Kemistin data-analysaattori
Testausdokumentti (1. iteraatio)

Taisto Gerdt
Timo Kääriäinen
Mika Miettinen
Pasi Rastas
Jari Tuominiemi

Ohjelmistotuotantoprojekti
Helsingin yliopisto
Tietojenkäsittelytieteen laitos
30. lokakuuta 2002

Versiohistoria:

Päiväys	Muutokset	Muutosten tekijä
28.10.2002	Alustava versio	Mika Miettinen
29.10.2002	Pieniä muutoksia	Mika Miettinen
30.10.2002	Ei muutoksia	Mika Miettinen

Sisältö

1 Johdanto **1**
1.1 Testauksen tarkoitus 1
1.2 Testausmenetelmät 1

2 Testitulokset **1**
2.1 Syntaksin tarkastus 1
2.1.1 parseCarbons 1
2.1.2 parseFormula 2
2.1.3 parseInput 2
2.1.4 parseFDB 2
2.2 Syöte- ja tulostiedostojen käsittely 3
2.2.1 readFile 3
2.3 Tietokannan luku ja päivitys 3
2.3.1 checkFDB 3

3 Testiaineistot **4**
3.1 readFile 4
3.2 parseCarbons 4
3.3 parseFormula 5
3.4 parseInput 5
3.5 parseFDB 7
3.6 checkFDB 7

1 Johdanto

1.1 Testauksen tarkoitus

Koska oikeita massaspektrometrin tuottamia syöteaineistoja ei ole ryhmän käytettävissä, pyritään tällä testauskierroksella vakuuttumaan lähinnä laskentaosien käyttämien muiden moduulien ja apufunktioiden oikeellisuudesta ja virheettömyydestä.

1.2 Testausmenetelmät

Kaikki funktiot testattiin yksittäin yksinkertaisten ajuriohjelmien avulla. Näiden ohjelmien tulostuksesta havaittiin mahdolliset virheet.

2 Testitulokset

2.1 Syntaksin tarkastus

2.1.1 parseCarbons

Yleistä: Funktiota käytetään tietokannan tietojen syntaksin tarkastuksessa. Testauksessa on tarkoitus varmistua, että funktio hyväksyy vain oikeanmuotoiset hiilijonot ja tulkitsee ne oikein.

Testiaineisto: Testiaineisto koostui joukosta funktiolle parametrina annettuja merkkijonoja. Nämä merkkijonot testaavat kaikki hiilijonon tunnistamiseen käytettävän äärellisen automaatin siirtymät.

Tulokset: Yksi merkkijonoista paljasti funktion toteuksessa virheen, joka korjattiin. Korjauksen jälkeen funktio jäsensi ja tulkitsi kaikki merkkijonot oikein.

2.1.2 parseFormula

Yleistä: Funktio on käytössä tietokannan tietojen syntaksin tarkastuksessa. Testauksella on tarkoitus varmistaa että funktio hyväksyy vain lailliset molekyylikaavat ja muodostaa niiden sisäisen esitysmuodon oikein.

Testiaineisto: Testiaineisto koostui joukosta merkkijonoja, jotka yksitellen annettiin funktiolle parametrina. Merkkijonot testaavat kaikki siirtymät äärellisestä automaatista, joka on funktion toteutuksena.

Tulokset: Yksi merkkijono virheellisesti hyväksyttiin ja yksi tulkittiin väärin. Virheiden korjauksien jälkeen funktio jäsensi ja tulkitsi kaikki merkkijonot oikein.

2.1.3 parseInput

Yleistä: Funktiota käytetään syötetiedoston syntaksin tarkastamiseen ja tiedoston muuttamiseen sisäiseen esitysmuotoon. Testauksella on tarkoitus varmistaa, että funktio hyväksyy vain oikeanlaiset syötteet.

Testiaineisto: Testiaineistona on viisi tiedostoa, joista neljä on eri tavalla virheellisiä ja viides täysin oikeellinen.

Tulokset: Testauksessa ei havaittu virheitä.

2.1.4 parseFDB

Yleistä: Funktio tarkastaa halutun fragmenttitietokannan tiedoston syntaksin oikeellisuuden ja muuttaa kyseisen tiedoston sisäiseen esitysmuotoon. Testauksella varmistetaan, että funktio hyväksyy vain syntaksiltaan oikeelliset tiedostot ja raportoi oikein sen miten tiedosto oli virheellinen.

Testiaineisto: Testiaineistona oli kahdeksan tiedostoa, joista seitsemän olivat syntaksiltaan virheellisiä ja kahdeksas täysin oikeellinen.

Tulokset: Testauksessa ei havaittu virheitä.

2.2 Syöte- ja tulostiedostojen käsittely

2.2.1 readFile

Yleistä: Funktio lukee tiedoston, jonka nimi annetaan parametrina ja lukee tiedoston matriisiin. Testauksella pyritään varmistumaan että funktio käsittelee oikein erilaisia tiedostoja.

Testiaineisto: Aineisto koostui kolmesta erityyppisestä tiedostosta, jotka testaavat että funktio käsittelee oikein kaikki ne tapaukset, jotka ovat käytännössä mahdollisia.

Testitulokset: Ei havaittuja virheitä.

2.3 Tietokannan luku ja päivitys

2.3.1 checkFDB

Yleistä: Funktio etsii `parseFDB`-funktion tuottamasta globaalista taulukosta annettua äitimolekyyliä ja sen tytärioneja vastaavat tiedot ja palauttaa ne taulukkona kutsujalle. Testauksella varmistetaan, että funktio löytää todellakin kaikki relevantit tiedot.

Syöte	Odotettu tulos	Saatu tulos
`/dev/null`	tyhjä taulukko	tyhjä taulukko
tiedosto, jota ei ole olemassa	-1	-1
tekstitiedosto	merkkijonomatriisi	merkkijonomatriisi

Taulukko 1: readFile-funktion testitulokset

Testiaineisto: Funktion yksinkertaisuuden vuoksi testaukseen käytetään vain kahta parametria: yksi on sellaisen äitimolekyylin massa, joka löytyy tietokannasta, toinen sellainen jonka massaista äitimolekyyliä ei taulukosta löydy.

Testitulokset: Testauksessa ei havaittu virheitä.

3 Testiaineistot

Tässä luvussa esitellään kaikki testiainestot joita funktioiden testauksessa käytettiin.

3.1 readFile

Testiaineisto koostui kolmesta erilaisesta tiedostosta. Nämä tiedostot ja testien tulokset esitetään taulukossa 1.

3.2 parseCarbons

Testiaineistona oli joukko merkkijonoja, jotka testasivat kaikki funktion toteutuksessa käytetyn äärellisen automaatin siirtymät. Nämä merkkijonot ja testauksen tulokset käyvät ilmi taulukosta 2 (hyväksyvän lopputuloksen yhteydessä on lisäksi huomautettu tulkittiinko merkkijonon esittämä hiilijono oiken).

Syöte	Odotettu tulos	Saatu tulos
`C1`	hyväksytty	hyväksytty (oikein)
`C1C2`	hyväksytty	hyväksytty (oikein)
`C1C2C`	hylätty	hylätty
`C10`	hyväksytty	hyväksytty (oikein)
`C90`	hylätty	hylätty
`CH`	hylätty	hylätty
`C2H`	hylätty	hylätty
`C11H`	hylätty	hyväksytty
`C0`	hylätty	hylätty
`H`	hylätty	hylätty

Taulukko 2: parseCarbons-funktion testitulokset

3.3 parseFormula

Testiaineisto koostui joukosta merkkijonoja, jotka tastasivat kaikki funktion toteutuksessa käytetyt äärellisen automaatin siirtymät. Merkkijonot ja tulokset käyvät ilmi taulukosta 3 (hyväksyvän lopputuloksen yhteydessä on lisäksi huomautettu tulkittiinko kyseisen merkkijonon esittämä atomikaava oikein).

3.4 parseInput

Testiaineistona oli viisi tekstitiedostoa, joista yksi oli täysin syntaksin mukainen syötetiedosto ja loput neljä virheellisiä. Näissä tiedostoissa oli kussakin yksi rivi, jossa yksi kolmesta mahdollisesta sarakkeesta oli väärin. Lisäksi yhdessä tiedostossa kaikki rivit olivat virheellisiä. Testitiedostot ja testauksen tulokset näkyvät taulukosta 4.

Syöte	Odotettu tulos	Saatu tulos
`C`	hyväksytty	hyväksytty (oikein)
`N2`	hyväksytty	hyväksytty (oikein)
`S`	hyväksytty	hyväksytty (oikein)
`S1`	hyväksytty	hyväksytty (oikein)
`Si`	hyväksytty	hyväksytty (oikein)
`Si1`	hyväksytty	hyväksytty (oikein)
`SO2`	hyväksytty	hyväksytty (väärin)
`HS`	hyväksytty	hyväksytty (oikein)
`CH3N`	hyväksytty	hyväksytty (oikein)
`NH`	hyväksytty	hyväksytty (oikein)
`C60`	hyväksytty	hyväksytty (oikein)
`CK`	hylätty	hyväksytty
`CH3CNH2HCOOH`	hyväksytty	hyväksytty (oikein)
`C-H4`	hylätty	hylätty

Taulukko 3: parseFormula-funktion testitulokset

Syöte	Odotettu tulos	Saatu tulos
rivin 1. sarake virheellinen	hylätty	hylätty
rivin 2. sarake virheellinen	hylätty	hylätty
rivin 3. sarake virheellinen	hylätty	hylätty
kaikki rivit virheellisiä	hylätty	hylätty
oikeellinen tiedosto	hyväksytty	hyväksytty

Taulukko 4: parseInput-funktion testitulokset

Syöte	Odotettu tulos	Saatu tulos
rivin 1. sarake virheellinen	hylätty	hylätty
rivin 2. sarake virheellinen	hylätty	hylätty
rivin 3. sarake virheellinen	hylätty	hylätty
rivin 4. sarake virheellinen	hylätty	hylätty
rivin 5. sarake virheellinen	hylätty	hylätty
rivin 6. sarake virheellinen	hylätty	hylätty
kaikki rivit virheellisiä	hylätty	hylätty
oikeellinen tiedosto	hyväksytty	hyväksytty

Taulukko 5: parseFDB-funktion testitulokset

Syöte	Odotettu tulos	Saatu tulos
`200`	ei löydettyjä tietoja	ei löydettyjä tietoja
`90`	kaikki relevantit löydetty	kaikki relevantit löydetty

Taulukko 6: checkFDB-funktion testitulokset

3.5 parseFDB

Testiaineisto koostui kahdeksasta tiedostosta, joista yksi oli syntaksiltaan täysin oikeellinen ja loput seitsemän virheellisiä. Virheelliset tiedostot sisälsivät kukin yhden rivin, jossa yksi kuudesta sarakkeesta oli väärin. Lisäksi yhdessä tiedostossa kaikki rivit olivat virheellisiä. Tiedostot ja testauksen tulokset käyvät ilmi taulukosta 5.

3.6 checkFDB

Testiaineisto koostui kahdesta funktiolle parametreina annetusta luvusta. Ensimmäinen luku on keksitty eikä esitä minkään äitimolekyylin massaa. Toinen luku on olemassaolevan äitimolekyylin massa. Nämä luvut ja testin tulokset näkyvät taulukossa 6.