## Opdracht 1: dobbelstenen

In deze opdracht ga je dobbelsteen gooien simuleren met behulp van python loops. Dobbelstenen bestaan uit zes vlakken met daarop elk een getal. De minimale waarde van een dobbelsteen is één, de maximale waarde is zes.

1. Importeer de Random module. (1pt)
2. Simuleer één dobbelsteengooi van één dobbelsteen. Wat heb je gegooid? Laat Python de gegooide waarde weergeven als uitvoer. *Hint: maak hierbij gebruik van de bij vraag a geïmporteerde Random module.* (2pt)
3. Simuleer nu 50 dobbelsteengooien van twee dobbelstenen met behulp van een Python loop. Bereken en geef als uitvoer wat je in totaal hebt gegooid (tel alle gooien bij elkaar op). *Hint: maak hierbij gebruik van een while loop en de Random module.* (3pt)
4. Bereken van de 50 dobbelsteengooien van twee dobbelstenen wat je gemiddeld hebt gegooid. Rond af naar het dichtstbijzijnde gehele getal (int) en geeft dit als uitvoer. Is de vorige vraag niet gelukt, neem dan een realistisch getal en bereken dan alsnog het gemiddelde van de 50 dobbelsteengooien van 2 dobbelstenen. (2pt)
5. Simuleer nu 50 dobbelsteengooien van één dobbelsteen. De uitvoer moet een lijst zijn met alle 50 gooien. Hint: maak eerst een lege lijst. (3pt)
6. Laat zien dat je het 20ste element uit de lijst met dobbelsteengooien kan selecteren. Is de vorige vraag niet gelukt, maak dan een lijst aan met 50 elementen en maak deze vraag alsnog. (1pt)
7. Maak een histogram plot van de verkregen lijst uit vraag 5 met behulp van Matplotlib en de plt.hist() functie. Voeg een titel, x-label en y-label toe. *Hint: voor een mooie grafiek gebruik bins = 6 of bins = 11.* (2pt)

## Opdracht 2: wiskundige vergelijkingen

In deze opdracht ga je de onderstaande wiskundige vergelijkingen plotten met behulp van NumPy en Matplotlib.

1. Importeer NumPy en Matplotlib. (1pt)
2. Plot , en met behulp van NumPy, math.pi en Matplotlib samen in een grafiek. Het bereik van x gaat van nul tot tien in 1000 gelijke stappen. Voeg een titel, legenda, xlabel en ylabel toe. *Hint: maak eerst een NumPy array van 1000 elementen die begint bij nul en eindigt bij tien.* (10pt)