

Funktiopiirturi

Java 2 Platform Standard Edition 5.0

Teppo Niinimäki

Karstulantie 8 G 115

00550 Helsinki

sähköposti: Teppo.Niinimaki@Helsinki.FI

Helsingin yliopisto

Tietojenkäsittelytieteen laitos

Ohjelmoinnin harjoitustyö (Java)

Ohjaaja Kliment Yanev

Helsingissä 24.2.2006

Sisällys

Käyttöohje.....	1
Yleistä.....	1
Asentaminen ja käynnistäminen.....	1
Laitteistovaatimukset.....	1
Toiminnot.....	1
Funktioiden syöttäminen.....	2
Virheilmoitukset.....	4
Rajoitukset.....	5
Ohjelman toiminta ja rakenne.....	6
Yleiskuvaus.....	6
Tärkeimmät toiminnot.....	6
Luokkarakenne.....	7
Ohjelman toiminta.....	7
Ohjelman rajoitukset ja niiden poistaminen.....	7
Parannusehdotuksia.....	8
Testausraportti.....	9

Liitteet

- LIITE 1: API-kuvaus (javadoc-muodossa)
- LIITE 2: Määrittelydokumentti
- LIITE 3: Luokkakaavio erikseen
- LIITE 4: Sekvenssikaaviot
- LIITE 5: Testaamiseen käytetyt metodit ja tulosteet testiajoista
- LIITE 6: Tuntikirjanpitolista
- LIITE 7: Käyttöohje HTML-muodossa

Käyttöohje

Yleistä

Funktiopiirturi on muotoa $y = f(x)$ olevien yhden muuttujan funktioiden kuvaajien piirtämiseen tarkoitettu ohjelma. Ohjelmaa voi käyttää yleiskuvan saamiseksi tarkasteltavan funktion kulusta. Lisäksi ohjelmalla pystyy vertaamaan yleisellä tasolla kahden tai useamman eri funktion kuvaajia toisiinsa. Funktiopiirturi sopiikin hyvin yksinkertaiseksi apuvälineeksi esimerkiksi opiskelijalle matemaattisten tehtävien tekemiseen, missä on apua erilaisten funktioiden luonteen selvittämisestä.

Asentaminen ja käynnistäminen

Kopioi Funktiopiirturi.jar-paketti haluamaasi hakemistoon, mitään asennusta ei tarvita. Ohjelma käynnistyy komennolla `java -jar Funktiopiirturi.jar` (Windowsissa voit myös käyttää komentoa `javaw -jar Funktiopiirturi.jar`, jolloin komentorivi-ikkuna ei aukea taustalle). Riippuen käyttöympäristöstä ohjelma saattaa käynnistyä myös kaksoisklikkaamalla Funktiopiirturi.jar-paketin kuvaketta.

Laitteistovaatimukset

Toimiakseen Funktiopiirturi vaatii, että koneella on asennettuna Java Runtime Environment 5.0 (todellinen versionumero 1.5) tai uudempi.

Toiminnot

Kuvaajan piirtäminen

Kirjoita haluamasi funktion oikea puoli funktion syöttökenttään ja paina Piirrä-nappia. Jos funktiossa ei ole virheitä, sen kuvaaja piirtyy koordinaatistoon.

Kuvan suurentaminen ja pienentäminen

Näkymää saa zoomattua oikeasta laidasta löytyvillä "+" ja "-" -napeilla. Jos "Säilytä kuvasuhde" on valittuna, pysyy horisontaalisen ja vertikaalisen suurennoksen suhde koko ajan samana. Tällöin ei

ole väliä käyttäkö x vai y -kohdan suurennosnappeja. Poistamalla valinnan kohdasta "Säilytä kuvasuhde", voi suurennosta säätää erikseen x ja y -akselien suhteen.

Alkuperäisen suurennoksen ja kuvasuhteen palauttaminen

Alkuperäiseen näkymään palaaminen tapahtuu Normalisoi näkymä -napista. Tällöin sekä horisontaalinen että vertikaalinen suurennos palautetaan 1.0 :an.

Kuvan tyhjentäminen

Koordinaatiston saa tyhjennettyä funktioista Tyhjennä-napista.

Syötettyjen funktioiden selaaminen

Jo syötettyjä funktioita voi selata syöttökentässä Ylös ja Alas -nuolinäppäimillä.

Ohjelman lopettaminen

Ohjelman saa lopetettua oikean yläkulman lopetuspainikkeesta (ruksi).

Funktioiden syöttäminen

Yhtälöistä syötetään vain oikea puoli. Esimerkiksi yhtälö $y=3x^2+1$ syötettäisiin tekstikenttään muodossa $3*x^2+1$. Vakiokertoimen ja muuttujan kirjoittaminen yhteen ei ole sallittua kuten edellä olevasta esimerkistä näkyy, vaan kertolaskua merkitään aina *-merkillä (tähti). Sulkuja voi käyttää laskujärjestyksen valitsemiseen. Funktiot kirjoitetaan muodossa **funktio parametri** tai muodossa **funktio (parametri)**. Välilyönnit ja muut tyhjät merkit operaatioiden ja operandien välissä ovat sallittuja.

Tuetut operaatiot ja niiden merkitseminen

- Yhteenlasku: +
- Vähennyslasku: -
- Kertolasku: *

- Jakolasku: /
- Potenssi: ^ (vaihtoehtoinen merkitsemistapa: #) (huom. Javan Linux-versiossa 1.5 ei toimi hattumerkki (^))

Tuetut funktiot

- Trigonometriset funktiot, niiden arkusfunktioit sekä hyperboliset funktiot: **sin**, **cos**, **tan**, **asin**, **acos**, **atan**, **sinh**, **cosh**, **tanh**
- Juuret: **sqrt** (neliöjuuri), **cbrt** (kuutiojuuri)
- Logaritmit: **ln** (luonnollinen logaritmi), **lg** (10-kantainen logaritmi)
- Eksponenttifunktio (e^x): **exp**
- Itseisarvo: **abs**

Tuetut symbolit

- Muuttujat: **x** (muuttuja, jonka suhteen kuvaaja piirretään)
- Vakiot: **pi**, **e** (Eulerin vakio)

Lukujen syöttäminen

Luvut syötetään 10-kantaisessa desimaalimuodossa. Desimaalierottimena käytetään pistettä. Desimaalipistettä edeltävän nollan voi jättää pois. Luvut on mahdollista kirjoittaa myös tieteellisessä 10-potenssimuodossa, käyttämällä isoa E-kirjainta merkitsemään eksponenttiosan alkua. Esim: **31**, **- .0785**, **1.22E-2**

Laskujärjestys

Laskujärjestys määräytyy operaatioiden prioriteetin mukaan. Seuraavassa on lueteltu operaatiot prioriteettiensa mukaisesti laskevassa järjestyksessä. Listassa ylempänä olevat siis suoritetaan alempia ennen.

1. sulut
2. potenssi (^)
3. funktiot (**sin**, ...), negaatio (–)
4. tulo (*), osamäärä (/)
5. summa (+), erotus (–)

Virheilmoitukset

Tässä osiossa esitellään ohjelman virheilmoitukset ja niiden selitykset.

"Sulut eivät ole tasapainossa. ..."

Syötetyssä lausekkeessa joko puuttuu yksi tai useampi lopetussulku tai niitä on ylimääräisiä. Korjaa sulut tasapainoon.

"Määrittelemätön symboli: <symboli>"

Ohjelma ei tunnista tätä symbolia. Tietoa tuetuista symboleista löytyy kohdasta Funktioiden syöttäminen.

"Tuntematon operaatio: <operaatio>"

Tässä kohtaa tulisi olla jokin binäärinen operaatio, ja syötettyä symbolia ei tunnisteta sellaiseksi. Lisää tietoa tuetuista operaatioista löytyy kohdasta Funktioiden syöttäminen

"Virheellinen lukuformaatti: <luku>"

Luku on syötetty virheellisesti. Syötä luku sallittua esitystapaa käyttäen. Lisää tietoa sallituista lukujen esitystavoista löytyy kohdasta Funktioiden syöttäminen

"Tunnistamaton funktio: <funktion nimi>"

Jos saat tämän virheilmoituksen, on ohjelmassa jokin sisäinen virhe. Mikäli haluat auttaa ohjelman jatkokehityksessä, ota yhteyttä ohjelman valmistajaan ja kerro, miten sait tämän virheen aikaiseksi.

Rajoitukset

Syötehistoria

Ohjelma muistaa enintään 100 viimeksi syötetty kaavaa. Suorituskertojen välillä ei säilytetä mitään tietoja, joten edellisen suorituskerran kaavahistoriaa ei voi selata.

Kuvaajien oikeellisuus

Ohjelma ei takaa piirtämiensä kuvaajien oikeellisuutta. Funktioiden epäjatkuvuuskohdissa sekä kohdissa, joissa funktion arvo muuttuu paljon pienellä välimatkalla (suuri kulmakerroin), kuvaaja saattaa piirtyä väärin.

Ohjelman toiminta ja rakenne

Yleiskuvaus

Funktiopiirturi on tarkoitettu muotoa $y=f(x)$ olevien matemaattisten funktioiden kuvaajien piirtämiseen. Funktiopiirturia ei ole tarkoitettu tarkkaan ongelmanratkaisuun, kuten nollakohtien etsimiseen. Kuvaajia voi olla ruudulla useita samaan aikaan, ja tyhjentäminen tapahtuu erillisestä napista. Kuvaajia on myös mahdollista tarkastella eri suurennoskertoimilla. Ohjelma ei säilytä mitään tietoja suorituskertojen välillä.

Tärkeimmät toiminnot

Syötetyn funktion jäsentäminen (parserointi)

Käyttäjän tekstimuodossa syöttämästä merkkijonosta luodaan FuncXtoY-luokan ilmentymä. FuncXtoY-luokan konstruktori muodostaa parametrina saamansa funktiota esittävän String-olion perusteella rekursiivisen Operation-oliorakenteen. Tämä tapahtuu pilkkomalla merkkijono ensin alkeisosasiin. Alkeisosaset sisältävä lista välitetään parserimetodille, joka listaa läpikäymällä kokoa laskusäännöt huomioon ottaen vastaavan rekursiivisen rakenteen. Esimerkiksi kaavassa olevat luvut korvataan rakenteessa Constant-olioilla, joille asetetaan vastaava arvo. Plus-merkkiä taas vastaa Add-olio, josta on viitteet summattaviin arvoihin, jotka myös ovat Operation-luokan ilmentymiä.

Funktioiden piirtäminen

SurfaceXY-luokan paint-metodia kutsuttaessa piirretään taustalle koordinaatisto sekä kyseisen pinnan sisältämät funktiot tämän päälle. Koordinaatistoon piirretään sopivin välein ruudukko, jonka viivojen etäisyys origosta merkitään kuvaan. Funktio piirretään yksi pikselipystyriivi kerrallaan vasemmalta oikealle siten, että ensin lasketaan funktion arvo pikselisarakkeen vasemmassa ja oikeassa reunassa. Nämä arvot skaalataan kuvaruudulle ja pystyviiva väliin jäävälle alueelle. Menetelmä ei anna oikeaa lopputulosta kaikissa tapauksissa; Erityisesti epäjatkuvuuskohdat piirtyvät väärin. Ongelman minimoimiseksi tarpeeksi isot pystysuuntaiset hyppäykset jätetään piirtämättä.

Syötettävien funktioiden rajoitukset:

Funktiot syötetään normaalissa matemaattisessa muodossa (esim. $2 \cdot x^3 + \sin x - 0.2$).

Laskujärjestys on normaali (korkeimmasta alhaisimpaa prioriteettiin): sulut, potenssi, funktiot ja negaatio, tulo ja osamäärä, summa ja erotus. Käytettävissä ovat seuraavat funktiot: **sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, sqrt, cbrt, exp, lg, ln, abs**. Myös literaalivakioita **pi** ja **e** voi käyttää. Muuttujaa, jonka suhteen kuvaaja piirretään merkitään **x**-kirjaimella.

Lukuvakiot syötetään 10-kantaisessa desimaalimuodossa. Desimaalierottimenä käytetään pistettä, ja sitä edeltävän nollan saa jättää pois halutessaan (esim. **.034**). Luvut voi syöttää myös tieteellisellä esitystavalla 10-potenssimuodossa käyttäen isoa E-kirjainta erottamaan mantissa ja eksponentti (esim. **2.01E-5**).

Luokkarakenne

Funktiopiirturin luokat on jaettu yleisen periaatteen mukaisesti käyttöliittymä- ja toimintalogiikkaluokkiin. Tästä säännöstä on kuitenkin muutama pieni poikkeama; Vaikka FuncXtoY-luokka lukeutuukin selvästi sisäisen toiminnan luokaksi, toimii se eräänlaisena välittäjäluokkana muiden toiminta- ja käyttöliittymäluokkien välillä. SufaceXY taas ei ole puhdas käyttöliittymäluokka, sillä se sisältää funktioiden ja koordinaatiston piirtämiseen tarvittavan logiikan. Ohjelman luokkakaavio löytyy LIITE 3 :stä.

Ohjelman toiminta

Tässä on keskitytty kahteen tärkeimpään osa-alueeseen ohjelman toiminnassa: käyttäjän syöttämän funktion jäsentämiseen ohjelman ymmärtämään muotoon sekä funktioiden kuvaajien piirtämiseen näytölle. Kummankin tapauksen sekvenssikaaviot löytyvät LIITE 4 :stä.

Ohjelman rajoitukset ja niiden poistaminen

Funktiopiirturista on hankala löytää montaa keinotekoisia rajoitusta. Yksi tällainen on kylläkin funktioiden syöttöhistorian pituuden rajoittaminen sataan kappaleeseen. Tämä johtuu siitä, että muistin jatkuva täyttäminen ilman, että mitään vapautetaan ei ole tarpeellista eikä pidemmästä historiasta ole mielestäni oleellisesti hyötyä. Tietenkin ohjelmaan voitaisiin tehdä mahdollisuus asettaa käytetty historian enimmäispituus.

Parannusehdotuksia

Parannusehdotuksia Funktiopiirturiin on erittäin helppo keksiä. Tärkeimmät lisäominaisuudet löytyvätkin määrittelydokumentin toteuttamattomista toiminnoista. Näihin lukeutuvat mm. mahdollisuus siirtyä katselemaan eri kohtia koordinaatistossa, tuki erikoisella tavalla merkittävälle operaatioille kuten kertomalle ja itseisarvolle, tuki lisäfunktioille sekä useamman parametrin funktioille ja automaattinen funktioiden nollakohtien ja leikkauskohtien merkintä. Muita hieman hankalammin toteutettavissa olevia parannuksia olisivat ainakin epäjatkuvuuskohtien oikeanlainen piirtäminen (joka vaatii niiden löytämistä), tuki $x:n$ funktioille $y:n$ suhteen sekä polaarikoordinaateilla piirrettäville funktioille, mahdollisuus poistaa kuvaajia yksitellen ja erilaisten ohjelmaan liittyvien asetusten säätömahdollisuus.

Testausraportti

Testausta suoritettiin alkuvaiheessa koodauksen edetessä luokittain yksikkötestauksena ja loppua kohden siirryttiin kaiken kattavaan systeemitestaukseen. Luokkien toiminnan ja yhteispelin lisäksi testattiin käyttöliittymän toimivuutta sekä yhteistoimintaa ydinohjelman kanssa.

Aluksi keskityttiin testaamaan kaavan muuntamista merkkijonosta rekursiiviseksi Operation-rakenteeksi. Ensin varmistettiin Operation-luokan sekä sen aliluokkien oikea toiminta. Lisäksi testattiin FuncXtoY-luokan split-metodia (kaavan pilkkoojaa) erilaisilla syötteillä. Ongelmakohtia löytyi odotetusti aluksi tieteellisen esitystavan lukujen tunnistamisessa yhdeksi kokonaisuudeksi. Seuraavaksi siirryttiin saman luokan constructor-metodin testaamiseen. Tässä käytettiin hyväksi Operation-luokan aliluokkiin ohjelmoituja toString-metodeja. Erityistä huomiota kiinnitettiin oikean laskujärjestyksen pysymiseen suluttomissa, operaatioita yhdistelevissä lausekkeissa. Lopuksi testattiin vielä koko FuncXtoY-luokan toimintaa erilaisilla lausekkeilla ja x:n alkuarvoilla. Funktion rakentamisen toimivuuden lisäksi kokeiltiin virheellisten syötteiden oikeaa havainnointia.

Lausekkeen parseroimisen ja arvon laskemisen lisäksi testattiin funktion piirtymistä oikein kuvaruudulle. Tässä olikin aluksi ongelmia saada kuvaaja piirtymään pikselintarkasti oikein. Lisäksi havaittiin jo tiedossa ollut ja odotettu puute kuvaajien piirtymisessä; Funktion epäjatkuvuuskohdissa kuvaajat saattavat piirtyä väärin, jos sellaisessa tapahtuu hyppäys. Tämän ilmiön voi havaita esimerkiksi funktiolla $y=(x-1)/abs(x-1)$. Ongelmaan ei ole löytynyt yksinkertaista ratkaisua. Epäjatkuvuuskohtien lisäksi kiinnitettiin huomiota funktioihin, jotka eivät ole määritelty kaikkialla. Funktioiden piirtymisen lisäksi testattiin koordinaatoruudukon ja siihen kuuluvien merkintöjen piirtyminen oikein. Funktioiden ja koordinaatiston piirtymistä testattiin myös eri suurennoskertoimilla.

Käyttöliittymää ja koko ohjelman toimintaa testattiin lähinnä käyttämällä ohjelmaa ja yrittämällä saada aikaan erilaisia epätavallisia tilanteita. Ohjelmaikkunan kokoa kokeiltiin muuttaa funktionäkymää zoomailtiin kumpaankin suuntaan. Syötettyjen funktioiden historiaa selailtiin molempiin suuntiin mahdollisimman kauas. Virheilmoituksia testattiin virheellisillä syötteillä. Lisäksi ohjelmaa testattiin sekä Windows- (XP) että Linux- (Ubuntu, Fedora) ympäristöissä. (Tässä yhteydessä löydettiinkin Java 1.5 :n Linux-versiosta bugi, joka estää mm. hattumerkin syöttämisen JTextField-tekstikenttään. Ongelma on ilmeisesti korjattu versiossa 1.6.)