Helsingin Yliopisto – Tietojenkäsittelytieteen laitos 02.09.12
Tietorakenteiden harjoitustyö loppukesä 2012 / Ohjaaja: Kristiina Paloheimo
Kati Kyllönen Opiskelijanumero: 11539913

Tira harjoitustyö – Sanaindeksointi – Testausdokumentti

Sisällysluettelo

1 Tira Harjoitustyö

1.1 Aihe: Sanaindeksointi

Sanaindeksointiohjelma etsii syötteenä saamistaan tekstitiedostoista haettavan merkkijonon esiintymistiheyden, sekä esiintymiskohdat kussakin tiedostossa. Haun suoritettuaan ohjelma tulostaa kaikki tiedostojen ne rivit, joilla haettavat merkkijonot esiintyvät.

Harjoitustyön tarkoituksena on haun toteuttamisen lisäksi tarkkailla ja dokumentoida Trie-hakupuun toteutusratkaisujen välisiä suorituseroja.

Suorituserojen tarkkailun on työn edetessä todettu olevan triviaalia, sillä eroavaisuudet valittujen toteutusratkaisujen välillä ovat olemattomia niillä syötteillä, jotka toteutusympäristön resurssit mahdollistavat. NetBeans toteutusympäristössä on mahdollista lisätä käytettävän muistin määrää projektin Properties-valikon kautta asettamalla Run categoriaan VM Options kenttään käytettävän keskusmuistin maksimimäärää osoittava parametri esim -Xmx512m. Tämän avulla maksimirivimäärä, joka ohjelmaan voitiin ladata jäi kuitenkin vain 7000 riviin.

2 Testaus

2.1 Käyttöliittymän testaus

Ohjelman tekstipohjaisen käyttöliittymän toimivuutta on testattu manuaalisesti ohjelman kehitystyön edetessä. Käyttöliittymä kysyy käyttäjältä ladattavan tiedoston hakemistopolkua ja tiedoston nimeä. Kun käyttöliittymälle syöttää tyhjän merkkijonon ohjelma siirtyy kysymään hakusanaa.

Tiedoston nimet:

Käyttöliittymää on testattu syöttämällä sekä olemassaolevia hakemistopollkuja ja tiedoston nimiä, sekä sellaisia joita ei ole olemassa. Myös erilaisia tiedostopäätteitä on annettu syöteenä. Jos ohjelmalle antaa syötteenä tiedoston joka on olemassa ohjelma lukee tiedoston ja lataa sen Trie hakupuuhun. Jos ohjelmalle antaa syötteenä tiedoston tai hakemistopolun tai tiedostopäätteen, jota ei ole olemassa ohjelma antaa tästä virheilmoituksen ja kysyy uutta tiedostonnimeä.

Hakusanat:

Käyttöliittymää on testattu syöttämällä sekä tiedostosta löytyviä sanoja, että sanoja jotka täsmäävät osittain tiedostossa löytyviin sanoihin, sekä sanoja jotka eivät löydy tiedostosta.

Ohjelma tulostaa tiedon mistä tiedostosta haettu sana löytyy, sekä ne rivit kokonaisuudessaan, rivinumeroineen, miltä sana löytyy

Testiympäristönä on ollut Windows 7 ja Netbeans 7.2, sekä Tietojenkäsittelytieteen laitoksen

Ubuntu ympäristö.

2.2 Tiedostorutiinien testaus

Tiedoston lukemista on testattu käyttöliittymätestauksen yhteydessä. Tiedostojen sisällöt ovat olleet mm seuraavanlaisia:

Windows ympäristöön luotuja testitiedostoja

- testirivit.txt 6 riviä tekstiä jokainen erilainen koostuen kirjaimista ja numeroista
- testirivit2.txt 7 riviä tekstiä, 2 riveistä sisältävät saman tekstin
- testirivit3.txt 1 rivi tekstiä
- testirivit4.txt 11 samankaltaista riviä
- testirivit5.txt 49 samankaltaista riviä
- testirivit6.txt tyhjä tiedosto
- testirivit7.txt 6 riviä alkaen numeroilla
- kalevala16.txt kalevalan runo 16
- Eng_pg17269.txt Gutenbergista ladattu englanninkielinen kirja

Unix ympäristöön luotuja testitiedostoja ovat L-alkuiset tiedostot, kuten

- LKalevala16.txt kalevalan runo 16
- Leng_pg17269.txt Gutenbergista ladattu engalnninkielinen kirja

Käytetyt tiedostot löytyvät projektihakemistosta kansiosta testitiedostot

2.3 Yksikkötestaus

Yksittäisten rutiinien toimivuutta on testattu JUNIT-testitapausten avulla

2.4 Integrointitestaus

Metodien yhteentoimivuutta on testattu manuaalisesti, käyttäen apuna testitulostuksia.

Apuna on käytetty testitulostuksia, jotka löytyvät koodista kommentoituna, joten testien uusiminen on mahdollista, joskin työlästä.

2.5 Aika- ja tilavaativuuksien testaus

Ohjelmiston aika- ja tilavaativuuksien testaus. Alkuperäistä tavoitetta vertailla eri toteutusvaihtoehtojen suorituskykyä ei nähty järkeväksi toteuttaa, sillä mittayksiköllä

millisekunti (sekunnin tuhannesosa) ei ollut nähtävissä eroja eri hakujen välillä. Hakuaika osoitti kaikissa tapauksissa 0 millisekuntia. Kun mittayksiköksi vaihdettiin mikrosekunti (millisekunnin tuhannesosa) oli pieniä eroavaisuuksia havaittavissa, tosin testien toistamisen yhteydessä nämä eroavaisuudet eivät olleet toistettavissa. Ilmeisimmin eroavaisuudet kumpusivat käyttöjärjestelmän toiminnasta taustalla.

Kaikilla toteuteuilla vaihtoehdoilla on tällä hetkellä aikavaativuus O(lasten lukumäärä).

Kaikilla toteutetuilla vaihtoehdoilla on tilavaativuus O(lasten lukumäärä)

Toteutetut vaihtoehdot ovat Trie-hakupuu yksisuuntaisella linkitetyllä listalla, Trie-hakupuu yksisuuntaisella järjestetyllä linkitetyllä listalla, Trie hakupuu dynaamisen taulun kanssa, taulu järjestetään lisäysjärjestysalgoritmilla. Ohjelmassa on myös lomitusjärjestelmän metodi, mutta se ei vielä tällä hetkellä täysin toimi.

3 Lähteet

Tietorakenteet kurssin 58131 kevät 2012 kurssimateriaali: Patrik Floréen
http://www.cs.helsinki.fi/u/floreen/tira2012/tira.pdf

Trie rakenteen esittely: Esa Junttila
http://www.cs.helsinki.fi/u/ejunttil/opetus/tiraharjoitus/trie.html

Algoritmit 2, Luento 6: Timo Männikkö
http://users.jyu.fi/~mannikko/algoritmit2/luennot/luento6.pdf