

Projekti raport - Programmeerimisharjutus

Nimi: Aleksander Viiret

Matrikli nr: B77743

E-mail: aleksander.viiret@ut.ee

Õppekava: Psühholoogia

Varasem kokku puude programmeerimise või IT-ga

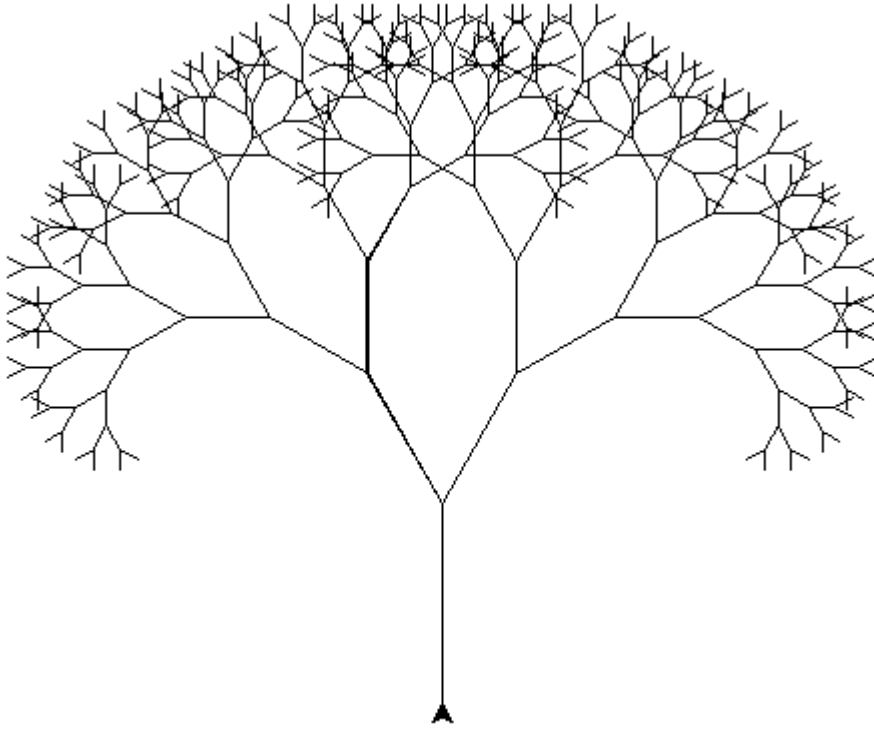
Olen terve elu hobikorras IT huviline olnud suuresti tänu sellele, et mu isa on IT spetsialist. Põgusalt õppisin Tallinna Tehnikaülikoolis arvutisüsteemide õppekaval kuid otsustasin eriala vahetada. Programmeerimisega olen kokku puutunud ka oma uurimistöö käigus kasutades Pythonit statististiliste analüüside ja masinõppe otstarbeks.

Ülesanne

Valisin ülesandeks joonistada kilpkonnagraafika abil puu kasutades rekursiooni.

Minu lahendus

Alustuseks tutvusin erinevate lahendustega selle tegemiseks ja leidsin YouTube'ist video, mis seletas lihtsat meetodit rekursiooni abil puu joonistamiseks (I know python, 2020). Selle video abil sain põhimeetodist arusaama ning kasutasin seda eeskujuks oma projekti aluse loomisel. Meetod põhineb rekursiivsel funktsioonil, mis seisneb selles, et esmalt kilpkonn joonistab funktsiooni sisendis antud pikkusega vertikaalse joone, millest saab puu tüvi. Siis pöörab kilpkonn teatud kraadide võrra vasakule ning funktsioon kutsutakse uuesti kuid seekord vähendatud joone pikkuse muutujaga. Järgmiseks pöörab kilpkonn paremale kahekordistatud kraadide võrra ning funktsiooni kutsutakse jälle vähendatud joone pikkuse muutujaga. Lõpuks pöörab kilpkonn jälle vasakule ning liigub tagurpidi. Funktsioon aga käivitub ainult, siis kui joone pikkuse muutuja on 10 või suurem. Tulemuseks on väheneva pikkusega sirgjoontest koosnev puu kujutis, mille igal joonel on omakorda kaks haru. Näidet selle meetodiga joonistatud puust on näha joonisel 1.

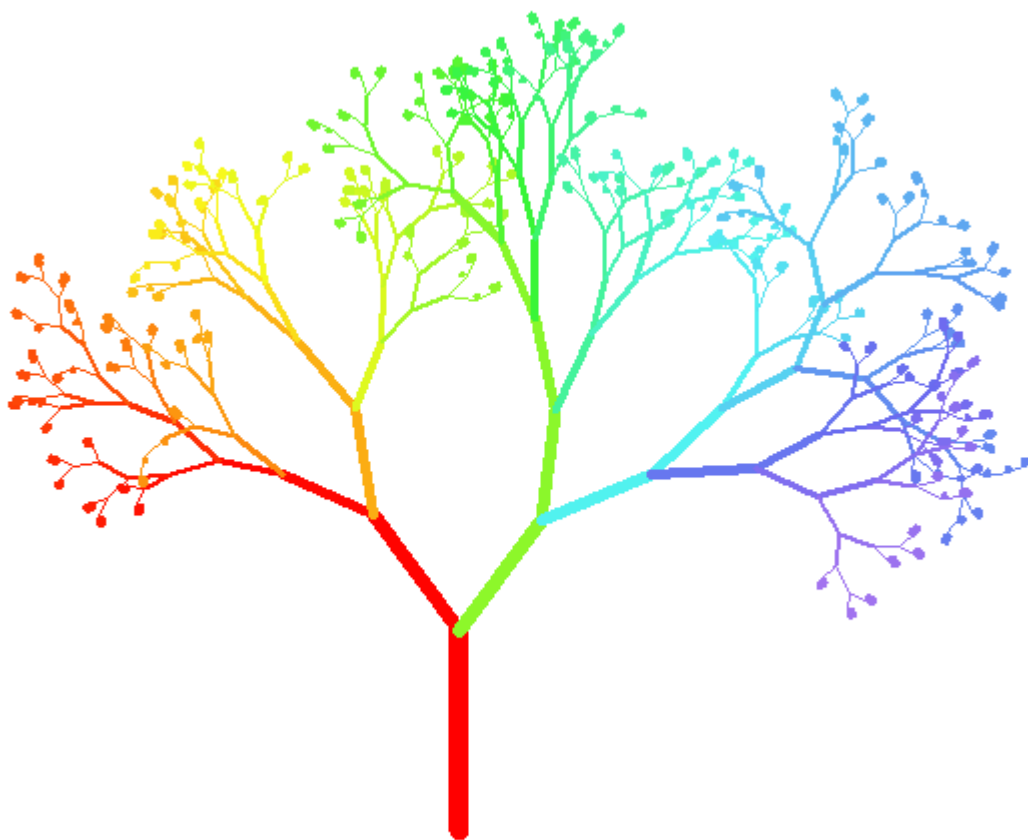


Joonis 1: Näide alusmeetodiga joonistatud puust

Järgmiseks mõtlesin kuidas asja huvitavamaks teha. Esimene mõte oli mingid muutujad juhuslikustada, et iga joonistatud puu oleks unikaalne ja rohkem orgaanilise välimusega. Selleks kasutasin random teegi funktsioone randint, et anda juhuslik täisarvuline väärtus 10 ja 40 vahel kilpkonna pöörete nurga muutujale ning uniform, et randomiseerida kordajat, mis vähendab joone pikkust ujukomaarvuliste väärtustega 0.6 ja 0.9 vahel.

Teine mõte oli teha puu värviliseks. Selleks kasutasin teeki colour, et genereerida nimekiri 512 värvikoodist hex formaadis punasest violetseni ning määrasin nimekirjast igale uuele joonele järgmise värvi. Selle tulemuseks oli, et puu oksad moodustavad vasakult paremale vikerkaarevärvilise gradiendi.

Järgmiseks sain kaks ideed GitHubist leitud rekursioonipuu lahendusest (momotofu, 2018). Esiteks määrata joone paksuseks murdosa selle pikkusest, mille tulemuseks on, et puu alumised pikemad oksad on jämedamad ning otsmised lühemad oksad on peenemad. Teiseks joonistada puu viimaste harude otsa lehed. Selleks lasin kilpkonnal joonistada täidetud ringi iga kord kui joone pikkuse muutuja muutub väiksemaks kui 10 ning uut haru antud joonele ei joonistata. Näidet valmis programmi väljundist on näha alloleval joonisel 2.



Joonis 2: Näide lõpptulemusest

Kasutatud allikad

I know python. (2020, jaanuar 30). *How to Draw Fractal or Recursion tree using python.*

<https://www.youtube.com/watch?v=RWtNQ8RQkO8>

momotofu. (2018). *Draw a tree using recursion with python.* Gist.

<https://gist.github.com/momotofu/25ae6f4e61b302f253c8a6d243fbe3dd>