# OWE2a Themaopdracht Kans 1

Student:

Klas:

**Begindatum:** donderdag 25 januari 2018, 9.00 uur
**Uiterlijke inleverdatum:** donderdag 1 februari 2018, 9.00 uur

## Casus

Je loopt stage bij de afdeling Humane Genetica van het Radboud UMC. Hier doet men onderzoek naar onder andere erfelijke vormen van kanker. Jouw opdracht is om nieuwe kandidaat genen te vinden voor erfelijke borstkanker. Als eerste verkenning heb je een bestand gekregen met alle menselijke genen en hun omschrijving.

Hieronder vind je een stukje uit het bestand:

---

| Symbol | Description | Chromosome | Aliases |
|---|---|---|---|
| A1BG | alpha-1-B glycoprotein | 19 | A1B, ABG, GAB, HYST2477 |
| A2M | alpha-2-macroglobulin | 12 | A2MD, CPAMD5, FWP007, S863-7 |
| A2MP1 | alpha-2-macroglobulin pseudogene 1 | 12 | A2MP |

...

---

Zoals je kan zien gaat het om een tab delimited bestand met vier kolommen; Symbol, Description, Chromosome en Aliases.

Jouw stagebegeleider geeft je de opdracht om alle genen betrokken bij borstkanker en de bijhorende aliassen uit dit bestand te filteren.

# Opdracht

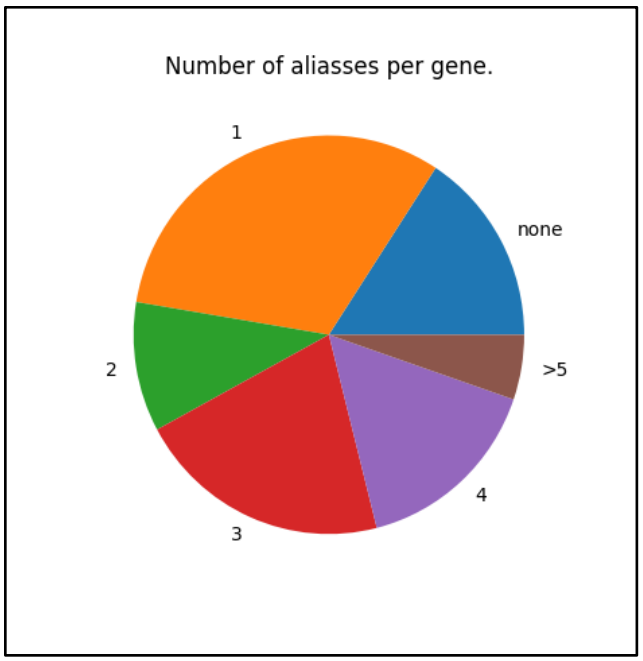
Je stagebegeleider wil graag dat je het volgende oplevert:

- Scherm output met per borstkanker gen de bijhorende aliassen
- Een grafiek met hoeveel borstkanker genen welk aantal aliassen hebben
    - Bijv. 3 genen die geen aliassen hebben, 5 genen die 1 alias hebben, 2 genen die 2 aliassen hebben, etc
        - N.B. deze getallen kloppen niet, slechts ter illustratie.
    - Bedenk of je een cutoff waarde wilt of niet
        - Dus een aantal waarboven je alle genen onder dezelfde noemer schuift, bijv. >6

De output kan er als volgt uitzien:

---

```
##########################
Total number of breast cancer genes: 10
##########################
Gene:        BRCA1
Aliases:     BRCAI, BRCC1, BROVCA1, FANCS, IRIS, PNCA4, PPP1R53
##########################
Gene: BRCA2
Aliases:     BRCC2, BROVCA2, FACD, FAD, FAD1, FANCD, FANCD1, GLM3
##########################
...
```




---

N.B. De hierboven genoemde aantallen kloppen niet. Dit is slechts ter illustratie. Voel je ook vrij om je eigen output en grafiek te maken, dit hoeft echt niet per se een taartdiagram te zijn.

# Eisen

Het op te leveren programma dient aan de volgende functionele eisen te voldoen:

1. Het script leest het bestand, bepaald de gevraagde eigenschappen per gen en verwerkt de data voor een grafiek
2. Het programma geeft output waarbij de gebruiker een overzicht krijgt van de aliassen van de genen en het totaal aan aliassen.

Het op te leveren programma dient aan de volgende technische (niet functionele) eisen te voldoen:

1. Het script is opgedeeld in functies. De volgende functies zijn in ieder geval aanwezig.
    - De functie **read_file()** leest het bestand in en maakt een dictionary met hierin alle genen en hun eigenschappen. De functie retourneert de dictionary.
    - De functie **search()** accepteert de dictionary en bekijkt welke van de genen betrokken zijn bij borstkanker aan de hand van de Description kolom.
        - Dit wordt gedaan met een regular expression die tegelijkertijd op de volgende termen zoekt (met of zonder hoofdletters):
            1. Breast cancer
            2. Breast carcinoma
            3. Ductal carcinoma
        - De functie retourneert een nieuwe dictionary met daarin als key::value paar Symbol::Aliases
    - De functie **generate_output()** print voor ieder borstkankergen de bijhorende aliassen naar de shell
    - De functie **visualize()** maakt een grafiek aan de hand van de genen en hun bijhorende aantal aliassen
        - Er wordt dus geteld hoeveel aliassen ieder gen heeft en dit wordt geplot naar een grafiek (x aantal genen heeft y aantal aliassen)
    - De functie **main()** roept al deze functies aan.
2. Het programma maakt gebruik van pickle voor het opslaan van een dictionary
    - Deze dictionary heeft als key::value paar de gennaam (Symbol) en de bijhorende aliassen (Aliases).
    - Deze wordt gepickled en bij opstarten van het programma wordt gekeken of er niet direct een dictionary in te laden is
    - Zo niet, dan genereert het programma een nieuwe dictionary aan de hand van het humane genen bestand en pickled deze alsnog.
3. Het programma maakt gebruik van gepaste exception handling om eventueel te verwachte errors af te vangen.
4. Voel je vrij om je eigen draai te geven aan de functies als je voelt dat het efficiënter of handiger kan. Zorg er wel voor dat de elementen die hierboven beschreven staan terugkomen in je code (grafiek, pickling, dictionaries, regular expressions, exception handling).
5. De snelheid van het programma doet er niet toe.
6. Het programma is geschreven in Python.

# Beoordeling

De beoordeling van het totale tentamen is als volgt.

| **Criterium** | | | **Max punten** | **Beoordeling** |
|---|---|---|---|---|
| 1 | Python Code (30) | | | |
| | a. | Commentaar | 10 | |
| | b. | Onderverdeling in functies | 10 | |
| | c. | Datatypes en variabelen | 10 | |
| 2 | Juiste werking (70) | | | |
| | a. | Parsen bestand naar dictionary | 10 | |
| | b. | Werken met dictionaries | 10 | |
| | c. | Regular expressions | 10 | |
| | d. | Matplotlib | 10 | |
| | e. | Exception handling | 20 | |
| | f. | Pickling | 10 | |
| Totaal | | | 100 | |