# Approximating pi

There are different ways to calculate the value of the number $\pi$ by means of a computer. You can, for example, calculate the partial sums of the Gregory-Leibniz sequence: $$ \pi = \frac41 - \frac43 + \frac45 - \frac47 + \dots $$ The Indian mathematician Madhaya of Sangamagrama proposed an alternative sequence development in the fourteenth century: $$ \pi = \sqrt{12}\,\left( 1 - \frac1{3\cdot 3} + \frac1{5\cdot 3^2} - \frac1{7\cdot 3^3} + \cdots \right) $$

## Assignment

- Write a function `GL` that prints the sum of the first *n* terms of the Gregory-Leibniz sequence. The amount of terms $n$ should be given to the function as a parameter.
- Write a function `MvS` that calculates the sum of the first *n* terms of the Madhava of Sangamagrama sequence. The number of terms $n$ should be given to the function as a parameter.
- Examine which of both sequences converges the fastest. Use this sequence to write a function `approach_pi`. This function should allow to determine an approached value of $\pi$, that is precise to $n$ decimals. The value $n$ should be given as an argument of the function. To determine the accuracy of the approach, you should check whether the difference between two consecutive **terms** in the sequence is smaller than $10^{-n-1}$. When the difference between the ($i-1$)th and the $i$th **term** becomes smaller than $10^{-n-1}$, the $i$th partial sum forms an approach of $\pi$ to $n$ decimals precisely. The function should give the *tuple* $(i, p)$ as a result, $i$ is the number of calculated terms and $p$ is the approached value of $\pi$.

## Example

```
>>> GL(2)
2.666666666666667
>>> MvS(2)
3.0792014356780038
>>> approach_pi(2)
(8, 3.1416743126988376)
```

Er bestaan verschillende manieren om de waarde van het getal $\pi$ te berekenen aan de hand van een computer. Zo kan je bijvoorbeeld een partieelsom berekenen van de reeks van Gregory-Leibniz: $$ \pi = \frac41 - \frac43 + \frac45 - \frac47 + \dots $$ De Indische wiskundige Madhava van Sangamagrama stelde reeds in de veertiende eeuw een alternatieve reeksontwikkeling voor: $$ \pi = \sqrt{12}\,\left( 1 - \frac1{3\cdot 3} + \frac1{5\cdot 3^2} - \frac1{7\cdot 3^3} + \cdots \right) $$

## Opgave

- Schrijf een functie `GL` die de som van de eerste *n* termen van de reeks van Gregory-Leibniz teruggeeft. Het aantal termen $n$ moet als parameter aan de functie meegegeven worden.
- Schrijf een functie `MvS` die de som van de eerste *n* termen van de reeks van Madhava van Sangamagrama teruggeeft. Het aantal termen $n$ moet als parameter aan de functie meegegeven worden.
- Onderzoek welk van deze twee reeksen het snelst convergeert. Gebruik deze reeks om

een functie benader_pi te schrijven. Deze functie moet toelaten om een benaderde waarde voor $\pi$ te bepalen, die tot op $n$ decimalen nauwkeurig is. De waarde $n$ moet als argument aan de functie meegegeven worden. Om de nauwkeurigheid van de benadering te bepalen, moet je nagaan of het verschil tussen twee opeenvolgende **termen** in de reeks kleiner is dan $10^{-n-1}$. Op het ogenblik dat het verschil tussen de ($i-1$)-de en de $i$-de **term** kleiner wordt dan $10^{-n-1}$, vormt de $i$-de partieelsom een benadering van $\pi$ tot op $n$ decimalen nauwkeurig. De functie moet het *tupel* $(i, p)$ als resultaat teruggeven, waarbij $i$ het aantal berekende termen is en $p$ de benaderde waarde van $\pi$.

## Voorbeeld

```
>>> GL(2)
2.666666666666667
>>> MvS(2)
3.0792014356780038
>>> benader_pi(2)
(8, 3.1416743126988376)
```