

- Määrittele Mathematicalle funktio $f(x) = x^2 - 1$. Laske arvot `f[5]` ja `f[10]`. Piirrä myös funktion kuvaaja.
- Luvun n kertoma on tulo $1 \cdot 2 \cdot 3 \cdots (n-1) \cdot n$. Sama tulohan syntyy, kun lasketaan ensin $1 \cdot 2 \cdot 3 \cdots (n-2) \cdot (n-1)$ ja kerrotaan tulos n :llä. Kertoman voi siis laskea myös rekursiion avulla. Määrittele rekursiivinen (itseään kutsuva) funktion `kertoma[n]`, joka laskee luvun n kertoman.
- Fibonaccin lukujono määritellään siten, että jonon kaksi ensimmäistä lukua ovat ykkösiä ja tämän jälkeen jonon luvut lasketaan kahden jonossa edellisenä olevan luvun summana. (Siis: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...) Määrittele nyt rekursiivinen (itseään kutsuva) funktio `Fibo[n]`, joka laskee n :nnen Fibonaccin luvun. Laske tällä 25 ensimmäistä lukua.
- Rekursio ei yleensä ole paras tapa laskea asioita, koska rekursiokierrosten lukumäärän kasvaessa laskenta-aika kasvaa eksponentiaalisesti ja koneen resurssit voivat käydä vähiin. Mathematica osaa mitata komennon suorittamiseen käytettyä aikaa `Timing`-komennolla. Mittaa paljonko aikaa määrittelemältesi `Fibo`-komennolta kuluu luvun `Fibo[25]` laskemiseen. Entäpä `Fibo[26]:n`? Kovin pitkälle ei kannata jatkaa.
- Fibonaccin lukujonon alkiot voidaan laskea myös suljetussa muodossa, jolloin laskentaan käytetty aika ei olennaisesti riipu siitä, monettako lukua ollaan laskemassa. Kokeile Mathematican omaa `Fibonacci`-komentoa ja laske sillä samat luvut kuin äsken laskit itse määrittelemällesi funktiolla. Mittaa tämän käyttämä aika. Laske `Fibonacci[100]`, `Fibonacci[1000]` ja `Fibonacci[100000]`. Rekursiolla tämä ei siis kannata.
- Vastaavasti voidaan määritellä rekursiivinen komento, jolla voi laskea Legendren polynomeja, jotka määritellään kaavalla

$$\begin{cases} p_0 &= 1 \\ p_1 &= x \\ p_n &= \frac{(2n-1)xp_{n-1} - (n-1)p_{n-2}}{n} \end{cases}$$

Laske ja piirrä samaan kuvaajaan Legendren polynomit p_6 , p_8 ja p_{10} . Tästäkin tapauksessa Mathematicalla on varastossa tavallista rekursiota tehokkaampi algoritmi. Sama onnistuu myös komennolla `LegendreP`.

- Määrittele funktio, joka pyytää käyttäjältä positiivista kokonaislukua ja tulostaa annetun luvun kertoman. Voit halutessasi lisätä funktioon tarkistuksia, onko annettu luku oikeaa muotoa.
- Ohjelmoinnista tutut `For`, `Do` ja `While`-rakenteet ovat mahdollisia myös Mathematicassa. Voit kokeilla näiden käyttöä esimerkiksi tekemällä funktion `summa[n]`, joka laskee n :n ensimmäisen positiivisen kokonaisluvun summan. Voit kokeilla samaan tarkoitukseen myös `Apply`-komentoa.