**Algoritmiteorian jatkokurssi, kevät 1999**
Harjoitus 3: ratkaisuja (KP, PO täyd. & tark.)

1. (a) Merkkijono $x$ kuuluu kieleen EQUIV täsmälleen silloin, kun kaava

$$(F \wedge G) \vee (\neg F \wedge \neg G)$$

on tautologia eli kun kaava

$$(F \vee G) \wedge (\neg F \vee \neg G)$$

ei ole toteutuva.
Määritellään muunnos $f$ asettamalla

$$f(x) = \begin{cases} 0, \text{ jos } x \neq \langle F, G \rangle \text{ kaikilla kaavoilla } F,\ G, \\ (F \vee G) \wedge (\neg F \vee \neg G) \text{ muulloin.} \end{cases}$$

Tällöin $x \in \mathrm{EQUIV} \Leftrightarrow f(x) \in \overline{\mathrm{SAT}}$. Muunnos $f$ on polynomisessa ajassa laskettava, joten $\mathrm{EQUIV} \leq_m^p \overline{\mathrm{SAT}}$.
Kieliluokka co-NP on suljettu polynomisessa ajassa laskettavien muunnosten suhteen (harjoitus 2, tehtävä 4 b). Koska SAT $\in$ NP, $\overline{\mathrm{SAT}} \in$ co-NP ja myös EQUIV $\in$ co-NP $= \Pi_1^p$.

(b) Kieleen B kuulukoot syntaktisesti kelvolliset lausekalkyylin kaavat. Tämä kieli tunnistetaan polynomisessa ajassa, siis B $\in$ P.
Helposti nähdään, että MINIM $= \mathrm{B} \cap \forall^q \overline{\mathrm{EQUIV}}$ polynomilla $q(n) = n - 1$:

$$\begin{aligned} & x \in \mathrm{MINIM} \\ \Leftrightarrow\ & x \in \mathrm{B} \text{ ja } F \not\equiv G \text{ kaikilla } G, \text{ joilla } |G| < |F| \\ \Leftrightarrow\ & x \in \mathrm{B} \text{ ja } \langle F, G \rangle \in \overline{\mathrm{EQUIV}} \text{ kaikilla } G, \text{ joilla } |G| < |F| \\ \Leftrightarrow\ & x \in \mathrm{B} \cap \forall^q \overline{\mathrm{EQUIV}}. \end{aligned}$$

Koska EQUIV $\in \Pi_1^p$, $\overline{\mathrm{EQUIV}} \in$ co-$\Pi_1^p = \Sigma_1^p$ ja MINIM $\in \Pi_2^p$.

2. 
- Olkoon NP suljettu pT-palautusten suhteen, ts. jos $A \in$ NP ja $B \leq_T^p A$, niin myös $B \in$ NP.
  Olkoon $A \in$ NP. Tällöin lauseen 3.2 kohtien

  $$\begin{aligned} (i) &\quad A \leq_T^p A; \\ (iii) &\quad \text{jos } A \leq_T^p B, \text{ niin myös } \overline{A} \leq_T^p B \text{ ja } A \leq_T^p \overline{B}; \end{aligned}$$

  perusteella $\overline{A} \leq_T^p B$ ja oletuksen mukaan $\overline{A} \in$ NP. Näin nähdään, että NP = co-NP.
- Olkoon NP = co-NP.
  Olkoon $A \in$ NP ja $B \leq_T^p A$. Tällöin $B \in \mathrm{P}(A)$ eli kieli $B$ tunnistetaan deterministisesti polynomisessa ajassa jollakin oraakkelikoneella $M$ ja oraakkelilla $A$. Korvataan $M$:n tilassa $q_?$ tapahtuva oraakkelikysely kielet $A$ ja $\overline{A}$ tunnistavien epädeterminististen Turingin koneiden $N$ ja $N'$ simuloinnilla seuraavasti:
  Ensin simuloidaan konetta $N$ oraakkelinauhalla olevalla syötteellä. Jos $N$ hyväksyy syötteen, jatketaan $M$:n tilasta $q_{\mathrm{yes}}$. Jos $N$ hylkää syötteen, syöte voi silti kuulua kieleen $A$. Silloin simuloidaan konetta $N'$ oraakkelinauhalla olevalla syötteellä. Jos $N$ hyväksyy syötteen, jatketaan $M$:n tilasta $q_{\mathrm{yes}}$. Jos $N$ hylkää syötteen, jatketaan $M$:n tilasta $q_{\mathrm{no}}$.
  Modifioitu $M$ toimii ekvivalentisti alkuperäisen $M$:n kanssa epädeterministisenä Turingin koneena, joka tunnistaa kielen $B$ polynomisessa ajassa, sillä $M$ tekee oraakkelikyselyjä polynomisen määrän, joihin $N$ ja $N'$ antavat vastauksen polynomisessa ajassa. Näin nähdään, että NP on suljettu pT-palautusten suhteen.

3. (a) Olkoon $\varphi$ (mahdollisesti kvantifioitu) lausekalkyylin kaava ja $x$ jokin tässä kaavassa esiintyvä muuttuja. Merkitään $\varphi[b/x]$:llä kaavaa, joka saadaan $\varphi$:stä sijoittamalla $x$:lle arvo $b \in \{0, 1\}$ ja sieventämällä. Voidaan todeta, että aina on $|\varphi[b/x]| < |\varphi|$. Tämän perusteella saadaan kielille SAT ja QBF seuraavat itsepalautusalgoritmit:

   SAT($\varphi$):

   - Jos $\varphi$:ssä ei ole yhtään muuttujaa, määritä sen vakioarvo; jos $\varphi \equiv 1$, niin hyväksy, muuten hylkää.
   - Muuten olkoon $x$ jokin $\varphi$:n muuttuja. Määritä oraakkelikutsuilla totuusarvot $b_0$ ja $b_1$:

   $$b_0 \leftarrow \mathrm{SAT}(\varphi[0/x]),$$
   $$b_1 \leftarrow \mathrm{SAT}(\varphi[1/x]).$$

   Jos $b_0 = 1$ tai $b_1 = 1$, niin hyväksy, muuten hylkää.

   QBF($\varphi$):

   - Jos kaava $\varphi$ on kvanttoriton, määritä sen arvo suoraan.
   - Muuten on kaksi tapausta: $\varphi = (\exists x)\psi$ ja $\varphi = (\forall x)\psi$ jollakin kaavalla $\psi$. Määritä oraakkelikutsuilla totuusarvot $b_0$ ja $b_1$:

   $$b_0 \leftarrow \mathrm{QBF}(\psi[0/x]).$$
   $$b_1 \leftarrow \mathrm{QBF}(\psi[1/x]).$$

   $\exists$-tapauksessa palauta tulos $b_0 \vee b_1$, $\forall$-tapauksessa $b_0 \wedge b_1$.

   (b) *Väite.* Jos kieli $A$ on itsepalautuva, niin $A \in \mathrm{PSPACE}$.

   *Todistus.* Olkoon $A = L(M, A)$, missä $M$ on polynomisessa ajassa $p(n)$ toimiva oraakkelikone.

   Konetta $M$ voidaan ajatella myös "rekursiivisena ohjelmana" ja simuloida sen toimintaa pinon avulla: aina kun $M$ syötteellä $x$ tekee oraakkelikyselyn jonosta $y$, $|y| < |x|$, talletetaan $M$:n kyselyhetkinen tilanne pinoon ja käynnistetään uusi $M$:n laskenta syötteellä $y$. Tällöin syötteellä $x$, jonka pituus on $n$, pinossa voi kerrallaan olla enintään $n$ talletettua tilannetta, kukin kooltaan $O(p(n))$ merkkiä. Siten konetta $M$ voidaan simuloida tilassa $O(np(n))$. $\Lambda$

4. Jos simuloitava kone on epädeterministinen, on mahdollista, että tarkistuksen

$$\underline{\text{verify}}\ \{\mathrm{Chk}(t-1, i+j, a_j) \mid j = -1, 0, 1, 2\}$$

löytämät merkkeihin $a_j$ johtavat laskennat kuuluvat epädeterministisen koneen tilannepuun eri polkuihin, mikä ei takaa, että olisi olemassa jokin polku, joka tuottaa yhtä aikaa kaikki merkit $a_j$. Siksi tällä menetelmällä ei voida todistaa, että $\mathrm{NTIME}(2^{cs(n)}) \subseteq \mathrm{ASPACE}(s(n))$.

5. *Väite.* Kaikilla $k \geq 0$ on $\Sigma_k\mathrm{P} = \Sigma_k^p$ ja $\Pi_k\mathrm{P} = \Pi_k^p$.

*Todistus.* Olkoon $k \geq 0$. Jos vaihdetaan keskenään $\Pi_k$-koneen hyväksyvät ja hylkäävät tilat sekä eksistentiaaliset ja universaaliset tilat, saadaan $\Sigma_k$-kone, joka tunnistaa alkuperäisen koneen tunnistaman kielen komplementin. Siten väitteen $\Sigma_k\mathrm{P} = \Sigma_k^p$ todistaminen riittää, koska silloin $\Pi_k\mathrm{P} = \text{co-}\Sigma_k\mathrm{P} = \text{co-}\Sigma_k^p = \Pi_k^p$.

Väite on tosi indekseillä $k = 0$ ja $k = 1$, sillä $\Sigma_0$-kone on deterministinen Turingin kone ja $\Sigma_1$-kone on epädeterministinen Turingin kone, ja siten $\Sigma_0^p = \Pi_0^p = \mathrm{P} = \Sigma_0\mathrm{P} = \Pi_0\mathrm{P}$, $\Sigma_1^p = \mathrm{NP} = \Sigma_1\mathrm{P}$ ja $\Pi_1^p = \text{co-NP} = \Pi_1\mathrm{P}$.

Olkoon $k \geq 1$. Oletetaan, että $\Sigma_k\mathrm{P} = \Sigma_k^p$ ja $\Pi_k\mathrm{P} = \Pi_k^p$ (induktio-oletus).

- Olkoon $A \in \Sigma_{k+1}^p$. Osoitetaan, että $A \in \Sigma_{k+1}\mathrm{P}$.

  Nyt $A = \exists^q B$ jollakin polynomilla $q$, missä kieli $B$ kuuluu luokkaan $\Pi_k^p$, joka on induktio-oletuksen mukaan sama kuin luokka $\Pi_k\mathrm{P}$. Olkoon $M$ kielen $B$ polynomisessa ajassa tunnistava $\Pi_k$-kone.

  Alternoiva eksistentiaalisessa tilassa aloittava Turingin kone $N$ generoikoon epädeterministisesti kaikki merkkijonot $y$, joilla $|y| \leq q(|x|)$, ja testatkoon merkkijonon $\langle x, y\rangle$ kuulumisen kieleen $B$ kuten $M$. Jos $M$ hyväksyy $\langle x, y\rangle$:n jollakin $y$, niin $N$ hyväksyy, muuten hylkää $x$:n.

  Kone $N$ tunnistaa kielen $A = \exists^q B$, sillä

$$
\begin{aligned}
& x \in A \\
\Leftrightarrow\ & \langle x, y\rangle \in B \text{ jollakin } y, \text{ jolla } |y| \leq q(|x|) \\
\Leftrightarrow\ & M \text{ hyväksyy } \langle x, y\rangle\text{:n jollakin } y, \text{ jolla } |y| \leq q(|x|) \\
\Leftrightarrow\ & N \text{ hyväksyy } x\text{:n.}
\end{aligned}
$$

  Kielen $A$ tunnistaminen koneella $N$ tapahtuu polynomisessa ajassa, sillä syötteellä $x$ merkkijonon $\langle x, y\rangle$ muodostaminen ja testaaminen kieleen $B$ kuuluvaksi konetta $M$ simuloiden tapahtuvat polynomisessa ajassa.

  Kone $N$ alternoi tasan yhden kerran useammin kuin $M$, joka alternoi enintään $k-1$ kertaa, joten $N$ alternoi enintään $k$ kertaa. Alkutilan mukaan $N$ on $\Sigma_{k+1}$-kone.

  Koska kieli $A$ tunnistetaan polynomisessa ajassa $\Sigma_{k+1}$-koneella, $A \in \Sigma_{k+1}\mathrm{P}$.

- Olkoon sitten $A \in \Sigma_{k+1}\mathrm{P}$. Osoitetaan, että $A \in \Sigma_{k+1}^p$.

  Olkoon $M$ polynomisessa ajassa $q$ kielen $A$ tunnistava $\Sigma_{k+1}$-kone. Määritellään $\mathrm{Continue}_M$ kieleksi, johon kuuluvat koneen $M$ hyväksymiseen johtavat universaaliset tilanteet $\alpha$, joista lähtien $M$ pysähtyy ajassa $q(|\alpha|)$ ja alternoi enintään $k-1$ kertaa. Saavutettavasta tilanteesta alkava $M$:n laskenta pysähtyy vaaditussa ajassa ja sallituilla alternaatioilla, joten $\mathrm{Continue}_M$ sisältää kaikki saavutettavat hyväksymiseen johtavat universaaliset tilanteet.

  Kieli $\mathrm{Continue}_M$ tunnistetaan alternoivalla Turingin koneella $N$, jonka alkutila on universaalinen. Tämä kone tarkistaa, että syöte on syntaktisesti kelvollinen, kirjoittaa annetun tilanteen $\alpha$ nauhasisällöt työnauhoilleen, siirtää nauhapäät tilannetta $\alpha$ vastaavaan kohtaan ja simuloi sitten konetta $M$ jatkaen laskentaa tilanteeseen $\alpha$ kuuluvassa koneen tilassa. Koneen $N$ laskenta-aikaa rajoitetaan laskurilla, jonka alkuarvoksi asetetaan $q(|\alpha|)$ ja jota vähennetään yhdellä jokaisen simulointiaskelen jälkeen. Jos laskuri nollautuu, $N$ hylkää syötteen.

  Koneen $M$ alternaatioita seuraava kone $N$ alternoi enintään $k-1$ kertaa. Alkutilan mukaan $N$ on $\Pi_k$-kone.

  Kone $N$ tunnistaa kielen $\mathrm{Continue}_M$ polynomisessa ajassa, sillä syötteen tarkistaminen ja siirtäminen työnauhoille sekä jatkolaskenta konetta $M$ aikarajoitetusti simuloiden tapahtuvat polynomisessa ajassa. Koska $N$ on $\Pi_k$-kone, kieli $\mathrm{Continue}_M$ kuuluu luokkaan $\Pi_k\mathrm{P}$, joka on induktio-oletuksen mukaan sama kuin luokka $\Pi_k^p$.

  Määritellään kielet

$$
\begin{aligned}
B &= \{x \mid M \text{ hyväksyy syötteen } x \text{ ennen ensimmäistä alternaatiota}\} \\
C &= \{\langle x, \alpha\rangle \mid \alpha \in \mathrm{Continue}_M\}, \\
D &= \{\langle x, \alpha\rangle \mid \alpha \text{ on koneen } M \text{ syötteen } x \text{ laskentapuuhun kuuluva} \\
&\qquad \text{tilanne, jossa } M \text{ alternoi ensimmäisen kerran}\}.
\end{aligned}
$$

  Nyt pätee

$$x \in A \Leftrightarrow x \in B \text{ tai } (\exists \alpha, |\alpha| \leq q(|x|))(\langle x, \alpha\rangle \in C \cap D)$$

eli $A = B \cup \exists^q(C \cap D)$ jollakin polynomilla $q$.

Kieli $B$ tunnistetaan epädeterministisesti polynomisessa ajassa, joten $B \in \Sigma_1^p \subseteq \Sigma_{k+1}^p$. Koska $C \leq_m^p \mathrm{Continue}_M \in \Pi_k^p$, myös $C \in \Pi_k^p$. Kieli $D$ tunnistetaan epädeterministisesti polynomisessa ajassa koneella, joka tarkistaa syötteen olevan muotoa $\langle x, \alpha \rangle$, simuloi konetta $M$ syötteellä $x$ ensimmäiseen alternaatioon saakka ja tällöin vertaa $\alpha$:aa saavutettuun tilanteeseen. Tämä kone hylkää syötteet, joilla laskenta päättyy ennen $M$:n ensimmäistä alternaatiota.

Tiedetään, että $C \in \Pi_k^p \subseteq \Sigma_{k+1}^p$ ja $D \in \Sigma_1^p \subseteq \Sigma_{k+1}^p$. Siten $C \cap D \in \Sigma_{k+1}^p$ ja, koska $\Sigma_{k+1}^p$ on suljettu polynomirajoitetun eksistenssikvantifioinnin suhteen, myös $\exists^q(C \cap D) \in \Sigma_{k+1}^p$ ja edelleen $A = B \cup \exists^q(C \cap D) \in \Sigma_{k+1}^p$.

On osoitettu, että jos $k \geq 1$ ja $\Sigma_k \mathrm{P} = \Sigma_k^p$ ja $\Pi_k \mathrm{P} = \Pi_k^p$, niin $\Sigma_{k+1}\mathrm{P} = \Sigma_{k+1}^p$. Komplementtiluokkia koskevan huomion mukaan tällöin on voimassa myös väite $\Pi_{k+1}\mathrm{P} = \Pi_{k+1}^p$. Koska väite on todistettu, kun $k = 0$ ja $k = 1$, sen voimassaolo kaikilla $k \geq 0$ seuraa induktiosta. $\Lambda$

6. *Väite.* Kun $k \geq 1$, $U_k$ on $\Sigma_k^p$-täydellinen.

   *Todistus.* Selvästi $U_k \in \Sigma_k \mathrm{P} = \Sigma_k^p$.

   Olkoon $A = L(M)$, missä $M$ on $\Sigma_k$-kone, $\mathrm{time}_M(n) \leq q(n)$, $q$ polynomi. Tällöin kuvaus

$$x \mapsto \langle M, x, 0^{q(|x|)} \rangle$$

   on polynomisessa ajassa laskettava palautus kielestä $A$ kieleen $U_k$. $\Lambda$