol. Therefore, literal backslashes must be doubled in a string: `\\`.

Example

Note that a docstring is nothing but an ordinary string, which requires that each backslash that occurs literally in the docstring must be repeated twice. [Click here](#) to convert the following interactive session into a docstring.

```
>>> molecule = Molecule(5, 5)
>>> molecule.atom((0, 2))
>>> molecule.atom((1, 1), forward=False)
>>> print(molecule)
.. / ..
. \ ...
.....
.....
.....
>>> molecule.deflection((0, 1), 'S')
[(0, 1), (1, 1), (1, 2), (1, 3), (1, 4)]
>>> molecule.diffraction((0, 1), 'S')
((1, 4), 'E')

>>> molecule.atom((4, 4))
>>> molecule.atoms([(3, 2), (3, 4)], forward=False)
>>> print(molecule)
.. / ..
. \ ...
.....
.. \ . \
.... /
>>> molecule.deflection((4, 2), 'N')
[(4, 2), (3, 2), (3, 1), (3, 0)]
>>> molecule.diffraction((4, 2), 'N')
((3, 0), 'W')

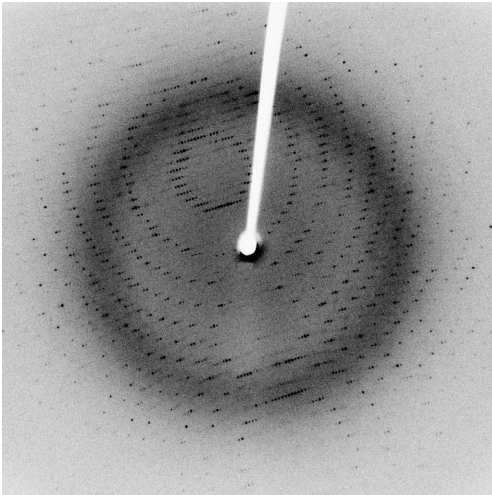
>>> molecule.atoms([(0, 3), (2, 3), (4, 1)], forward=False)
>>> print(molecule)
.. / \ .
. \ ...
... \ .
.. \ . \
. \ . . /
>>> molecule.deflection((1, 0), 'E')
[(1, 0), (1, 1), (2, 1), (3, 1), (4, 1), (4, 2), (4, 3), (4, 4), (3, 4), (3, 3), (3, 2), (2, 2), (1, 2), (0, 2), (0, 3), (1, 3), (2, 3), (2, 4)]
>>> molecule.diffraction((1, 0), 'E')
((2, 4), 'E')

>>> molecule.atom((6, 6))
Traceback (most recent call last):
AssertionError: invalid position

>>> molecule.atom((1, 1))
Traceback (most recent call last):
AssertionError: invalid position
```

Röntgendiffractie is een techniek die gebruikt wordt om de structuur van vaste stoffen te bepalen. Als de techniek toegepast wordt op kristallen, dan kan daarmee de kristalstructuur zeer nauwkeurig bepaald worden. Uit het diffractiepatroon — de verstrooiing van de röntgenstralen — is het immers mogelijk om de positie van de atomen in de moleculen te reconstrueren. In de

vroege jaren '50 begonnen wetenschappers deze techniek toe te passen op biologische moleculen zoals DNA en eiwitten. De techniek wordt ook gebruikt voor het bepalen van de structuur van mineralen, eenvoudige en meer complexe organische verbindingen.



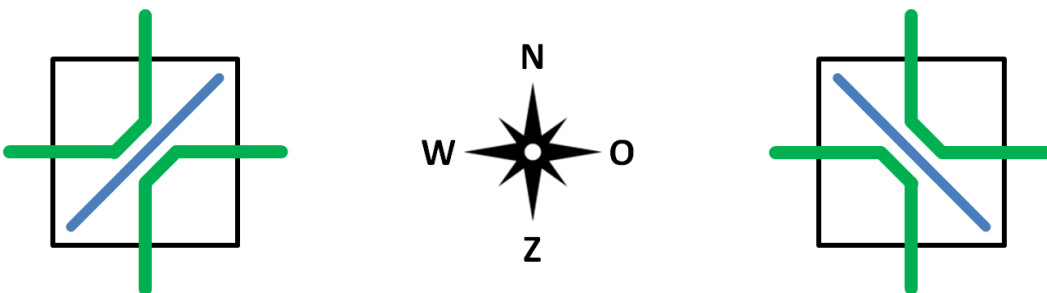
Diffractionpatroon dat ontstaat wanneer röntgenstralen gericht worden op een kristallijn materiaal, in dit geval een eiwit. Elk stip in het patroon wordt een reflectie genoemd, en ontstaat door een coherente interferentie van verstrooide röntgenstralen die door het eiwit passeren.

Röntgendiffractie is een zeer nauwkeurige microscopische techniek. Waar atoomkrachtmicroscopie op dit ogenblik maar net toelaat om individuele atomen te zien, kan de positie van de atomen met röntgendiffractie nog 100-1000 maal nauwkeuriger bepaald worden.

Opgave

We modelleren de verstrooiing van röntgenstralen die aanbotsen tegen de atomen van een vlakke molecule, door het rechthoekig gebied dat wordt ingenomen door de molecule onder te verdelen in m rijen en n kolommen ($m, n \in \mathbb{N}_0$). Elk atoom van de molecule ligt in een cel van dit rechthoekig rooster, en cellen bevatten nooit meer dan één atoom.

De röntgenstralen die we laten invallen op de molecule zijn steeds gericht naar het noorden, oosten, zuiden of westen. Op basis van de manier waarop ze invallende röntgenstralen afbuigen worden atomen onderverdeeld in twee categorieën. Atomen met **voorwaartse reflectie** (links in onderstaande figuur) buigen röntgenstralen die naar het oosten gericht zijn af naar het noorden (en *vice versa*) en buigen röntgenstralen die naar het zuiden gericht zijn af naar het westen (en *vice versa*). Atomen met een **achterwaartse reflectie** (rechts in onderstaande figuur) buigen röntgenstralen die naar het oosten gericht zijn af naar het zuiden (en *vice versa*) en buigen röntgenstralen die naar het noorden gericht zijn af naar het westen *vice versa*.

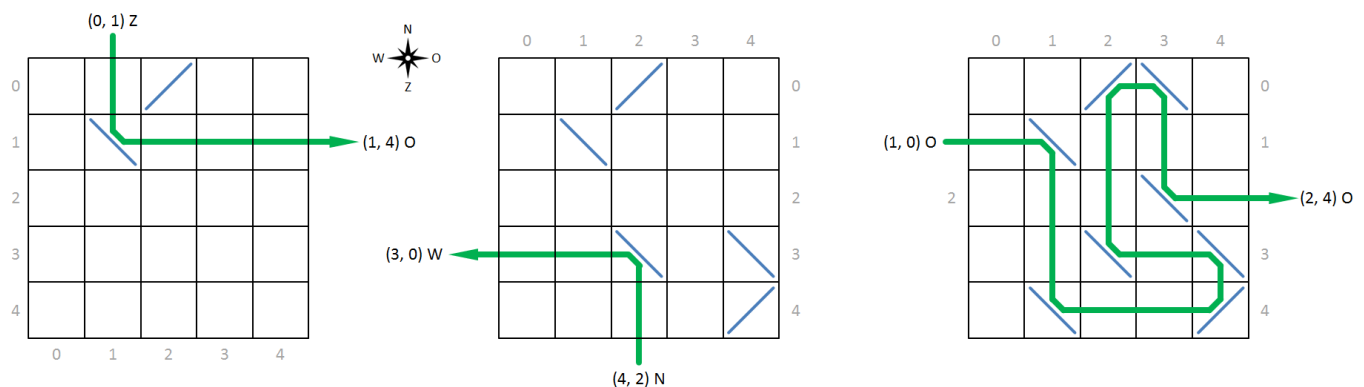


Op basis van de manier waarop ze invallende röntgenstralen afbuigen worden atomen onderverdeeld in twee categorieën. Atomen met **voorwaartse reflectie** (links) buigen röntgenstralen die naar het oosten gericht zijn af naar het noorden (en *vice versa*) en buigen

röntgenstralen die naar het zuiden gericht zijn af naar het westen (en *vice versa*). Atomen met een **achterwaartse reflectie** (rechts) buigen röntgenstralen die naar het oosten gericht zijn af naar het zuiden (en *vice versa*) en buigen röntgenstralen die naar het noorden gericht zijn af naar het westen *vice versa*).

Om te kunnen verwijzen naar specifieke cellen in het rooster, nummeren we de rijen oplopend van noord naar zuid en de kolommen van west naar oost, waarbij we steeds beginnen te nummeren vanaf nul (grijze getallen op onderstaande figuur). We kunnen de positie van een cel in het rooster dan voorstellen door een tuple (r, k) , waarbij r het nummer van de rij en k het nummer van de kolom is waarop de cel gelegen is. Een röntgenstraal die invalt op een molecule beschrijven we aan de hand van de positie van de cel waar de straal op de molecule invalt, samen met de richting waarnaar de straal gericht is. Je opdracht bestaat erin om voor een invallende röntgenstraal te bepalen waar en in welke richting hij de molecule zal verlaten.

Hieronder tonen we enkele voorbeelden van röntgenstralen die invallen op een molecule en door de atomen afgebogen worden. De linkse molecule heeft bijvoorbeeld een atoom met voorwaartse reflectie op positie $(0, 2)$ en een atoom met achterwaartse reflectie op positie $(1, 1)$. Zoals aangegeven door de groene lijn zal een röntgenstraal die naar het zuiden gericht is en invalt op positie $(0, 1)$ door het atoom op positie $(1, 1)$ afgebogen worden naar het oosten en de molecule in die richting verlaten op positie $(1, 4)$.



Voorbeelden van röntgenstralen die invallen op een molecule en door de atomen afgebogen worden. De linkse molecule heeft bijvoorbeeld een atoom met voorwaartse reflectie op positie $(0, 2)$ en een atoom met achterwaartse reflectie op positie $(1, 1)$. Zoals aangegeven door de groene lijn zal een röntgenstraal die naar het zuiden gericht is en invalt aan de cel op positie $(0, 1)$ door het atoom op positie $(1, 1)$ afgebogen worden naar het oosten en de molecule verlaten aan de cel op positie $(1, 4)$.

Gevraagd wordt om een klasse `Molecule` te schrijven waarvan elk object een molecule voorstelt in de vorm van een rechthoekig $m \times n$ rooster. Elke cel van het rooster kan een atoom bevatten met voorwaartse of achterwaartse reflectie, dat zich gedraagt zoals hierboven beschreven. De klasse moet minstens de volgende methoden ondersteunen:

- Een initialisatiemethode waaraan het aantal rijen $m \in \mathbb{N}_0$ en het aantal kolommen $n \in \mathbb{N}_0$ van het rooster moet doorgegeven worden. Na initialisatie van een object bevinden er zich nog geen atomen op het rooster, maar tijdens de levensduur van het object kunnen atomen met voorwaartse of achterwaartse reflectie op het rooster geplaatst worden door het aanroepen van de methoden `atoom` en `atomen`.
- Een methode `atoom` waarmee een atoom op het rooster kan geplaatst worden. Aan de methode moet de positie doorgegeven worden van de cel waarop het atoom moet

geplaatst worden. Indien de gegeven positie buiten het rooster ligt of indien op deze positie reeds een atoom geplaatst werd, dan moet de methode een `AssertionError` opwerpen met de boodschap `ongeldige positie`. Standaard heeft het te plaatsen atoom een voorwaartse reflectie. De methode heeft echter ook nog een tweede optionele parameter `voorwaarts` waaraan een Booleaanse waarde kan doorgegeven worden (standaardwaarde: `True`). Indien aan deze parameter de waarde `False` doorgegeven wordt, dan heeft het te plaatsen atoom een achterwaartse reflectie.

- Een methode `atomen` waarmee één of meer atomen met dezelfde soort reflectie op het rooster kunnen geplaatst worden. De posities van deze atomen moeten als een collectie (een lijst, tuple of verzameling) aan de methode doorgegeven worden. Deze posities moeten aan dezelfde voorwaarden voldoen als de positie die wordt doorgegeven aan de methode `atoom`. Indien dit niet het geval is, dan moet de methode een `AssertionError` opwerpen met de boodschap `ongeldige positie`. De methode heeft ook nog een tweede optionele parameter `voorwaarts` die op dezelfde manier moet geïnterpreteerd worden als bij de methode `atoom`, en die het soort reflectie aangeeft van de atomen die op het rooster moeten geplaatst worden.
- Een methode `__str__` die een stringvoorstelling teruggeeft van de atomen die reeds op het rooster geplaatst werden. Deze stringvoorstelling bestaat uit `m` regels, die elk `n` symbolen bevatten die telkens van elkaar gescheiden worden door een spatie. Het `k`-de symbool op de `r`-de regel (regels en symbolen genummerd vanaf nul) geeft aan of de cel op positie `(r, k)` een atoom bevat, en welk soort reflectie dat atoom heeft. Atomen met voorwaartse reflectie worden voorgesteld door een slash (`/`) en atomen met achterwaartse reflectie door een backslash (`\`). Cellen zonder atoom worden aangeduid met een punt (`.`).
- Een methode `afbuiging` waaraan twee argumenten moeten doorgegeven worden, die de coördinaten van een invallende röntgenstraal voorstellen: *i*) de positie van de cel waar de straal op de molecule invalt en *ii*) de richting waarnaar de straal gericht is. Deze richting wordt aangeduid met één van de volgende letters: N voor noord, Z voor zuid, W voor west en O voor oost. De methode moet een lijst van posities in het rooster teruggeven die de volgorde aangeeft van de cellen waarlangs de röntgenstraal passeert bij zijn doortocht doorheen de molecule. Hierbij wordt de straal door de atomen afgebogen zoals beschreven in de inleiding van de opgave.
- Een methode `diffRACTIE` waaraan dezelfde twee argumenten (met dezelfde betekenis) moeten doorgegeven worden als aan de methode `afbuiging`. De methode moet een tuple teruggeven dat twee elementen bevat: *i*) de positie van de cel waar de invallende röntgenstraal de molecule terug verlaat, en *ii*) de richting waarin de straal de molecule verlaat.

Opmerking: Strings in Python gebruiken het backslash karakter als escape symbool. Daarom moet een letterlijke backslash in een string verdubbeld worden: `\\`.

Voorbeeld

Merk op dat een docstring een gewone string is, waardoor elke letterlijke backslash die voorkomt in de docstring moet verdubbeld worden. [Klik hier](#) om onderstaande interactieve sessie om te vormen tot een docstring.

```
>>> molecule = Molecule(5, 5)
>>> molecule.atoom((0, 2))
>>> molecule.atoom((1, 1), voorwaarts=False)
>>> print(molecule)
... / ...
```

```

.\ . . .
. . . .
. . . .
. . . .
>>> molecule.afbuiging((0, 1), 'Z')
[(0, 1), (1, 1), (1, 2), (1, 3), (1, 4)]
>>> molecule.diffractie((0, 1), 'Z')
((1, 4), 'O')

>>> molecule.atoom((4, 4))
>>> molecule.atomen([(3, 2), (3, 4)], voorwaarts=False)
>>> print(molecule)
.. / ..
.\ . . .
. . . .
.. \ . \
. . . . /
>>> molecule.afbuiging((4, 2), 'N')
[(4, 2), (3, 2), (3, 1), (3, 0)]
>>> molecule.diffractie((4, 2), 'N')
((3, 0), 'W')

>>> molecule.atomen([(0, 3), (2, 3), (4, 1)], voorwaarts=False)
>>> print(molecule)
.. / \ .
.\ . . .
. . . \ .
.. \ . \
.\ . . /
>>> molecule.afbuiging((1, 0), 'O')
[(1, 0), (1, 1), (2, 1), (3, 1), (4, 1), (4, 2), (4, 3), (4, 4), (3, 4), (3, 3), (3, 2), (2, 2), (1, 2), (0, 2), (0, 3), (1, 3), (2, 3), (2, 4)]
>>> molecule.diffractie((1, 0), 'O')
((2, 4), 'O')

>>> molecule.atoom((6, 6))
Traceback (most recent call last):
AssertionError: ongeldige positie

>>> molecule.atoom((1, 1))
Traceback (most recent call last):
AssertionError: ongeldige positie

```