# Pi

Alle som har jobbet med matematikk etter barneskolen har på ett eller annet tidspunkt kommet bort i den matematiske konstanten π (pi). Den kommer fram blant annet når man beregner sirkler, i trigonometri og i det absolutte vinkelmålet radianer. Pi har blitt brukt helt siden før vår tidsregning, men hva betyr egentlig pi og hvordan kan vi komme fram til en tilnærming av den eksakte verdien til denne konstanten? Disse spørsmålene ønsker jeg å besvare i denne artikkelen.

# Hva er pi?

Pi er definert som forholdet mellom omkretsen til en sirkel og diameteren til en sirkel, et forhold som er likt for alle størrelser. Dette kan vi utlede fra formelen for omkretsen av en sirkel (her bruker vi verdien for radiusen i stedet for diameter):

Denne definisjonen kan vi så bruke i Geogebra for å beregne pi. Vi plotter først en enhetssirkel med radius = 1. Dette gir oss en sirkel med omkrets på ca. 6,283.

Chart

Description automatically generated with medium confidence

Text

Description automatically generated with medium confidenceVi kan så sette verdien for radiusen og omkretsen til sirkelen inn i formelen og ser at vi får verdien for pi:

Pi er et irrasjonalt tall, dette betyr at det har et uendelig antall ikke-gjentagende desimaler. Dette betyr at alle verdier vi bruker for pi bare er tilnærminger siden det ikke finnes noen endelig verdi.

# Hvordan kan vi lage en tilnærming av pi?

Babylonerne kalkulerte arealet av en sirkel ved å ta , dette betyr at de estimerte pi til å være lik 3. Denne tilnærmingen fungerte sikkert fint på den tiden, men i dag vil vi gjerne har noe mer nøyaktighet. En ofte brukt tilnærming i dag er , dette gir oss 2 riktige desimaler og fungerer fint i de aller fleste praktiske tilfeller. Moderne datamaskiner har gjort det sånn at vi i dag har kommet fram til flere billioner desimaler av pi, mer enn vi noensinne kommer til å trenge. Det finnes mange ulike tilnærmingsmetoder som kan gi oss i å finne flere desimaler av pi. Under vil jeg vise fram noen av disse metodene.

## Leibniz metoden

Leibniz metoden gir oss en tilnærming av pi og består av denne uendelige alternerende rekka:

Denne rekka kan også utrykkes som den følgende summen:

Ved å skrive en Python-program kan vi se hvor mange riktige desimaler vi får ved å endre den øvre grensen i summen. Her setter vi N som den øvre grensen og itererer over hver tier potens av N. Resultatet setter vi så inn i en tabell.

Text

Description automatically generated

|  |  |  |
| --- | --- | --- |
| N verdi | Tilnærming | Antall riktige desimaler |
| N = 100 | 4 | 0 |
| N = 10­1 | 3.0418396189294032 | 0 |
| N = 102 | 3.1315929035585537 | 1 |
| N = 103 | 3.140592653839794 | 2 |
| N = 104 | 3.1414926535900345 | 3 |
| N = 105 | 3.1415826535897198 | 4 |
| N = 106 | 3.1415916535897743 | 5 |
| N = 107 | 3.1415925535897915 | 6 |
| N = 108 | 3.141592643589326 | 7 |

Ut ifra tabellen kan vi se at for hver tier potens av N får vi ett til riktig desimal av pi.

Chart

Description automatically generated

Vi kan også plotte resultatene mot hverandre. Vi ser at annenhver N verdi gir et over og under-estimat av pi som vises på den røde linjen.

## Monte Carlo metoden

En litt morsom måte å finne en tilnærmingsverdi for pi på er via det som heter Monte Carlo metoden. Denne metoden går ut på å finne forholdet mellom arealet av en sirkel med diameter på 2 og et kvadrat med sidelengde på 2. Dette forholdet vil gi oss verdien til pi. For å lage en tilnærming av dette bruker vi et Python-program. Dette programmet genererer en stor mengde punkter med tilfeldige x og y koordinater mellom 0 og 1. Vi kan så bruke Pytagoras til å finne lengden på hypotenusen som dannes av disse to punktene. Siden vi vet at lengden til alle punktene på enhetssirkelen er 1 betyr det at alle verdiene som er lavere eller lik 1 er en del av sirkelen og at alle verdiene som er høyere en 1 er en del av firkanten. Vi tar så summen av punktene inne i sirkelen og deler på det totale antallet punkter. Vi ganger så med 4 fordi vi bruker en kvart sirkel.

Under ser vi resultatet av programmet når vi bruker 106 tilfeldige punkter. De grønne prikkene representerer prikkene innenfor radiusen til sirkelen, de røde prikkene representerer prikkene utenfor radiusen.

A picture containing chart

Description automatically generated

Chart

Description automatically generated

Hvis vi plotter resultatene fra Monte Carlo metoden ser vi at vi får tilfeldige over og under-estimater, men at de gradvis nærmer seg den eksakte verdien for pi liknende som grafen for Leibniz metoden.

Sist, men ikke minst vil jeg takke min gode venn Sebastian Mandal som lot meg bruke Python programmet han hadde laget for Monte Carlo metoden. All kode brukt kan bli funnet i følgende GitHub repository: <https://github.com/simsine/pi>

# Kilder:

<https://www.scientificamerican.com/article/what-is-pi-and-how-did-it-originate/>

<https://www.countbayesie.com/blog/2015/3/3/6-amazing-trick-with-monte-carlo-simulations>

<https://www.exploratorium.edu/pi/history-of-pi>

<https://en.wikipedia.org/wiki/Leibniz_formula_for_%CF%80>