Tik-79.148 **Kevät 2000**
Tietojenkäsittelyteorian perusteet
Laskuharjoitus 11
Ratkaisut

1. Koska rekursiiviset funktiot saavat argumenteikseen luonnollisia lukuja, niillä ei voi suoraan käsitellä merkkijonoja. Mikäli näin kuitenkin halutaan tehdä, täytyy merkkijonot koodata numeroiksi systemaattisella tavalla. Tällaista koodausta kutsutaan *Gödel-numeroinniksi* (loogikko Kurt Gödelin 1906–1978 mukaan). On olemassa monia vaihtoehtoisia tapoja määritellä Gödel-numerointi. Kirjassa koodataan n-kirjaimisen aakkoston merkkijonot $n+1$-kantaiseksi kokonaisluvuiksi, jotka edelleen muutetaan kymmenjärjestelmän luvuiksi.

   Tehtävän aakkoston $\Delta = \{a, b, c\}$ symbolit asetetaan järjestykseen

   $$d_1 = a$$
   $$d_2 = b$$
   $$d_3 = c$$

   Gödel-numeroinnin kantaluvuksi saadaan $\beta = |\Delta| + 1 = 4$.

   Merkkijono $w = d_{i_1} d_{i_2} \ldots d_{i_k}$ muutetaan numeroksi käyttäen tavallista tapaa lukujärjestelmän vaihtamiseksi:

   $$\mathsf{gn}(w) = \beta^{k-1} \cdot i_1 + \beta^{k-2} \cdot i_2 + \cdots + \beta^1 \cdot i_{k1} + i_k$$

   Oppikirjassa määritellään lisäksi symboli d_0, joka vastaa tyhjää merkkijonoa e. Symboli on lisätty sen vuoksi, että muussa tapauksessa olisi olemassa lukuja (kuten β), joita ei vastaisi mikään merkkijono. Tyhjän merkkijonon lisääminen puolestaan aiheuttaa sen, että yhdellä merkkijonolla on monta Gödel-numeroa, jotka saadaan lisäämällä jonon symbolien väliin tyhjiä merkkejä. Tällöin merkkijonon varsinainen Gödel-numero on se, jossa ei ole yhtään tyhjää merkkiä.

   (a) Merkkijonon abc Gödel-numero löydetään seuraavasti.

   $$\begin{aligned}\mathsf{gn}(abc) &= 1 \cdot 4^2 + 2 \cdot 4^1 + 3 \cdot 4^0 \\ &= 27\end{aligned}$$

   (b) Gödel-numeroa 19 vastaava merkkijono on:

   $$19 = 1 \cdot 4^2 + 0 \cdot 4^1 + 3 \cdot 4^0$$
   $$\mathsf{gn}^{-}1(19) = d_1 d_0 d_3 = aec = ac$$

   Gödel-numeroinnin käytännön merkitys on siinä, että sen avulla voidaan määritellä funktioita, jotka käsittelevät toisia funktioita. Tällöin funktio saa argumentikseen toista funktiota vastaavan Gödel-numeron.

   Gödel itse käytti numerointiaan kuuluisassa "epätäydellisyystodistuksessaan", jossa hän osoitti, että millä tahansa riittävän ilmaisuvoimaisella[1]

[1] Formalismi on "riittävän ilmaisuvoimainen", mikäli kielellä pystytään esittämään luonnollisten lukujen teoria.

formalismilla voidaan esittää väitteitä, joiden totuusarvoa ei pystytä selvittämään. Tällä kurssilla esiintulleista formalismeista "riittävän ilmaisuvoimaisia" ovat Turingin koneet (Turingin koneen pysähtymisongelma), μ-rekursiiviset funktiot (löytääkö rajoittamaton minimointi vastauksen) sekä vapaat kieliopit (onko $w \in L(G)$).

2. Tyypin 0 kielioppi on nelikko $G = (V, \Sigma, R, S)$, missä V, Σ ja S on määritelty samoin kuin yhteydettömien kielioppien tapauksessa. Kieliopin säännöt ovat muotoa

$$R \subseteq V^*(V - \Sigma)V^* \to V^*$$

Säännön vasemmalla puolella saa olla mikä tahansa merkkijono, kunhan se vain sisältää ainakin yhden nonterminaalin.

Tehtävän ratkaisee kielioppi G:

$$\begin{aligned}
\Sigma =& \{a\} \\
V =& \Sigma \cup \{S, [,], A, N\} \\
R =& \{S \to [NA], S \to a, [N \to [NN, \\
& NA \to AAN, N] \to], [A \to a[, [] \to e\}
\end{aligned}$$

Esimerkiksi sana a^{2^3} saadaan johdettua seuraavasti:

$$\begin{aligned}
S &\to [NA] \to [NNA] \to [NNNA] \to [NNAAN] \to [NNAA] \\
&\to [NAANA] \to [NAAAAN] \to [NAAAA] \to [AANAAA] \\
&\to [AAAANAA] \to [AAAAAANA] \to [AAAAAAAAN] \\
&\to [AAAAAAAA] \to a[AAAAAAA] \to aa[AAAAAA] \\
&\to aaa[AAAAA] \to aaaa[AAAA] \to aaaaa[AAA] \\
&\to aaaaaa[AA] \to aaaaaaa[A] \to aaaaaaaa[] \to aaaaaaaa
\end{aligned}$$

Ideana on aluksi johtaa yksi A, joka toimii perustapauksena $n = 0$. Seuraavaksi johdetut N-kirjaimet osoittavat halutun kahden potenssin. Saadut N-kirjaimet kuljetetaan A-kirjainten yli siten, että kukin A korvataan aina kahdella A:lla. Lopuksi kuljetetaan sanan alkua merkinnyt hakasulku A-kirjainten yli ja korvataan ne a-kirjaimilla.

3. Tehtävän kieliopin perusajatus on samanlainen kuin edellisenkin.

$$\begin{aligned}
\Sigma =& \{a\} \\
V =& \Sigma \cup \{S, [,], F, N, A\} \\
R =& \{S \to [N], N \to FNA, N \to e \\
& FA \to AaF, Fa \to aF, F] \to], \\
& [A \to [, [a] \to a[, [] \to e\}
\end{aligned}$$

Esimerkiksi sana a^{3^2} saadaan johdettua seuraavasti:

$$
\begin{aligned}
S &\to [N] \to [FNA] \to [FFNAA] \to [FFFNAAA] \to [FFFAAA] \\
&\to [FFAaFAA] \to [FFAaAaFA] \to [FFAaAaAaF] \to [FFAaAaAa] \\
&\to [FAaFaAaAa] \to [FAaaFAaAa] \to [FAaaAaFaAa] \to [FAaaAaaFAa] \\
&\to [FAaaAaaAaFa] \to [FAaaAaaAaaF] \to [FAaaAaaAaa] \\
&\to [AaFaaAaaAaa] \to [AaaFaAaaAaa] \to [AaaaFAaaAaa] \\
&\to [AaaaAaFaaAaa] \to [AaaaAaaFaAaa] \to [AaaaAaaaFAaa] \\
&\to [AaaaAaaaAaFaa] \to [AaaaAaaaAaaFa] \to [AaaaAaaaAaaaF] \\
&\to [AaaaAaaaAaaa] \to [aaaAaaaAaaa] \to a[aaAaaaAaaa] \to aa[aAaaaAaaa] \\
&\to aaa[AaaaAaaa] \to aaa[aaaAaaa] \to aaaa[aaAaaa] \to aaaaa[aAaaa] \\
&\to aaaaaaa[Aaaa] \to aaaaaa[aaa] \to aaaaaaa[aa] \to aaaaaaaa[a] \\
&\to aaaaaaaaa[] \to aaaaaaaaa
\end{aligned}
$$

4. Ongelmaluokkaan NP kuuluvat kaikki ne ongelmat, jotka voidaan ratkaista epädeterministisellä Turingin koneella polynomisessa ajassa syötteen pituuteen. Käytännössä tämä tarkoittaa sitä, että kaikki ongelmat, joiden vastauksen oikeellisuus voidaan tarkistaa polynomisessa ajassa, ovat NP-luokassa.

   Ongelma on NP-täydellinen, mikäli kaikki NP-luokan ongelmat voidaan palauttaa siihen polynomisessa ajassa. Tässä perusajatuksena on osoittaa, että ongelma A on vähintään yhtä vaikea ratkaistavaksi kuin ongelma B. Tämä tehdään näyttämällä, miten ongelma B voidaan muuttaa ongelmaksi A. Muunnoksesta käytetään termiä *reduktio*.

   Ongelman todistaminen NP-täydelliseksi tapahtuu yleensä kahdessa vaiheessa:

   (a) Todistetaan, että ongelma kuuluu luokkaan NP.

   (b) Osoitetaan, että jokin NP-täydellinen ongelma voidaan redusoida tutkittavaan ongelmaan.

   Graafin ydin on joukko solmuja, joille pätevät seuraavat ehdot:

   (a) Ytimeen kuuluvien solmujen välillä ei ole kaaria.

   (b) Kaikkiin ytimeen kuulumattomiin solmuihin päästään yhdellä askeleella jostain ytimeen kuuluvasta solmusta.

   Graafin ytimen löytämisongelma kuuluu selvästikin luokkaan NP, koska se voidaan ratkaista epädeterministisellä Turingin koneella seuraavasti:

   (a) Arvataan solmujoukko K. Tähän kuluu $O(|K|)$ askelta.

   (b) Käydään läpi kaikki K:n solmuista lähtevät kaaret (a, b). Mikäli kaari menee johonkin toiseen K:hon kuuluvaan solmuun, hylätään vastaus. Muussa tapauksessa merkitään solmu b tarkistetuksi. Tähän kuluu $O(|E|)$ askelta.

   (c) Lopuksi tarkistetaan, että kaikki ytimeen kuulumattomat solmut ovat merkittyjä. Tämä voidaan tehdä ajassa $O(|V|)$.

   Seuraavaksi täytyy redusoida jokin NP-täydellinen ongelma peittävän joukon etsimisongelmaan. Yksi perustavinta laatua olevista NP-täydellisistä

ongelmista on niin kutsuttu 3-SAT-ongelma. Ongelmassa on annettuna joukko lauselogiikan klausuuleita, joissa kaikissa on täsmälleen kolme literaalia, ja tarkoituksena on löytää jokin tapa asettaa loogisille muuttujille totuusarvot siten, että kaikissa klausuuleissa on vähintään yksi tosi literaali.

Esimerkiksi klausuulijoukko:

$$\{\{x_1, \neg x_2, x_3\}, \{\neg x_1, \neg x_2, \neg x_3\}\}$$

on toteutuva, koska valitsemalla x_1:n ja x_2:n arvoiksi *tosi* sekä x_3:n arvoksi *epätosi*, on kummassakin klausuulissa on vähintään yksi tosi literaali. Muuttuja x_1 tekee ensimmäisen klausuulin todeksi ja muuttuja x_3 tekee toisen klausuulin todeksi ($\neg x$ on tosi täsmälleen silloin, kun x ei ole tosi). Tällaista muuttujien arvojen valintaa kutsutaan *valuaatioksi*, ja siitä käytetään merkintää $\mathcal{V} = \{x_1, x_2, \neg x_3\}$, tai $\mathcal{V}(x_1) = \mathcal{V}(x_2) = T$, $\mathcal{V}(x_3) = F$.

3-SAT-ongelma voidaan muuntaa graafin peittämisongelmaksi seuraavalla tavalla:

Olkoon 3-SAT-ongelmassa esiintyvien muuttujien joukko $x_1, \ldots, x_n$ ja klausuulien joukko $C = \{C_1, \ldots, C_m\}$ Muodostetaan graafi $G = (V, E)$ siten, että jokaista klausuulia C_i kohden verkkoon tulee yksittäinen solmu c_i ja jokaista muuttujaa x_i kohden tulee kaksi solmua: x_i ja $\neg x_i$. Graafin kaarirelaatio E määritellään seuraavasti:

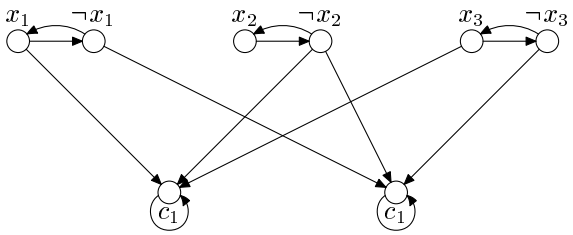
(a) Kaikilla x_i, kaaret $(x_i, \neg x_i), (\neg x_i, x_i) \in E$.

(b) Kaikilla $x_i \in C_j$, $(x_i, c_j) \in E$.

(c) Kaikilla $\neg x_i \in C_j$, $(\neg x_i, c_j) \in E$.

(d) Kaikilla c_i, $(c_i, c_i) \in E$.

Esimerkiksi ylläolevaa klausuulijoukkoa vastaava verkko on:



Graafin rakenne on suunniteltu siten, että solmu x_i kuuluu ytimeen, jos ja vain jos muuttuja x_i on tosi valuaatiossa, joka toteuttaa klausuulijoukon. Samoin solmu $\neg x_i$ kuuluu ytimeen, mikäli x_i ei ole tosi toteuttavassa valuaatiossa. Solmujen x_i ja $\neg x_i$ välinen keskinäinen poissulkevuus toteutetaan sillä, että niiden välillä on kaaret kumpaankin suuntaan, joten niiden molempien mukaanottaminen rikkoisi ytimen määritelmän. Toisaalta, jomman kumman solmuista täytyy kuulua ytimeen, sillä niihin ei tule mitään muita kaaria, eikä määritelmän toinen ehto voi muutoin toteutua. Mitään muita solmuja ei ytimeen voi kuulua, sillä kaikista solmuista c_i on kaari takaisin itseensä.

Ylläolevassa verkossa on ydin $K = \{x_1, x_2, \neg x_3\}$, joka vastaa valuaatiota $x_1 = T$, $x_2 = T$, $x_3 = F$.

Todistetaan vielä formaalisti, että graafilla on ydin tismalleen silloin, kun klausuulijoukko on toteutuva.

Mikäli graafilla G on ydin K, voidaan klausuulijoukon toteuttava valuaatio muodostaa seuraavasti: $\mathcal{V}(x_i) = T$, joss $x_i \in K$, muuten $\mathcal{V}(x_i) = F$. Koska K on ydin, on jokaista c_i:tä kohden K:ssa ainakin yksi solmu x siten, että $(x, c_i) \in E$. Graafin rakenteen perusteella $x \in C_i$, joten klausuuli C_i toteutuu. Koska sama pätee kaikille klausuuleille, toteuttaa valuaatio $\mathcal{V}$ klausuulijoukon.

Mikäli klausuulijoukko C on toteutuva, sillä on ainakin yksi valuaatio $\mathcal{V}$, joka toteuttaa sen. Muodostetaan K siten, että $x_i \in K$, jos $\mathcal{V}(x_i) = T$, ja $\neg x_i \in K$, jos $\mathcal{V}(x_i) = F$. Koska kaikille klausuuleille löytyy ainakin yksi literaali x, joka on tosi, on kaikilla solmuilla c_i graafin rakenteen perusteella vähintään yksi ytimeen kuuluva edeltäjä. Valuaatiossa on kullakin loogisella muuttujalla tismalleen yksi arvo, joten K:ssa ei voi olla kahta solmua, joiden välillä on kaari. Näin ollen K on ydin.