Teorija složenosti algoritama

Milenko Mosurović

Univerzitet Crne Gore

studijska godina 2020/21.

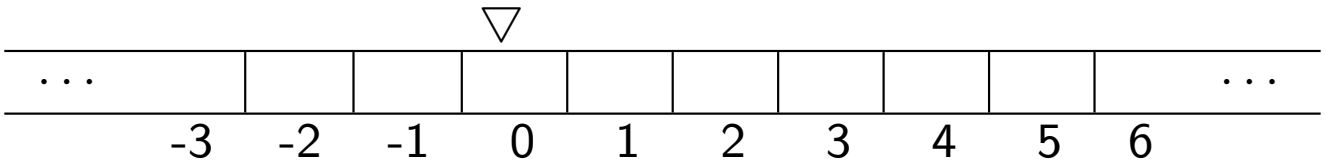
Uvodne napomene o predmetu

- Fond časova 3+1.
- Domaći 10 bodova
- I kolokvijum (teorija) 25 bodova (oko VIII, IX nedelje nastave).
- II kolokvijum (projekat-programiranje) 25 bodova (oko XII nedelje nastave).
- Završni ispit (teorija) nosi 40 bodova.
- Aksiome znanja: ako nešta znam onda je to tačno; ako nešta znam onda znam da to znam; ako nešta ne znam onda znam da to ne znam.
- Prelazna ocjena se dobija za 50 ili više bodova.
- Literatura: Skenirana sveska na sajtu (ne obuhvata čitavo gradivo) i dodatni materijali (na engleskom).
- Napomena. **Slajdovi nisu za učenje.** Na njima je napisano samo ono što je teže objasniti riječima (formule, slike,. . .).

Tjuringova mašina (TM)

- kao model izračunjivosti -

Mehanički opis TM



- $A = \{a_1, \ldots, a_t\} \neq \emptyset$ -azbuka; $a_0 = \square$ -prazan simbol
- $\bigtriangledown$ - glava mašine (označava radno polje)
- r, l, o (ili $+1, -1, 0$) -pomjeranja
- Operacioni uređaj $\quad Q = \{q_0, \ldots, q_s\} \neq \emptyset$ -skup stanja
- $q_0 \in Q$ - početno stanje
- $\mathcal{P}$ - program;
- (q, a, q', a', m) - komanda programa, $\quad m \in \{r, l, o\}$
- $\perp \notin Q$ - završno stanje

Matematički model TM

- $\mathfrak{M} = (A, Q, q_0, \mathcal{P})$
- $A = \{a_1, \ldots, a_t\} \neq \emptyset$ -azbuka;
- $Q = \{q_0, \ldots, q_s\} \neq \emptyset$ -skup stanja
- $q_0 \in Q$ - početno stanje
- $\mathcal{P} \subseteq \Big(Q \times (A \cup \{\square\})\Big) \times \Big((Q \cup \{\perp\}) \times (A \cup \{\square\}) \times M\Big)$ - program
  $M = \{r, l, o\}; \quad \square \notin A; \quad \perp \notin Q$
- $((q, a), (q', a', m)) \in \mathcal{P} \quad \Leftrightarrow \quad (q, a, q', a', m) \in \mathcal{P}$
- Nedeterministička Tjuringova mašina - NDTM; $\quad \mathcal{P}$ je relacija
- Deterministička Tjuringova mašina - DTM; $\quad \mathcal{P}$ je funkcija

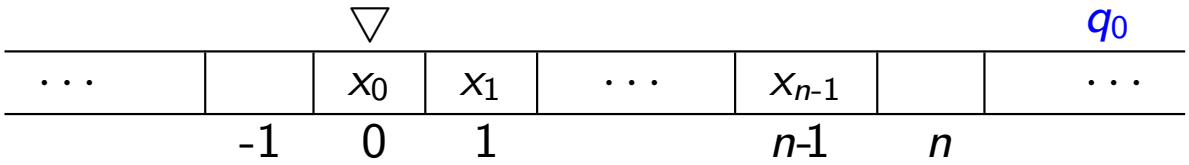
$$\mathcal{P} \; : \; \Big(Q \times (A \cup \{\square\})\Big) \to \Big((Q \cup \{\perp\}) \times (A \cup \{\square\}) \times M\Big)$$

- $\mathrm{Dom}(\mathcal{P}) \subseteq Q \times (A \cup \{\square\})$; $\quad$ Ako $(q, a) \notin \mathrm{Dom}(\mathcal{P})$ onda proširimo $\mathcal{P}(q, a) = (q', a, o)$, $q' \notin Q$ i $\forall b \in A \cup \{a_0\}\ \mathcal{P}(q', b) = (q', b, r)$.

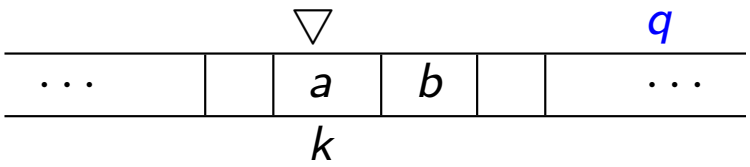
Opis rad TM

- $x = x_0 \dots x_{n-1} \in A'$, $|x| = n$.

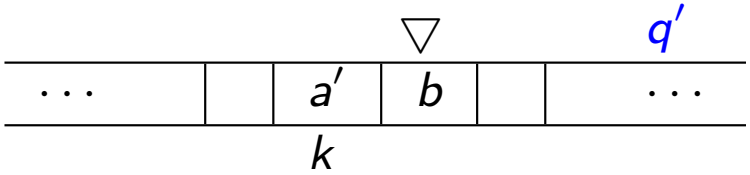
$\lceil \lambda \in A',$
$\omega \in A' \& a \in A \Rightarrow \omega a \in A' \rfloor$



- Dokle smo stigli: sadržaj trake, pozicija glave, stanje. Konfiguracija
- Jedan Korak mašine



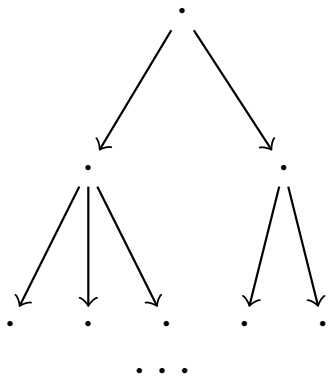
$\Downarrow \quad (q, a, q', a', r)$

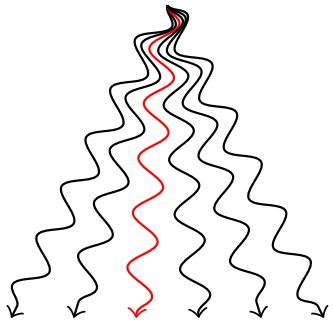


Izračunavanje TM

- Konfiguracija: zapis oblika $C = \$\omega(q, a)\omega'\$$, gdje su $\omega, \omega' \in \overline{A}'$
- $C_0 = \$(q_0, x_0)x_1 \dots x_{n-1}\$$ - polazna konfiguracija, odgovara ulazu $x = x_0 \dots x_{n-1}$
- C' dobijamo iz C primjenom neke komande iz $\mathcal{P}$
  u oznaci $C \Rightarrow C'$ ili $C \overset{\mathcal{P}}{\Rightarrow} C'$ ako ...

  Pr. 1^o) $C = \$\omega b(q, a)\omega'\$$; $(q, a, q', a', l) \in \mathcal{P}$; $C' = \$\omega(q', b)a'\omega'\$$
  2^o) $C = \$(q, a)\omega'\$$; $(q, a, q', a', l) \in \mathcal{P}$; $C' = \$(q', \square)a'\omega'\$$
- Izračunavanje koje odgovara ulazu (na ulazu) x: $\mathfrak{C} = C_0 C_1 C_2 \dots$;
  C_0-polazna koja odgovara x, a $C_i \Rightarrow C_{i+1}$ ili $C_{i+1} = C_i$.
- $C = \$\omega(\perp, a)\omega'\$$ - završna konfiguracija; Zapisuje se i $C = \$\omega a\omega'\$$
- $\mathfrak{C}$ - prihvatljivo ako $(\exists m)C_m$ - završna konfiguracija
- **Def.** Ulaz $x = x_0 \dots x_{n-1} \in A'$ je prihvatljiv TM $\mathfrak{M} = (A, Q, q_0, \mathcal{P})$ ako postoji izračunavanje $\mathfrak{C}$ TM $\mathfrak{M}$ koje odgovara ulazu x i koje je prihvatljivo.





- prihvatljivo izračunavanje
- ostala izračunavanja

- $\mathfrak{C}$ nije prihvatljivo ako: 1^o) TM vječno radi ili 2^o) dođe u ćorsokak (tj. $C = \$\omega(q, a)\omega'\$$ a komanda oblika $(q, a, \cdot, \cdot, \cdot) \notin \mathcal{P}$).

- $L_{\mathfrak{M}} = \{\omega \in A' \ : \ \omega$ je prihvatljiv TM $\mathfrak{M}\}$ - jezik TM $\mathfrak{M}$.
- Glavni zadatak: Za $\omega \in A'$ i jezik $L \subseteq A'$ da li je $\omega \in L$?

  Ovo je pitanje prepoznaănja jezika. Odgovor je "DA" ili "NE".
- Ako postoji izračunavanje $\mathfrak{C} = C_0 C_1 C_2 \ldots$ TM $\mathfrak{M}$ koje odgovara ulazu x i kod koga je za neko $m > 1$, $C_m = \$\omega(\perp, y_0)y_1 \ldots y_r\$$ završna konfiguracija onda kažemo da TM $\mathfrak{M}$ ulaz x transformiše u izlaz $y = y_0 y_1 \ldots y_r$ i označavamo sa $f_{\mathfrak{M}}(x) = y$.
- $\mathfrak{M}$ DTM onda $f_{\mathfrak{M}}$ -funkcija.

Složenost

- **Vremenska složenost** (v.s.)-br. koraka do završne konfiguracije.
- $t(\mathfrak{M}, x) = \min\{m \ : \ \exists C_0 \dots C_m \dots, C_m\text{-završna konf.}\}$ v.s. $\mathfrak{M}$ na x.
- $T(\mathfrak{M}, n) = \max\{t(\mathfrak{M}, x) \ : \ x \in L_{\mathfrak{M}}, |x| = n\}$ v.s. TM $\mathfrak{M}$ na ulazu veličine n - najgori slučaj.
- $T_p(\mathfrak{M}, n) = \sum_{x : x \in L_{\mathfrak{M}}, |x| = n} p(x) \cdot t(\mathfrak{M}, x)$ -v.s. prosječni (očekivani) slučaj; $p(x)$ - vjerovatnoća da se na ulazu pojavi x.

- **Prostorna složenost** (p.s.)-br. simbola najduže konfiguracije.
- Dužina (p.s.) konf. $C = \$b_1 \dots b_{k-1}(q, b_k)b_{k+1} \dots b_r\$$ je $s(C) = r$.
- Za prihvatljivo $\mathfrak{C} = C_0 C_1 C_2 \dots$ p.s. $s(\mathfrak{M}, \mathfrak{C}) = \max\{s(C_i) \ : \ i \in \mathbb{N}\}$.
- $s(\mathfrak{M}, x) = \min\{s(\mathfrak{M}, \mathfrak{C}) \ : \ \mathfrak{C}\text{-izračunavanje koje prihvata } x\}$.
- $S(\mathfrak{M}, n) = \max\{s(\mathfrak{M}, x) \ : \ x \in L_{\mathfrak{M}}, |x| = n\}$ p.s. TM $\mathfrak{M}$ na ulazu veličine n - najgori slučaj. Slično $S_p(\mathfrak{M}, n)$ -prosječni slučaj.

- Primjer. (vremensko prostorni dijagram $\mathfrak{D}$) $\mathfrak{M} = (A, Q, q, \mathcal{P})$, $A = \{0, 1\}$, $Q = \{q, p\}$, $\mathcal{P} = \{(q, 0, q, 0, r);\ (q, 1, q, 1, r);$ $(q, \square, p, \square, l);\ (p, 0, \perp, 1, o);\ (p, 1, p, 0, l);\ (p, \square, \perp, 1, o)\}$.
  Za ulaz $x = 1011$ dobijamo niz konf. (tj. vremensko prostorni diag.)

\$	$\square$	$(q, 1)$	0	1	1	$\square$	\$
\$	$\square$	1	$(q, 0)$	1	1	$\square$	\$
\$	$\square$	1	0	$(q, 1)$	1	$\square$	\$
\$	$\square$	1	0	1	$(q, 1)$	$\square$	\$
\$	$\square$	1	0	1	1	$(q, \square)$	\$
\$	$\square$	1	0	1	$(p, 1)$	$\square$	\$
\$	$\square$	1	0	$(p, 1)$	0	$\square$	\$
\$	$\square$	1	$(p, 0)$	0	0	$\square$	\$
\$	$\square$	1	$(\perp, 1)$	0	0	$\square$	\$

  Vidimo v.s. je 9 a p.s. je 6.
- Šta prethodni program računa?

$$\begin{array}{rr} & 1101100111111 \\ + & \qquad\qquad 1 \\ \hline & 1101101000000 \end{array}$$

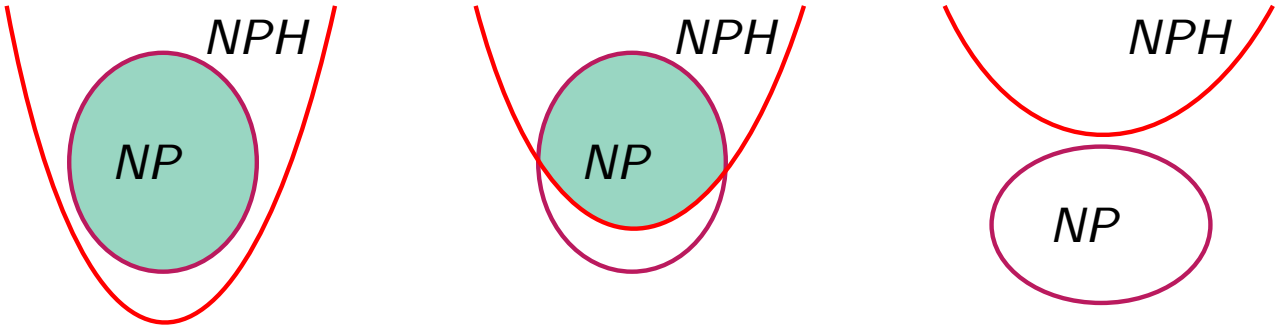
Klase složenosti P i NP

- $P = \{L \ : \ (\exists \text{DTM } \mathfrak{M})(\exists d \in \mathbb{N}) L = L_{\mathfrak{M}} \text{ i } T(\mathfrak{M}, n) \leq d \cdot n^d\}$
- $NP = \{L \ : \ \exists (\text{NDTM } \mathfrak{M})(\exists d \in \mathbb{N}) L = L_{\mathfrak{M}} \text{ i } T(\mathfrak{M}, n) \leq d \cdot n^d\}$
- Umjesto $T(\mathfrak{M}, n) \leq d \cdot n^d$ možemo pisati $T(\mathfrak{M}, n) \leq p(n)$, gdje je $p(\cdot)$ polinom ili $T(\mathfrak{M}, n) \leq c \cdot n^d$, gdje je $c \in \mathbb{R}^+$.
- Svaka DTM je NDTM pa je $P \subseteq NP$.
- Najvažnije pitanje teorijskog računarstva:

Da li je $P = NP$?

- **Def.** Rećićemo da je jezik $L \subseteq \Omega$ polinomijalno svodljiv na jezik $L_1 \subseteq \Omega_1$, u oznaci $L \lhd L_1$, ako postoji DTM $\mathfrak{M}$ i postoji polinom $p(\cdot)$ tako važi:
  1^o) $T(\mathfrak{M}, n) \leq p(n)$ i
  2^o) svaki ulaz $x \in \Omega$ mašina $\mathfrak{M}$ transformiše u $y \in \Omega_1$ na takav način da je
$$x \in L \ \Leftrightarrow \ x \in L_1.$$
- **Def.** $NP-$težak (kraće NPH). $NPH = \{L \ : \ (\forall \hat{L} \in NP)\, \hat{L} \lhd L\}$.

- **Def.** $NP-$kompletan (kraće NPC). $NPC = NP \cap NPH$.



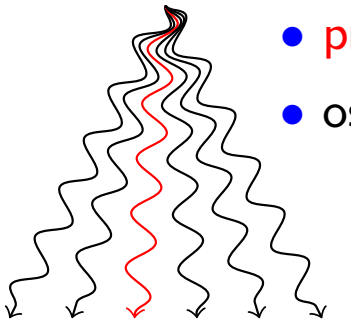
Na slikama obijeni dio prikazuje klasu NPC

- Koja od 3 slike je ispravna?
- Jezik (problem, zadatak) SAT
- x_i - promenljive, $\overline{x}_i$ - njihove negacije, $x^\delta = \begin{cases} x & , \quad \text{ako je } \delta = 1 \\ \overline{x} & , \quad \text{ako je } \delta = 0 \end{cases}$
- Klauzula $D_j = \{x_{i_k}^{\delta_{i_k}} \ : \ k \in K_j\}$ označava $\bigvee\limits_{k \in K_j} x_{i_k}^{\delta_{i_k}}$.
- Formula $\Phi(x_1, \ldots, x_n) = \{D_j \ : \ 1 \leq j \leq m\}$ označava $\bigwedge\limits_{j=1}^{m} D_j$

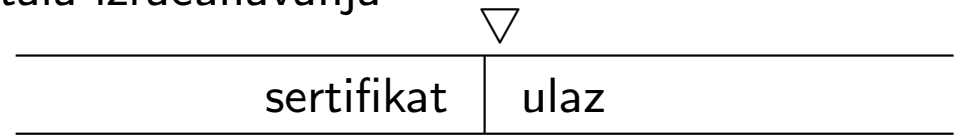
- Formula je u KNF i možemo joj pridružiti f-ju $f : \{0,1\}^n \to \{0,1\}$.
- Ako je $\left(\forall(\alpha_1, \ldots, \alpha_n) \in \{0,1\}^n\right) \Phi(\alpha_1, \ldots, \alpha_n) = 1$ formula je tautologija.
- Ako $\left(\exists(\alpha_1, \ldots, \alpha_n) \in \{0,1\}^n\right) \Phi(\alpha_1, \ldots, \alpha_n) = 1$ formula je zadovoljiva.
- Primjer. $\Phi(x_1, x_2, x_3) = (x_1 \vee x_2 \vee \overline{x}_3) \wedge x_2 \wedge (\overline{x}_1 \vee x_2 \vee \overline{x}_3)$ nije tautologija (zbog (0,0,0)) a jeste zadovoljiva (zbog (0,1,0)). $\Phi(x_1, x_2, x_3) = (x_1 \vee \overline{x}_2 \vee \overline{x}_3) \wedge x_2 \wedge (\overline{x}_1 \vee \overline{x}_2 \vee \overline{x}_3) \wedge x_3$ nije zadovoljiva.
- SAT $= \{\Phi \in$ KNF : Φ-zadovoljiva$\}$.
- Pitanje: Da li je $\Phi \in$ SAT? Ovo je pitanje poznato kao SAT problem tj. problem zadovoljivosti Bulovih formula. Ovo je u stvari prepoznavanje jezika SAT. Odgovori su: "DA" ili "NE".

- **Teorema**. (Kuk-Levinova) Problem SAT je *NP*-kompletan. (Kratko: SAT$\in$ *NPC*)
- Važna. Stalno pitam na kolokvijumu. Znači da $NPC \neq \emptyset$.

- Komentar o NP.



  - prihvatljivo izračunavanje
  - ostala izračunavanja



$\star\star$ NDTM=DTM+nedeterministička komanda

- **Dokaz**. (Kukove teoreme) Da bi dokazali SAT$\in NPC$ treba da dokažemo dvije stvari:
  I) SAT $\in NP$.
  II) SAT je $NP-$težak (tj. SAT $\in NPH$).

I) SAT $\in NP$

$\Phi(x_1, \ldots, x_n) = \{D_j \ : \ 1 \leq j \leq m\}, \quad D_j = \{x_{i_k}^{\delta_{i_k}} \ : \ k \in K_j\}.$

a) Nedeterministič (Ndt.) pogađamo rješenje. Ndt. izaberemo 0 ili 1 za x_1, pa 0 ili 1 za x_2, $\cdots$, 0 ili 1 za x_n. Komande za Ndt. izaber x_1: $(p_1, \square, p_2, 0, I)$; $(p_1, \square, p_2, 1, I)$ vode ka različitim izračunavanjima.

b) Provjeravamo da li Ndt. izabrani niz 0 i 1 zadovoljava Φ. Gledamo redom vrijednosti od D_j. Ako su sve $= 1$ onda $\Phi \in$ SAT, ako naiđemo na vrijednost $D_j = 0$ vraćamo $\Phi \notin$ SAT. Vrijednost za D_j određujemo slično gledajući vrijednosti za $x_{i_k}^{\delta_{i_k}}$.

Opisali smo polinomijalni algoritam tj. TM, pa je SAT $\in NP$.

II) SAT je $NP-$težak

Neka je $L \in NP$ proizvoljan jezik. Dokažimo da je $L \lhd$ SAT.

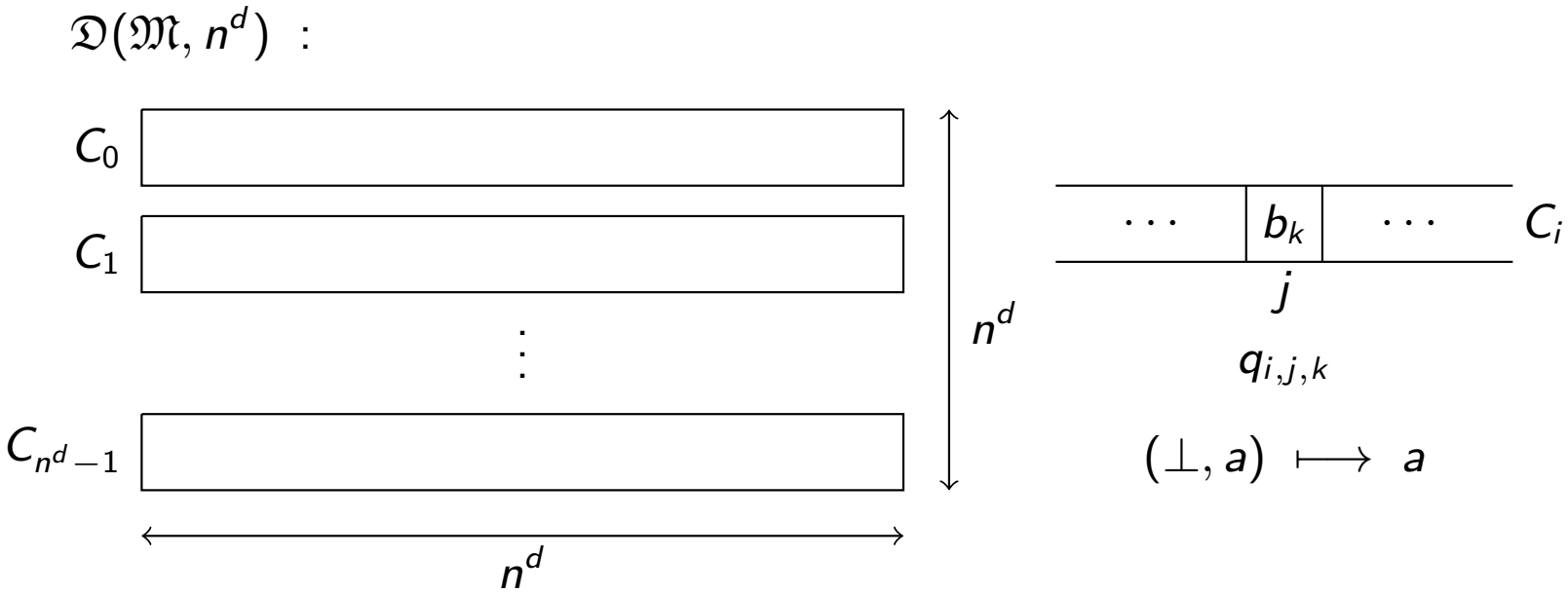
$L \in NP \Leftrightarrow \exists(\text{NDTM } \mathfrak{M})(\exists d \in \mathbb{N}) L = L_{\mathfrak{M}}$ i $T(\mathfrak{M}, n) < n^d$

Za ulaz x, $|x| = n$ (n dovoljno veliki broj) važi:

$x \in L \Leftrightarrow x \in L_{\mathfrak{M}}$ i $T(\mathfrak{M}, |x|) < n^d \Leftrightarrow \exists \mathfrak{C} = C_0, \ldots, C_{n^d-1}, \ldots$ izračunavanje koje odgovara x i C_{n^d-1} završna konf. $\Leftrightarrow$ $\exists \mathfrak{D}(\mathfrak{M}, n^d) = C_0, \ldots, C_{n^d-1}$ vrem. prostorni diag. dimenzija $n^d \times n^d$ $\Leftrightarrow \psi_{\mathfrak{D}} \in$ SAT.

Na osnovu $\mathfrak{D}(\mathfrak{M}, n^d)$ konstruišemo $\psi_{\mathfrak{D}}$. Tačnije na osnovu $\mathfrak{M}$ i n^d.

To ostvarujemo u 4 koraka tj. $\psi_{\mathfrak{D}} = \psi_1 \wedge \psi_2 \wedge \psi_3 \wedge \psi_4$.



$\hat{A} = A' \cup Q \times A' = \{b_1, \ldots, b_l\}$, $l = const$-jer ne zavisi od ulaza. $q_{i,j,k}$ - uzima vrijednost tačno ako je u konfiguraciji C_i na polju j upisan simbol $b_k \in \hat{A}$; $0 \leq i, j < n^d$, $1 \leq k \leq l$.

1. ψ_1 - u svakoj konfiguraciji na svakom polju se nalazi tačno jedan simbol iz $\hat{A}$.
2. ψ_2 - Prelaz sa jedne na sledeću konfiguraciju je u skladu sa programom.
3. ψ_3 - C_0 je polazna konfiguracija koja odgovara ulazu.
4. ψ_4 - C_{n^d-1} je završna konfiguracija (tj. svi simboli iz A').

Formula ψ_1

$q_{i,j,1} \vee q_{i,j,2} \vee \cdots \vee q_{i,j,l}$ - upisan bar jedan

$\neg q_{i,j,k_1} \vee \neg q_{i,j,k_2}$ - nisu upisan dva simbola b_{k_1}, b_{k_2}

$$\psi_1 = \bigwedge_{i=0}^{n^d-1} \bigwedge_{j=0}^{n^d-1} \left[(q_{i,j,1} \vee q_{i,j,2} \vee \cdots \vee q_{i,j,l}) \wedge \left(\bigwedge_{k=1}^{l-1} \bigwedge_{s=k+1}^{l} (\neg q_{i,j,k} \vee \neg q_{i,j,s}) \right) \right]$$

Očigledno: $|\psi_1| = O(n^{2d})$; složenost $O(n^{2d})$

ψ_1 je u KNF.

Formula ψ_2

- Pri prelazu sa jedne konfiguracije na drugu može doći do promjene samo sadržaja radnog polja, a glava mašine se može pomjeriti na susjedna polja. Ostala polja ostaju ista.
- Drugim rječima pri zapisu nove konfiguracije sadržaj polja j može da zavisi od sadržaja polja $j-1$, j i $j+1$ u prethodnoj konfiguraciji.

$\mathfrak{D}(\mathfrak{M}, n^d)$:

		$j-1$	j	$j+1$	
C_i	$\cdots$	b_{k_1}	b_{k_2}	b_{k_3}	$\cdots$
C_{i+1}	$\cdots$		b_{k_4}		$\cdots$

- Interesuju nas četvorke $(b_{k_1}, b_{k_2}, b_{k_3}, b_{k_4}) \in \hat{A}^4$ tj. četvorke (k_1, k_2, k_3, k_4) indeksa. Ima ih $|\hat{A}^4| = t^4 = const$.
- $(1, 0, 1, 1)$ -nije dozvoljena, $(1, 0, 1, 0)$ -jeste dozvoljena; $((q, 1), 0, 1, (p, 0))$ - jeste dozvoljena ako komanda oblika $(q, 1, p, x, r) \in \mathcal{P}$, $x \in A$ u suprotnom nije dozvoljena.

- $NR = \{(k_1, k_2, k_3, k_4) : (b_{k_1}, b_{k_2}, b_{k_3}, b_{k_4})\text{-nije regularna}, 1 \leq k_i \leq l\}$
- $\neg q_{i,j-1,k_1} \vee \neg q_{i,j,k_2} \vee \neg q_{i,j+1,k_3} \vee \neg q_{i+1,j,k_4}$
- $$\psi_2 = \bigwedge_{i=0}^{n^d-1} \bigwedge_{j=0}^{n^d-1} \left[\bigwedge_{(k_1,k_2,k_3,k_4)\in NR} (\neg q_{i,j-1,k_1} \vee \neg q_{i,j,k_2} \vee \neg q_{i,j+1,k_3} \vee \neg q_{i+1,j,k_4}) \right]$$
- Očigledno: $|\psi_2| = O(n^{2d})$; složenost $O(n^{2d})$
- ψ_2 je u KNF.

Formula ψ_3

- $x = x_0 x_1 \cdots x_{n-1}$ -ulaz

- $C_0 \equiv \$(q_0, x_0)x_1 \cdots x_{n-1}\$$ - polazna konfiguracija ili
  $C_0 \equiv \$b_{k_0} b_{k_1} \cdots b_{k_{n-1}}\$$, gdje je $b_{k_0} = (q_0, x_0)$, $\ldots$

$$\psi_3 = \left(\bigwedge_{j=0}^{n-1} q_{0,j,k_j} \right) \wedge \left(\bigwedge_{j=n}^{n^d-1} q_{0,j,k^o} \right), \quad b_{k^o} = a_0$$

- Očigledno: $|\psi_3| = O(n^d)$; složenost $O(n^d)$

- ψ_3 je u KNF.

Formula ψ_4

- Završna konfiguracija sadrži samo simbole iz $A' = \{a_0, a_1, \ldots, a_t\}$ tj. $C_{n^d-1} \equiv \$b_{k_0} b_{k_1} \cdots b_{k_{n^d-1}}\$$, gdje je $b_{k_j} \in A'$

- Neka je $A' = \{a_0, a_1, \ldots, a_t\} = \{b_1, \ldots, b_{t+1}\}$

  $q_{n^d-1,j,1} \vee q_{n^d-1,j,2} \vee \cdots \vee q_{n^d-1,j,t+1}$ - upisan bar jedan simbol iz A'

$$\psi_4 = \bigwedge_{j=0}^{n^d-1} \left(\bigvee_{k=1}^{t+1} q_{n^d-1,j,k} \right)$$

- Očigledno: $|\psi_4| = O(n^d)$; složenost $O(n^d)$

- ψ_4 je u KNF.

- $\psi_{\mathfrak{D}} = \psi_1 \wedge \psi_2 \wedge \psi_3 \wedge \psi_4$
- $|\psi_{\mathfrak{D}}| = O(n^{2d})$; složenost $O(n^{2d})$
- Treba dokazati da je

$$x \in L \Leftrightarrow \psi_{\mathfrak{D}} \in \text{SAT}.$$

  Dokaz, slijedi iz ranije izloženog.

- Komentar dokaza. $\psi_{\mathfrak{D}} \in$ SAT onda postoji valuacija promjenljivih $q_{i,j,k}$ koja zadovoljava $\psi_{\mathfrak{D}}$. Na osnovu te valuacije konstruišemo vremensko prostorni dijagram tako što ako je $q_{i,j,k}$ tačno onda u konfiguraciji C_i u polje j upišemo simbol b_k. Formula ψ_1 nam garantuje da će u svako polje biti upisan tačno jedan simbol i sl. Obrnuto ako je $x \in L$ postoji vremensko prostorni diagram...
  Ako je u konfiguraciji C_i u polje j upišan simbol b_k onda promenljivoj $q_{i,j,k}$ dodjelimo vrijednost tačno inače dodjelimo joj vrijednost netačno.

Odnos jezika sa "podjezikom"

- $$\left.\begin{array}{lcl} L & \in & NP \\ L_1 & \in & NPC \\ L_1 & \subseteq & L \end{array}\right\} \Rightarrow L \in NPC.$$

$$\left.\begin{array}{lcl} L & \in & NCP \\ L_1 & \in & NP \\ L_1 & \subseteq & L \end{array}\right\} \not\Rightarrow L \in NPC.$$

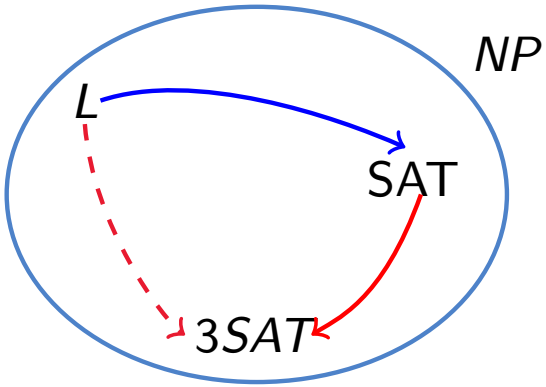
- Zaključak. Ako neko jezik sadrži težak dio onda je čitav jezik težak. Obrnuto ne važi tj. težak jezik može sadržati dijelove koji nisu teški.

$SAT_n = \{\psi \in \text{ SAT } : \psi(x_1, \ldots, x_n)\}$;
$SAT_{n,i} = \{\psi \in SAT_n \ : i \equiv (\alpha_1, \ldots, \alpha_n)_{(2)}$ i $\psi(\alpha_1, \ldots, \alpha_n) = 1\}$
Očigledno $SAT_{n,i} \in P$ dok SAT$= \cup_{n,i} SAT_{n,i}$.

$3SAT$

- $3KNF = \{\psi \in \text{ KNF } : \psi = \bigwedge D_i,\ D_i = x_{j_1}^{\delta_{j_1}} \vee x_{j_2}^{\delta_{j_2}} \vee x_{j_3}^{\delta_{j_3}}\}$
  Svaka klauzula D_i ima tačno 3 literala.
  $3SAT = 3KNF \cap \text{SAT}$ tj.
  $3SAT = \{\psi \in \text{ SAT } : \psi = \bigwedge D_i,\ D_i = x_{j_1}^{\delta_{j_1}} \vee x_{j_2}^{\delta_{j_2}} \vee x_{j_3}^{\delta_{j_3}}\}$

- **Teorema**. $3SAT \in NPC$ tj. problem $3SAT$ je $NP-$kompletan.
- **Dokaz**. 1^o) $3SAT \subset \text{SAT} \Rightarrow 3SAT \in NP$
  2^o) Dokažimo da je $3SAT$ $NP-$težak.



- Dovoljno je dokazati $\text{SAT} \lhd 3SAT$.

- $\Phi \in$ KNF transformišemo u $\Phi' \in 3KNF$.
- $\Phi(x_1, \ldots, x_n) = \bigwedge_{j=1}^{m} D_j$
- Induktivno formiramo Φ_i, $i = 0, 1, \ldots, m$; $\quad \Phi' = \Phi_m$.
- Ideja (i-tog koraka): D_i zamjenimo sa $\phi_i = \bigwedge_{j=1}^{k_i} D_j^i \in 3KNF$.
- $\Phi_0 = \Phi$; ako $\Phi_{i-1} = \bigwedge_{j=1}^{i-1} \phi_j \wedge \bigwedge_{j=i}^{m} D_j$ onda $\Phi_i = \bigwedge_{j=1}^{i} \phi_j \wedge \bigwedge_{j=i+1}^{m} D_j$
- Očigledno: $\phi_i \in 3KNF \Rightarrow \Phi' = \Phi_m \in 3KNF$.
- Treba dokazati: $\Phi_{i-1} \in$ SAT $\Leftrightarrow \Phi_i \in$ SAT.
- Primjer. $D = x_1^{\delta_1} \vee x_2^{\delta_2} \vee x_3^{\delta_3} \vee x_4^{\delta_4}$

  $\phi = (x_1^{\delta_1} \vee x_2^{\delta_2} \vee y) \wedge (\overline{y} \vee x_3^{\delta_3} \vee x_4^{\delta_4}) \in 3KNF$

  Ako je npr. $x_2^{\delta_2} = 1$ onda biramo $y = 0$. Ako $x_j^{\delta_j} = 0$ za svako j onda $\phi \equiv y \wedge \overline{y} = 0$.

- Neka smo formirali Φ_{i-1} formirajmo Φ_i.
- $D_i = (x_{i_1}^{\delta_{i_1}} \vee \cdots \vee x_{i_k}^{\delta_{i_k}})$
- $k = 3$: $\phi_i = D_i$
- $k > 3$:

$$\phi_i = (x_{i_1}^{\delta_{i_1}} \vee x_{i_2}^{\delta_{i_2}} \vee y_1) \wedge (\overline{y_1} \vee x_{i_3}^{\delta_{i_3}} \vee y_2) \wedge \cdots \wedge (\overline{y}_{j-2} \vee x_{i_j}^{\delta_{i_j}} \vee y_{j-1})$$

$$\wedge \cdots \wedge (\overline{y}_{k-4} \vee x_{i_{k-2}}^{\delta_{i_{k-2}}} \vee y_{k-3}) \wedge (\overline{y}_{k-3} \vee x_{i_{k-1}}^{\delta_{i_{k-1}}} \vee x_{i_k}^{\delta_{i_k}})$$

$\Phi_{i-1} \in \text{SAT} \Leftrightarrow \Phi_i \in \text{SAT}$

$(\Rightarrow)$:

$\Phi_{i-1} \in \text{SAT} \Rightarrow (\exists(\alpha_1, \ldots, \alpha_n) \in \{0,1\}^n)\Phi_{i-1}(\alpha_1, \ldots, \alpha_n) = 1$

Specijalno $D_i(\alpha_1, \ldots, \alpha_n) = 1$ pa $(\exists j)x_{i_j}^{\delta_{i_j}} = 1$

Biramo: $y_t = 1$ za $1 \leq t < j-1$ i $y_t = 0$ za $j-1 \leq t \leq k-3$.

Za izabrane vrijednosti: $\phi_i = 1$ pa i $\Phi_i = 1$ tj. $\Phi_i \in \text{SAT}$.

$(\Leftarrow)$:

$\Phi_i \in \text{SAT} \Rightarrow \left(\exists(\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_{k-3}) \in \{0,1\}^{n+k-3}\right)$
$\Phi_i(\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_{k-3}) = 1$

Tada $\Phi_{i-1}(\alpha_1, \ldots, \alpha_n) = 1$. Zaista ako $D_i(\alpha_1, \ldots, \alpha_n) = 0$ onda $x_{i_j}^{\delta_{i_j}} = 0$ za $1 \leq j \leq k$. Pa zbog $\phi_i = 1$ mora biti $y_1 = 1$. Ako je $y_{j-2} = 1$ onda je i $y_{j-1} = 1$ $\lceil$ jer je $(\overline{y}_{j-2} \vee x_{i_j}^{\delta_{i_j}} \vee y_{j-1}) = 1$ $\rfloor$.

Na kraju zbog $y_{k-3} = 1$ dobijamo $(\overline{y}_{k-3} \vee x_{i_{k-1}}^{\delta_{i_{k-1}}} \vee x_{i_k}^{\delta_{i_k}}) = 0$. Što je nemoguće.

Dakle, $D_i = 1$ i $\Phi_{i-1} \in \text{SAT}$.

- $k = 2: \quad \phi_i = (x_{i_1}^{\delta_{i_1}} \vee x_{i_2}^{\delta_{i_2}} \vee y) \wedge (\overline{y} \vee x_{i_1}^{\delta_{i_1}} \vee x_{i_2}^{\delta_{i_2}})$
- $k = 1$:
  $\phi_i = (x_{i_1}^{\delta_{i_1}} \vee y \vee z) \wedge (x_{i_1}^{\delta_{i_1}} \vee y \vee \overline{z}) \wedge (x_{i_1}^{\delta_{i_1}} \vee \overline{y} \vee z) \wedge (x_{i_1}^{\delta_{i_1}} \vee \overline{y} \vee \overline{z})$.

Bojenje grafa

- Neka je dat graf $G = (V, E)$. Par (G, f), gdje je f preslikavanje $f : V \to \mathbb{N}$, nazivamo bojenje čvorova grafa G. Bojenje je pravilno ako je $(\forall (u, v) \in E)\ f(u) \neq f(v)$. Kažemo da bojenje koristi k boja ako je $|f(V)| \leq k$.
- Ako bojenje (G, f) koristi k boja onda postoji bojenje (G, f') takvo da je $|f'(V)| \subseteq \{1, \ldots, k\}$ tj. $f' : V \to \{1, \ldots, k\}$.
- Kažemo da se graf (čvorovi grafa) G može obojiti sa k boja ako postoji pravilno bojenje (G, f) čvorova grafa G koje koristi k boja.
- $L_{bg} = \{(G, k)\ :\ \text{graf } G \text{ se može obojiti sa } k \text{ boja}\}$
- Zadatak k obojivosti grafa: Dat je graf G i $k \in \mathbb{N}$ da li je $(G, k) \in L_{bg}$.

- **Teorema**. Zadatak k obojivosti grafa je $NP-$kompletan (tj. pripada klasi NPC).

  $\lceil\ L_{bg} \in NPC\ \rfloor$

- **Dokaz**. 1^o) ($L_{bg} \in NP$:) $k \in \mathbb{N}$; $G = (V, E)$, $V = \{v_1, \ldots, v_n\}$.

  Algoritam: Nedeterministički izaberemo boju iz skupa $\{1, \ldots, k\}$ za v_1, zatim za v_2, ... Na kraju nedeterministički izaberemo boju za v_n. Za svaku granu $(v, u) \in E$ provjerimo da li je $f(v) \neq f(u)$. Ako jeste odgovor je "DA" u suprotnom odgovor je "NE".

  2^o) ($L_{bg} \in NPH$:) Dovoljno je dokazati $3SAT \lhd L_{bg}$.

  $\Phi(x_1, \ldots, x_n) = \bigwedge_{i=1}^{m} D_i$, $D_i = x_{j_1}^{\delta_{j_1}} \vee x_{j_2}^{\delta_{j_2}} \vee x_{j_3}^{\delta_{j_3}}$ tj. $D_i = \{x_{j_1}^{\delta_{j_1}}, x_{j_2}^{\delta_{j_2}}, x_{j_3}^{\delta_{j_3}}\}$

  Na osnovu $\Phi \in 3KNF$ formiramo $G = (V, E)$ i $k \in \mathbb{N}$

  V : $x_1, \ldots, x_n$, $\overline{x}_1, \ldots, \overline{x}_n$, $y_1, \ldots, y_n$, $D_1, \ldots, D_m$.

  E : $(\forall i)\ (x_i, \overline{x}_i)$; $(\forall i \neq j)\ (y_i, y_j)$; $(\forall i \neq j)\ (x_i, y_j)$; $(\forall i \neq j)\ (\overline{x}_i, y_j)$;

  $(\forall x_j^{\delta_j} \notin D_i)\ (x_j^{\delta_j}, D_i)$;

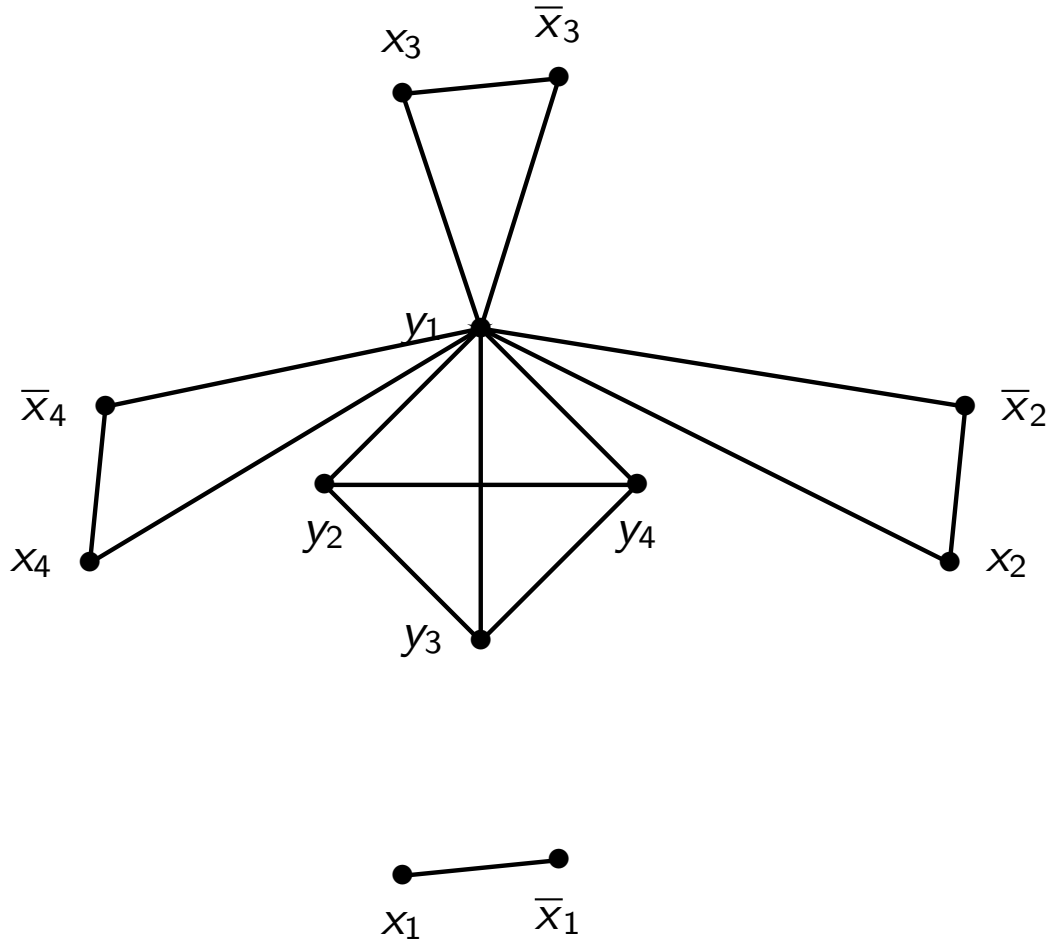
  k : $k = n + 1$. Pretpostavka $n \geq 4$.

  Složenost: (dominira formiranje grana) $O(n^2 + n \cdot m)$

  Pitanje $(G, k) \in L_{gb}$?

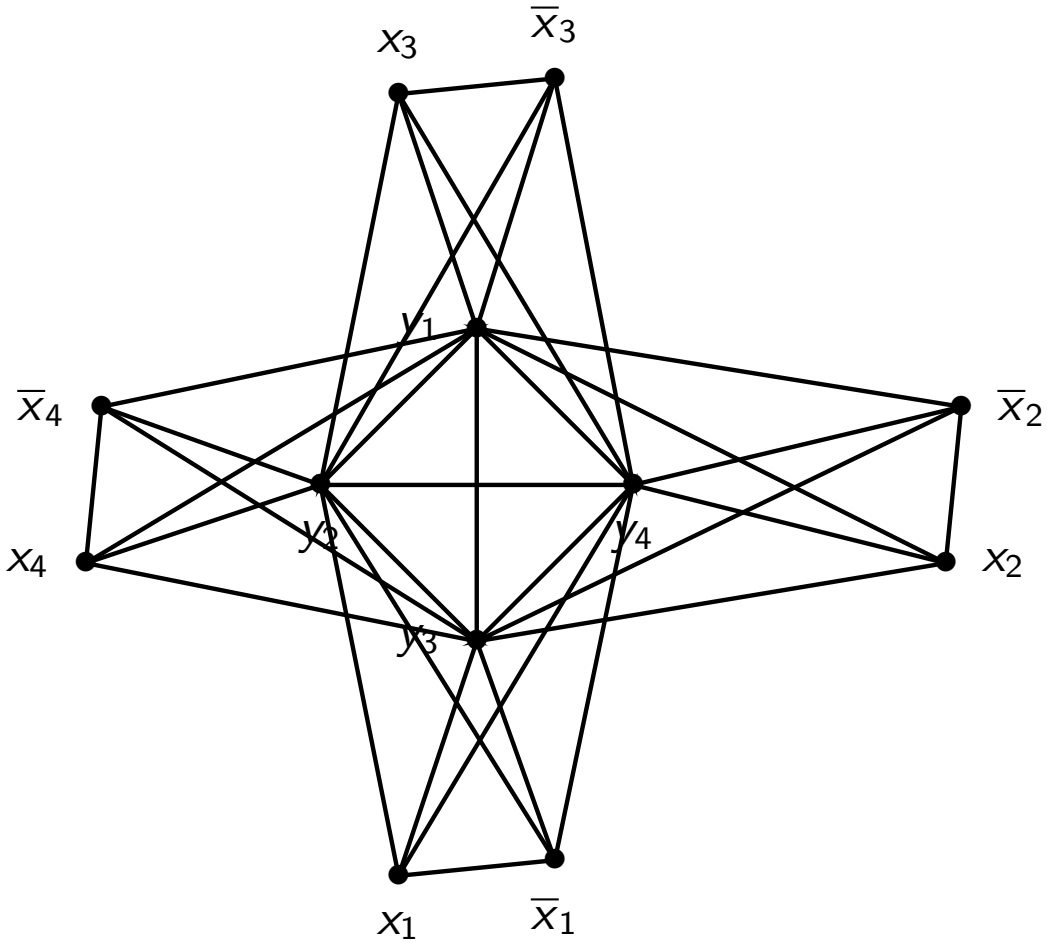
Koristimo n boja $T_1, \ldots, T_n$ za tačno i boju F za netačno.
Treba nam n različitih boja da obojimo čvorove y_i.

$$\Phi = \Phi(x_1, x_2, x_3, x_4)$$



Primjer.

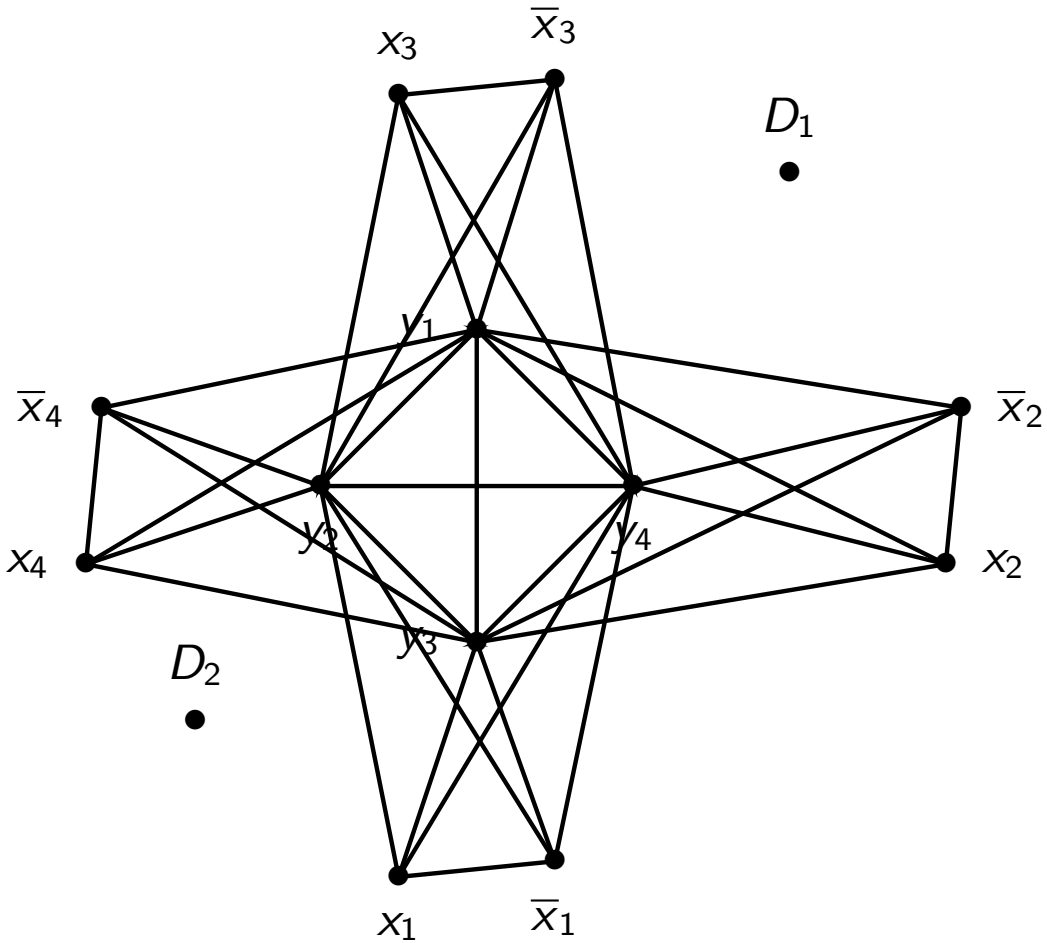
$$\Phi = \Phi(x_1, x_2, x_3, x_4)$$



- Koristimo n boja $T_1, \ldots, T_n$ za tačno i boju F za netačno.
- Treba nam n različitih boja da obojimo čvorove y_i.
- $y_i \to T_i$ (tj. y_i bojimo sa T_i)
- Ne može $x_i \to T_j$ ili $\overline{x}_i \to T_j$ za $i \neq j$
- Ako $x_i \to T_i$ smatramo ga tačnim, tada $\overline{x}_i \to F$ i smatramo ga netačnim. Može biti i $x_i \to F$, $\overline{x}_i \to T_i$.

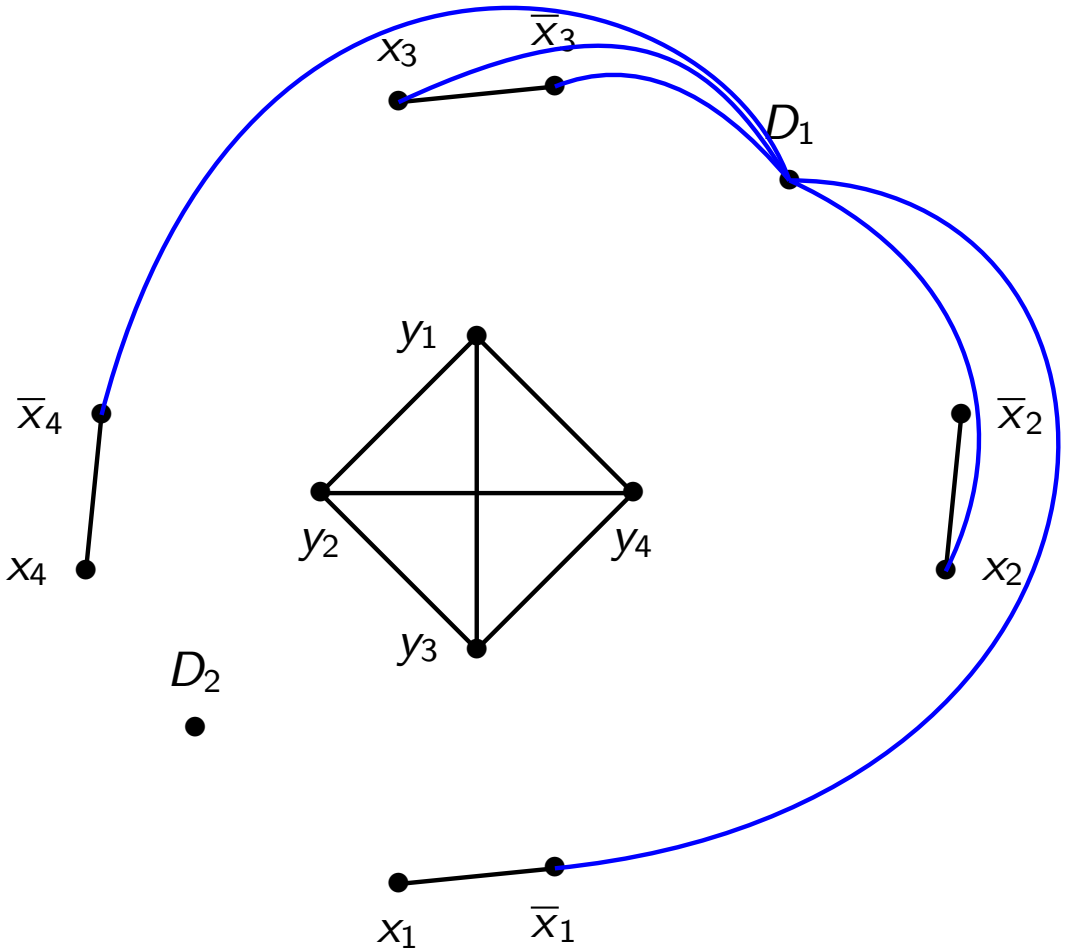
Primjer.

$$\Phi = \Phi(x_1, x_2, x_3, x_4) = (x_1 \vee \overline{x}_2 \vee x_4) \wedge (x_2 \vee \overline{x}_3 \vee \overline{x}_4)$$



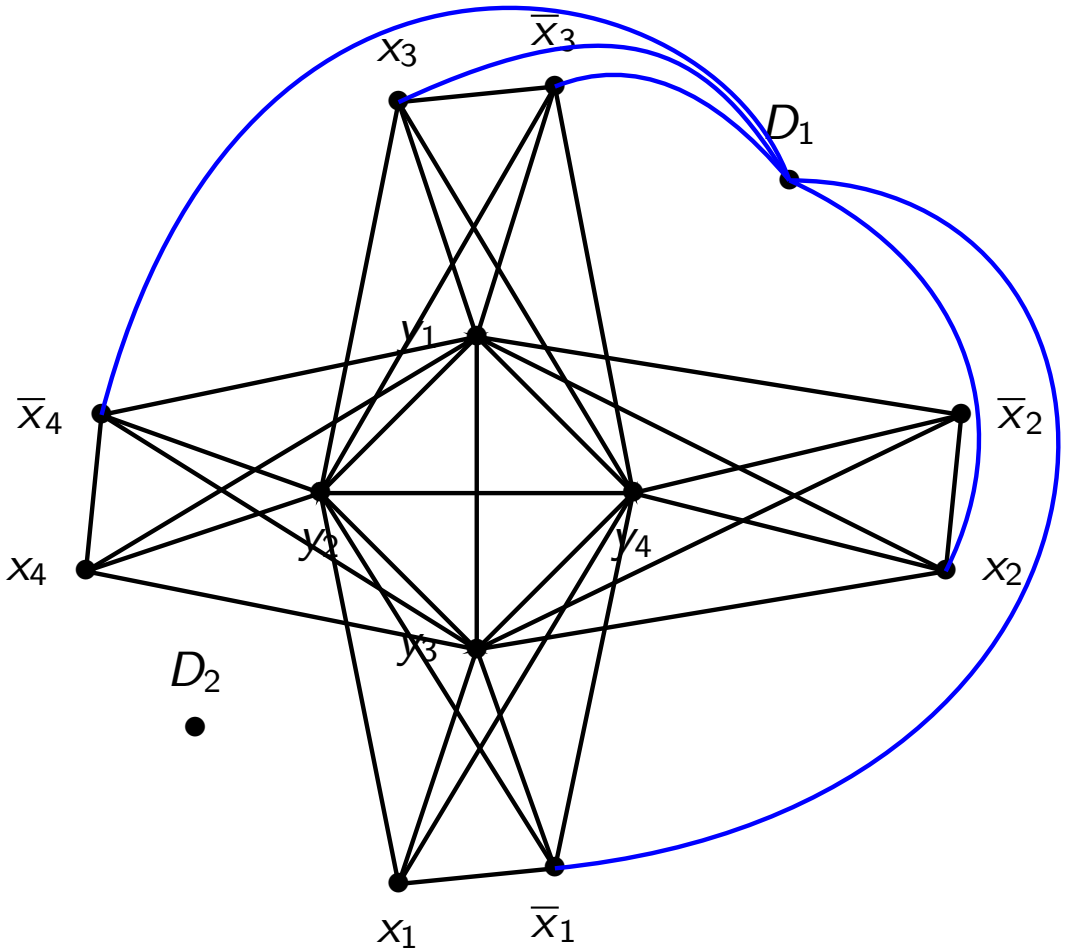
Primjer.

$$\Phi = \Phi(x_1, x_2, x_3, x_4) = (x_1 \vee \overline{x}_2 \vee x_4) \wedge (x_2 \vee \overline{x}_3 \vee \overline{x}_4)$$

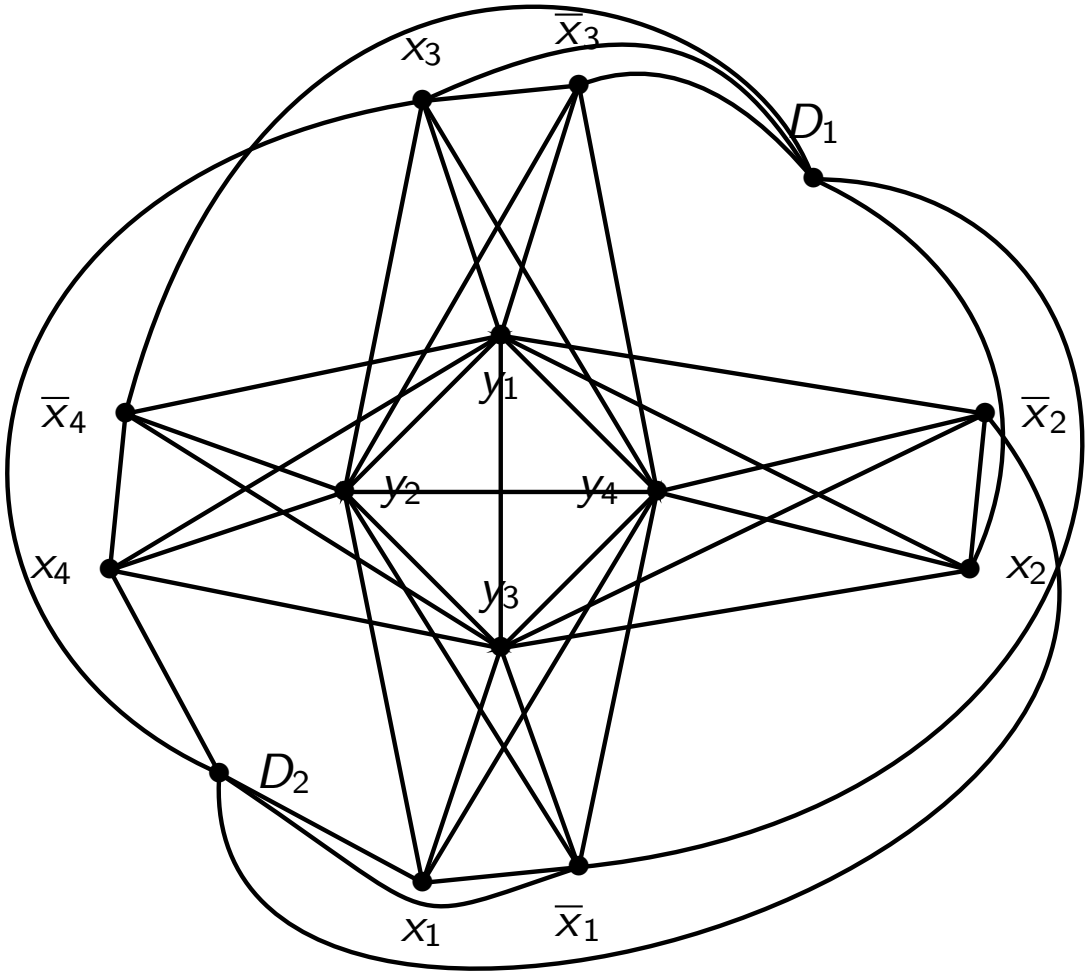


Primjer.

$$\Phi = \Phi(x_1, x_2, x_3, x_4) = (x_1 \vee \overline{x}_2 \vee x_4) \wedge (x_2 \vee \overline{x}_3 \vee \overline{x}_4)$$



Primjer. $\Phi = \Phi(x_1, x_2, x_3, x_4) = (x_1 \vee \overline{x}_2 \vee x_4) \wedge (x_2 \vee \overline{x}_3 \vee \overline{x}_4)$



- Koristimo n boja $T_1, \ldots, T_n$ za tačno i boju F za netačno.
- Treba nam n različitih boja da obojimo čvorove y_i
- $y_i \to T_i$ (tj. y_i bojimo sa T_i)
- Ne može $x_i \to T_j$ ili $\overline{x}_i \to T_j$ za $i \neq j$
- Ako $x_i \to T_i$ smatramo ga tačnim, tada $\overline{x}_i \to F$ i smatramo ga netačnim. Može biti i $x_i \to F$, $\overline{x}_i \to T_i$.
- Klauzula D_i može dobiti samo boju kojom je obojen neki od literala $x_j^{\delta_j} \in D_i$. Dok D_i ne može dobiti boju F, zbog $n \geq 4$ (spojen sa x_j i sa $\overline{x}_j$ za neko j).

- Dokažimo $\Phi \in 3SAT \Leftrightarrow (G, k) \in L_{gb}$

- $(\Rightarrow)$: Pretpostavljamo $\Phi \in 3SAT$ i za $x_i = \alpha_i \in \{0, 1\}$ važi $\Phi = 1$. Tada, $y_i \to T_i$; $x_i \to T_i$ i $\overline{x}_i \to F$ ako je $\alpha_i = 1$ dok $x_i \to F$ i $\overline{x}_i \to T_i$ ako je $\alpha_i = 0$; Iz $\Phi \in 3SAT \Rightarrow (\exists j) x_j^{\delta_j} \in D_i$ i $\alpha_j^{\delta_j} = 1$ pa je čvor $x_j^{\delta_j}$ obojen sa T_j. $D_i \to T_j$. Ovako bojenje je pravilno tj. $(G, k) \in L_{gb}$

- ($\Leftarrow$) : Boju kojom je obojen y_i nazovimo T_i a $(n+1)$-vu boju koju ne koristimo za bojenje čvorova y_i nazovimo F. Izaberimo $\alpha_i \in \{0,1\}$ sa: $\alpha_i = \begin{cases} 0 & , \text{ ako } x_i \to F \\ 1 & , \text{ ako } x_i \to T_i \end{cases}$

  D_i nije obojeno bojom F jer ima samo 3 literala sa kojima nije spojen a postoje bar 4 promenljive. Tj. postoji promjenljiva (čvor) tako da je D_i spojen i sa njom i sa njenom negacijom

  Znači D_i obojen sa T_j pa $(\exists j) x_j^{\delta_j} \in D_i$ i $x_j^{\delta_j}$ obojen sa T_j tj. $\alpha_j^{\delta_j} = 1$ i $D_i(\alpha_1, \ldots, \alpha_n) = 1$. Otuda je i $\Phi(\alpha_1, \ldots, \alpha_n) = 1$.

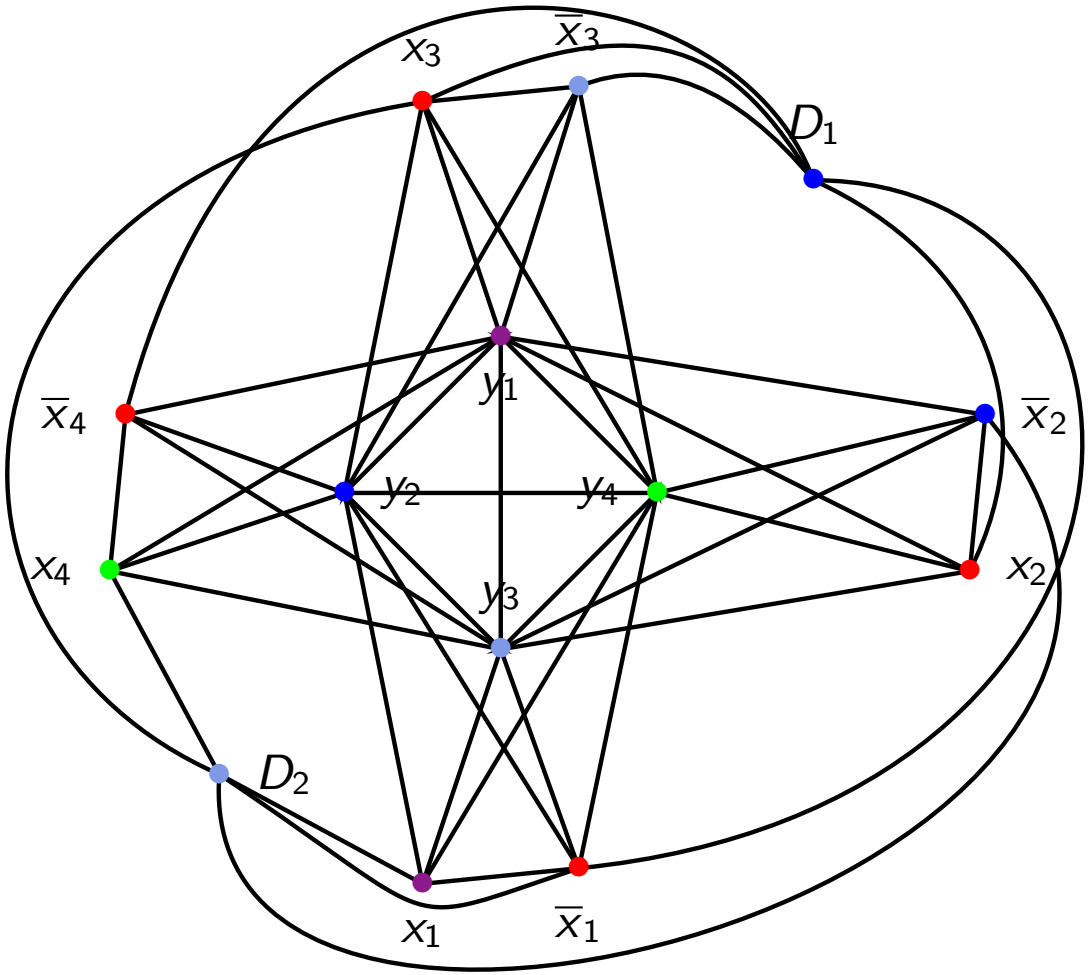
- Formula (iz primjera)

$$\Phi = \Phi(x_1, x_2, x_3, x_4) = (x_1 \vee \overline{x}_2 \vee x_4) \wedge (x_2 \vee \overline{x}_3 \vee \overline{x}_4)$$

  je zadovoljiva. Npr. $x_1 = 1, x_2 = 0, x_3 = 0, x_4 = 1$.

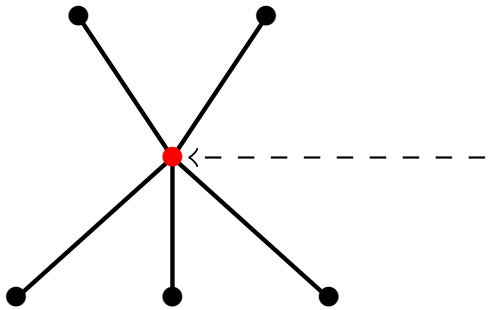
  Bojenje koje odgovara ovoj valuaciji je prikazano na slici. Koristimo boje: T_1, T_2, T_3, T_4, F.

$x_1 = 1, x_2 = 0, x_3 = 0, x_4 = 1$ T_1, T_2, T_3, T_4, F



Bojenje grana grafa

- $G = (V, E)$;
- Bojimo grane, tako da su susjedne grane obojene različitom bojom.
- $g : E \to \{1, \ldots, k\}$ i $(\forall (u, v), (v, w) \in E)\, g(u, v) \neq g(v, w)$
- Stepen čvora i stepen grafa



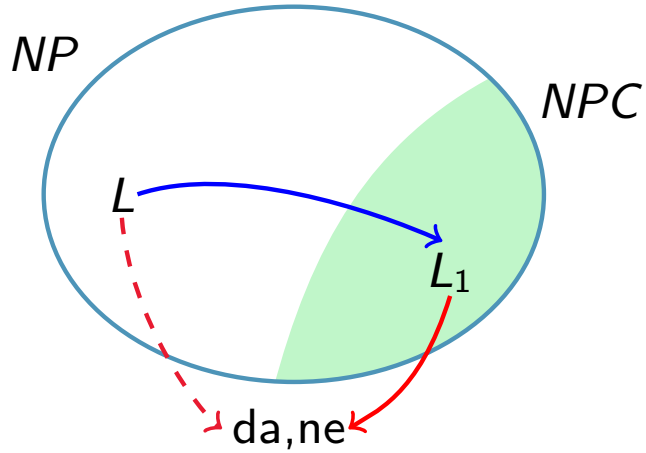
Stepen čvora je 5.
Da obojimo grane treba nam 5 boja.

- **Teorema**. Ako je stepen grafa m onda se grane grafa mogu pravilno obojiti sa $(m + 1)$ bojom.
- Zadatak. Da li se grane grafa mogu pravilno obojiti sa k boja? Zadatak je *NPC*.
- Jedini "težak" slučaj je $k = m$. Jer za $k < m$ odgovor je "NE", a za $k \geq m + 1$ odgovor je "DA".

Još dva *NPC* zadatka

- Neka je dat graf $G = (V, E)$.
- Za podgraf $G' = (V', E')$ grafa G (tj. $V' \subseteq V$ i $E' \subseteq E \cap V' \times V'$) za koji važi $E' = V' \times V'$ i $V' = k$ kažemo da je k-klika grafa G. Drugim rječima k-klika je potpun podgraf sa k čvorova.
- Zadatak (o k-kliki). Za dati graf G i broj k da li postoji k-klika grafa G.
- **Teorema**. Zadatak o k klici je $NP-$kompletan.
- Niz čvorova $x_1, x_2, \ldots, x_k$ za koji je $(x_i, x_{i+1}) \in E$ nazivamo put u grafu G. Početak puta je u čvoru x_1 a kraj u čvoru x_k. Ako se početak i kraj puta poklapaju (tj. $x_1 = x_k$) onda kažemo da je put zatvoren.
- Put se naziva Hamiltonov ako sadrži svaki čvor grafa tačno jednom. Zatvoreni put kod koga kad izostavimo poslednji čvor dobijamo Hamiltonov put naziva se zatvoren Hamiltonov put.
- Zadatak (o trgovačkom putniku-Hamiltonovom putu). Da li u datom grafu G postoji Hamiltonov put?
- **Teorema**. Zadatak o trgovačkom putniku je $NP-$kompletan.

Još o klasama složenosti



Ako za jedan *NPC* zadatak imamo polinomijalni algoritam. Onda svaki NP zadatak ima polinomijalni algoritam.

- $t : \mathbb{N} \to \mathbb{N}$, $\quad \mathrm{DTIME}(t(\cdot)) =$
  $\{L : (\exists DTM\ \mathfrak{M})(\exists c \in \mathbb{R}^+) L = L_{\mathfrak{M}}$ i $T(\mathfrak{M}, n) \leq c \cdot t(n)\}$
- $P = \bigcup_{c \in \mathbb{N}} \mathrm{DTIME}(n^c)$
- Slično: $\mathrm{NTIME}(t(\cdot))$, *NP*
- $L \in NP$ ako
  $(\exists DTM\ \mathfrak{M})(\exists\, p(\cdot), q(\cdot)$-polinomi) tako da važi:
  $x \in L \quad \Leftrightarrow \quad (\exists\, u \in A')\ |u| \leq p(|x|), (x, u) \in L_{\mathfrak{M}}, T(\mathfrak{M}, |x|) \leq q(|x|)$

- $ExpTime = \bigcup_{c \in \mathbb{N}} \mathrm{DTIME}(2^{n^c})$ oznake: *EXP*, *EXPTIME*,...

- Slično: *NExpTime*; *DExpTime* ili $2ExpTime$, 2^{2^n}; $3ExpTime$,...
- $s : \mathbb{N} \to \mathbb{N}$, $\quad \text{SPACE}(s(\cdot)) =$
  $\{L : (\exists DTM\ \mathfrak{M})(\exists c \in \mathbb{R}^+)L = L_{\mathfrak{M}}$ i $S(\mathfrak{M}, n) \leq c \cdot s(n)\}$
- $PSPACE = \bigcup_{c \in \mathbb{N}} \text{SPACE}(n^c)$
- $PSPACE = \{L : (\exists DTM\ \mathfrak{M})(\exists p\text{-pol.})L = L_{\mathfrak{M}}$ i $S(\mathfrak{M}, n) \leq p(n)\}$
- Slično: NSPACE$(s(\cdot))$, *NPSPACE*, *ExpSpace*, *NExpSpace*, $2ExpSpace$, ...

- $P \subseteq NP \subseteq PSPACE \subseteq NPSPACE \subseteq ExpTime$

  $ExpTime \subseteq NExpTime \subseteq ExpSpace \subseteq NExpSpace \subseteq 2ExpTime \cdots$
- Zna se: $P \neq ExpTime$ i $PSPACE = NPSPACE$
- **Teorema**.(Savitch) $PSPACE = NPSPACE$.

  Preciznije: NSPACE$(f(n)) \subseteq$ DSPACE$(f^2(n))$ za $f \in \Omega(\log n)$.
- Znači da nedeterminizam ne utiče na prostor.

- Ako je C-klasa složenosti onda možemo definisati i klase:

  $coC = \{L \ : \ L^c \in C\}; \quad L^c = A' \setminus L,$

  $C-$težak,

  $C-$kompletan.
- SAT^c-je skup formula koje nisu zadovoljive (tj. čije su negacije tautologije)
- SAT$^c \in coNP$; $\quad \psi \in \text{SAT} \Leftrightarrow \exists\alpha_1\exists\alpha_2\cdots\exists\alpha_n\, \psi(\alpha_1,\ldots,\alpha_n) = 1$

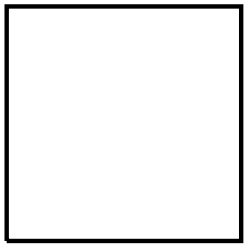
  $\neg\psi \in \text{SAT}^c \Leftrightarrow \forall\alpha_1\forall\alpha_2\cdots\forall\alpha_n\, \psi(\alpha_1,\ldots,\alpha_n) = 1$
- $QBF \quad : \quad \Psi = Q_1x_1Q_2x_2\cdots Q_nx_n\, \psi(x_1,\ldots,x_n), \quad Q_i \in \{\forall,\exists\}.$
- $TQBF$ istinite QBF (kvantifikovane Bulove formule) tj. $TQBF = \{\Psi \in QBF \ : \ \Psi = 1\}$
- **Teorema**. Jezik $TQBF$ je $PSPACE$-kompletan.
- $coNP \subseteq PSPACE$, vjeruje se $NP \neq coNP$
- Zadatak. Dokazati $P \subseteq NP \cap coNP$.

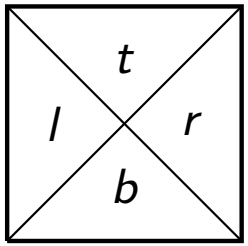
- Ako je C-deterministička klasa složenosti onda je $coC = C$
- Znači da determinizam ne utiče na co klasu.
- **Teorema**. Ako je $P \neq NP$ onda postoji jezik $L \in NP \setminus P$ takav da $L \notin NPC$.

- Neka je dat jezik O (orakl-oracle). TM sa čarobnjakom O je mašina koja u jednom koraku može da nam odgovori da li riječ na traci (obično je ona na posebnoj traci) pripada jeziku O.
- Pitanje da li je $w_o \in O$ postavljamo kad se nađemo u nekom posebnom stanju npr. q_{upit} a odgovor dobijamo tako što mašina prelazi u neko stanje q koje je u skupu DA odnosno NE stanja.
- Klasa C^O je u stari klasa C sa čarobnjakom O. Odnosno zadaci se rješavaju na TM sa čarobnjakom O i imaju istu složenost kao zadaci u klasi C.
- Primjer. P^O je skup jezika koji mogu biti riješeni na DTM sa čarobnjakom O za polinomijalno vrijeme.
- Zadatak. Ako je $L \in P$ dokazati da je $P^L = P$.

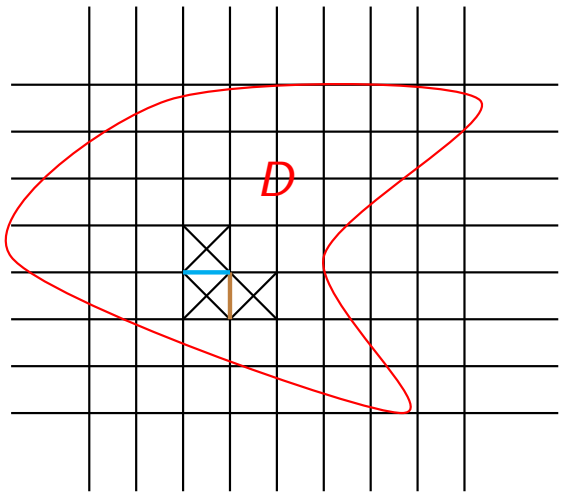
Problem popločavanja

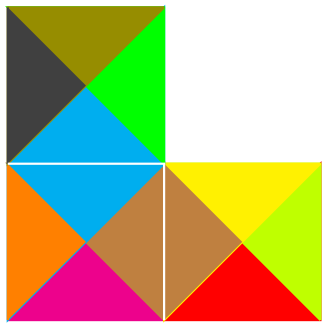
- B -skup boja
- Ploča - kvadrat 1×1 ne dopušta simetrije i rotacije



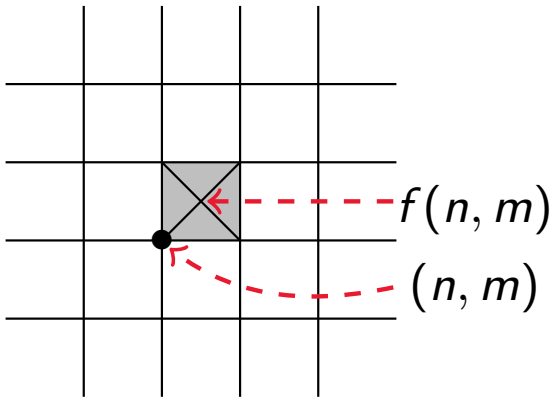



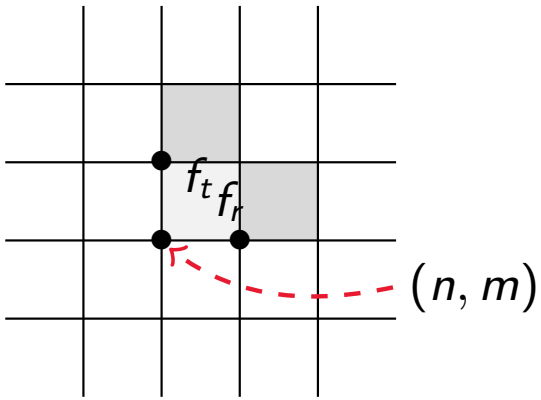
- Matematički: $t = (l, b, r, t) \in B^4$ je ploča.
- $\tau \subseteq B^4$ skup (tipova) ploča.
- Opšti problem popločavanja: $\tau, \quad D \subseteq \mathbb{Z} \times \mathbb{Z}$.
-



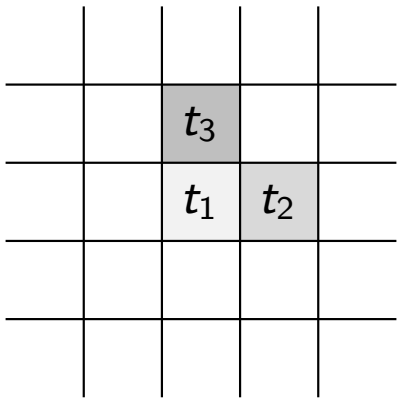


- Matematički: $f : D \to \tau, \quad f = (f_l, f_b, f_r, f_t)$
- $f_r(n, m) = f_l(n + 1, m); \quad f_t(n, m) = f_b(n, m + 1)$
-





- Domino problem: $H \subseteq \tau \times \tau \; V \subseteq \tau \times \tau$
-



$(t_1, t_2) \in H$

$(t_1, t_3) \in V$

Primjer:

$(t_1, t_2), (t_1, t_3) \in H$

$(t_4, t_2) \in H, \; (t_4, t_3) \notin H$

Tri problema popločavanja

(1) $n \times n$ popločavanje: Dat je broj n i skup ploča τ. Neka je K_n kvadrat dimenzija $n \times n$ čije su ivice obojene bijelom bojom. Da li se kvadrat K_n može pravilno popločati sa pločama iz τ?

* $K_n = \{(i,j) \ : \ 0 \leq i,j < n\}$ -kvadrat dimenzija $n \times n$. $f : K_n \to \tau$?
* $L_K = \{(\tau, n) \ : \ (\exists f) f : K_n \to \tau$ pravilno bojenje$\}$

(2) n popločavanje koridora: Dat je broj n i skup ploča τ. Da li postoji broj m tako da se pravougaonik P_n^m dimenzija $n \times m$ čie su ivice obojene bijelom bojom može pravilno popločati sa pločama iz τ?

* $L_P = \{(\tau, n) \ : \ (\exists m \in \mathbb{N})(\exists f) f : P_n^m \to \tau$ pravilno bojenje$\}$

(3) $n \times n$ naizmenično popločavanje(popločavanje sa dva igrača): Dat je broj n i skup ploča τ. Neka je K_n kvadrat dimenzija $n \times n$ čije su ivice obojene bijelom bojom. Dva igrača A(lisa) i B(ob) naizmenično biraju ploču i pravilno postavljaju na sledeće polje. Igrač A se trudi da poploča K_n a igrač B želi da ga spriječi u tome. Da li A popeđuje (tj. poploča K_n) pod uslovom da A i B imaju najbolje strategije?

Teorema. (o složenosti problema popločavanja)
Neka je $n \in \mathbb{N}$ i τ skup ploča ($|\tau|$ linearno ograničen sa n).

a) Ako je n zadato unarno. Tada,
   1. Problem $n \times n$ popločavanja je *NP*-kompletan (tj. $L_K \in NPC$).
   2. Problem n popločavanja koridora je *PSPACE*-kompletan.
   3. Problem $n \times n$ naizmeničnog popločavanja je *ExpTime*-kompletan.

b) Ako je n zadato binarno. Tada,
   1. Problem $n \times n$ popločavanja je *NExpTime*-kompletan.
   2. Problem n popločavanja koridora je *ExpSpace*-kompletan.
   3. Problem $n \times n$ naizmeničnog popločavanja je 2*ExpTime*-kompletan.

- Postoje i drugi problemi popločavanja. Navedeni problemi su odlučivi, a postoje problemi popločavanja koji su neodlučivi.
- Unarni zapis $1000_{(10)}$ je sa $\underbrace{11\cdots1}_{1000}$, a binarni 1111101000.
- Unarni zapis broje n ima $O(n)$ cifara a binarni $O(\log n)$ cifara.
- Za ulaz veličine $k = \log n$ i broj koraka $f(n)$ dobijamo složenost $g(k) = f(2^k)$. Npr. za $f(n) = n^2$ dobijamo $g(k) = (2^k)^2 = 2^{2k}$.

Dokaz. (Teoreme o složenosti problema popločavanja)
Dokazaćemo samo *a*) 1. tj. $L_K \in NPC$ kad je n zadato unarno.
Treba dokazati: 1^o) $L_K \in NP$ i 2^o) $L_K \in NPH$.

1^o) $L_K \in NP$

Algoritam. Ulaz: τ- skup ploča, i broj n (veličina ulaza).
Izlaz: DA/NE.
Metod: I-korak. Ponavljaj dok ne popločaš sva polja kvadrata: nedeterministički izaberei jednu ploču i stavi je na sledeće polje kvadrata.
II-korak. Pregledaj sve ivice postavljenih ploča (jednu po jednu) i provjeri da li ima istu boju kao ivica njoj susjedne ploče. Ako se sve boje slažu odgovori DA, inače odgovori NE.

Složenost. I-korak. Nederministički izbor ploče košta nas $O(n)$. Treba popločati n^2 polja pa je složenost $O(n^3)$. II-korak. Svaka ploča ima 4 ivice pa treba pregledato $O(n^2)$ ivica, što daje složenost $O(n^2)$. Algoritam je nedeterministički i radi za polinomijalno vrijeme pa je $L_K \in NP$.

2°) $L_K \in NPH$

Neka je $L \in NP$ proizvoljan jezik. Treba dokazati $L \lhd L_K$.

$L \in NP \Leftrightarrow \exists(\text{NDTM } \mathfrak{M})(\exists d \in \mathbb{N}) L = L_{\mathfrak{M}}$ i $T(\mathfrak{M}, n) < n^d$

Za ulaz x, $|x| = n$ (n dovoljno veliki broj) važi:

$x \in L \Leftrightarrow x \in L_{\mathfrak{M}}$ i $T(\mathfrak{M}, |x|) < n^d \Leftrightarrow \exists \mathfrak{C} = C_0, \ldots, C_{n^d-1}, \ldots$ izračunavanje koje odgovara x i C_{n^d-1} završna konf. $\Leftrightarrow$ $\exists \mathfrak{D}(\mathfrak{M}, n^d) = C_0, \ldots, C_{n^d-1}$ vrem. prostorni diag. dimenzija $n^d \times n^d$ $\Leftrightarrow (\tau, n^d) \in L_K$.

Za sve ekvivalencije, osim poslednje, od ranije znamo da važe.

Dokažimo posledniju (naznačena crvenom bojom) ekvivalenciju.

Na osnovu TM $\mathfrak{M}$ i x konstruišemo τ tako da $(\tau, n^d + 1)$ kodira vremensko prostorni dijagram $\mathfrak{D}(\mathfrak{M}, n^d)$.

Treba da kodiramo:

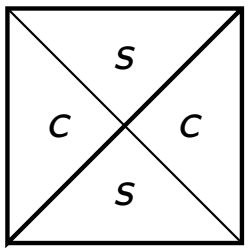
1) Ulaz - tj. polaznu konfiguraciju;
2) Rad mašine - tj. prelaz sa konfiguracije na konfiguraciju;
3) Završnu konfiguraciju.

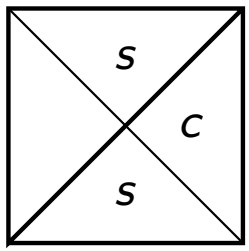
Kodiramo rad mašine

Prelaz sa konfiguracije na konfiguraciju je u skladu sa programom $\mathcal{P}$.

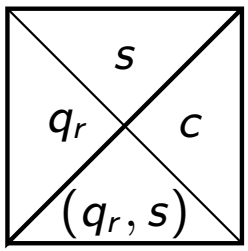
Normalizacija TM: $(\forall q \in Q)$ ako $(-,-,q,-,r),(-,-,q,-,l) \in \mathcal{P}$ onda q menjamo sa ekvivalentnim stanjima $q_l \in Q_L$ i $q_r \in Q_R$ a navedene komande redom sa $(-,-,q_r,-,r),(-,-,q_l,-,l)$
Novo $Q = Q_0 \cup Q_R \cup Q_R$

I) $\forall s \in \overline{A} = A \cup \{a_0\}$

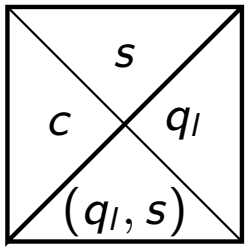


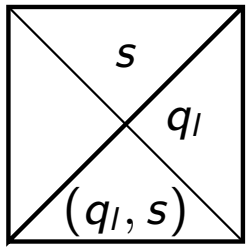


II) $\forall s \in \overline{A}$, $\forall q_r \in Q_R$

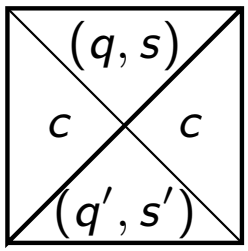


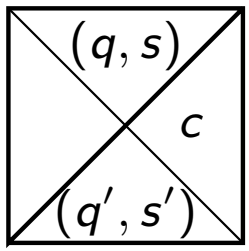
III) $\forall s \in \overline{A},\ \forall q_l \in Q_L$



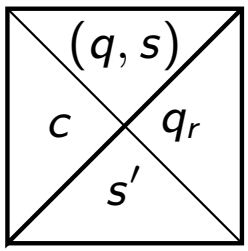


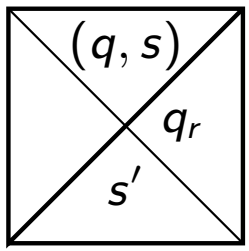
IV) $\forall (q, s, q', s', o) \in \mathcal{P}$



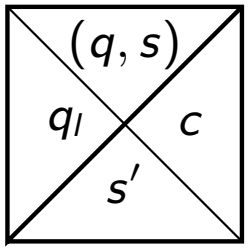


V) $\forall (q, s, q_r, s', r) \in \mathcal{P},\ q_r \in Q_R$

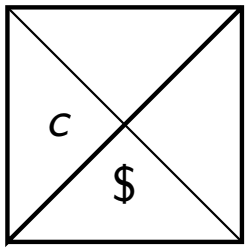


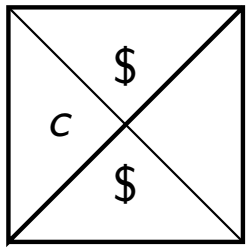


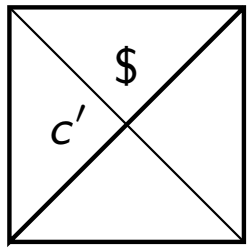
VI) $\forall (q, s, q_l, s', l) \in \mathcal{P},\ q_l \in Q_L$



VII) Uvodimo 3 specijalne ploče zbog desne ivice kvadrata

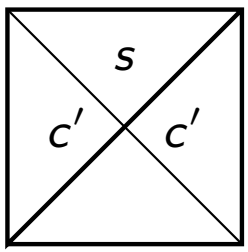


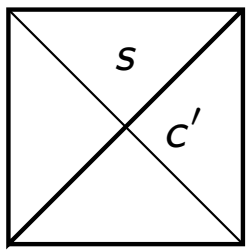




Kodiramo završnu konfiguraciju

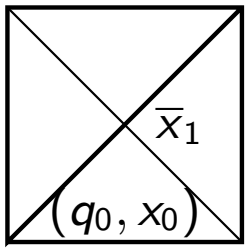
VIII) $\forall s \in \overline{A}$

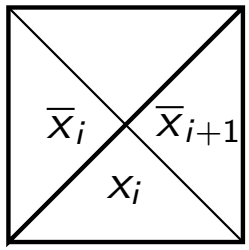


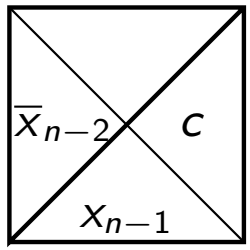


Kodiramo polaznu konfiguraciju

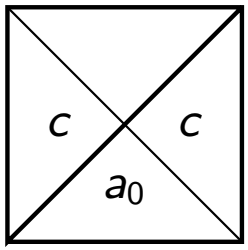
IX) Kodiramo ulaz $x = x_0 x_1 \cdots x_{n-1}$ $\forall x_i$



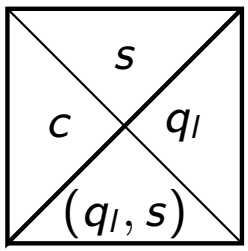


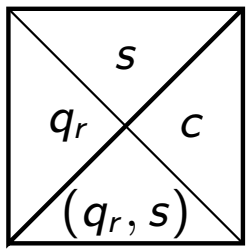


X) Kodiramo prazna polja u polaznoj konfiguraciji



$*$) Normalizacija nam je neophodna zbog npr. ploča II) i III)





- Ploče smo prikazali vizuelno. U suštini mi ih zapisujemo kao četvorke, npr. za ploče pod I) i II) (c, s, c, s), $(\ , s, c, s)$, $(q_r, (q_r, s), c, s)$ i sl.
- Samo ploče pod IX) zavise od ulaza. Ostale ploče zavise samo od TM (tj. azbuke, skupa stanja, programa). Kako je TM fiksirana (ne zavisi od ulaza) mi imamo konaňo mnogo ploča. Pa njihova konstrukcija zahtjeva $O(1)$ vremena.
- Konstrukcija ploča pod IX) zahtjeva $O(n)$ vremena.
- Treba dokazati $x \in L \Leftrightarrow (\tau, n^d + 1) \in L_K$.
  $(\Rightarrow):\quad x \in L \Rightarrow \exists \mathfrak{D}(\mathfrak{M}, n^d + 1) = C_0, \ldots, C_{n^d-1}$
  Neka su u polju j u konfiguracijama C_{i-1} i C_i redom upisani simboli s i s' tada u kvadrat na polje (i, j) stavljamo ploču (c, s', c, s) i sl.
  Ovako popločavanje je pravilno tj. $(\tau, n^d + 1) \in L_K$.
  $(\Leftarrow):\quad (\tau, n^d + 1) \in L_K \Rightarrow$ postoji pravilno popločavanje kvadrata.
  Neka je u polje (k, m) prvi put stavljena ploča $(c', \ , \ , \$)$ (npr. u polje (n^d, n^d)). Tada na osnovu ploča u pravougaoniku $k \times m$ možemo napraviti vremensko prostorni dijagram. Npr. ako je na polje (i, j) stavljena ploča (X, Y, Z, s) onda u C_i na polju j pišemo s.

Polinomijalna hijerarhija

- $L \in NP$ ako $(\exists DTM\ \mathfrak{M})(\exists\, p(\cdot), q(\cdot)\text{-polinomi})$ tako da važi:
  $x \in L \quad \Leftrightarrow \quad (\exists\, u \in A')\ |u| \leq p(|x|), (x,u) \in L_{\mathfrak{M}}, T(\mathfrak{M}, |x|) \leq q(|x|)$

- Za $DTM\ \mathfrak{M}$ govorićemo da je polinomijalna DTM i označavati sa DTM^p ako $(\exists\, p(\cdot)\text{-polinom})\ T(\mathfrak{M}, n) \leq p(n)$. Slično $NDTM^p$.
- Za $u \in A'$ koristimo oznaku $u^{<k}$ da označimo da je $|u| \leq k$.

- $L \in \Sigma_1^p = NP$ ako $(\exists DTM^p\ \mathfrak{M})(\exists\, p(\cdot)\text{-polinom})$ tako da važi:
  $x \in L \quad \Leftrightarrow \quad (\exists\, u^{<p(|x|)} \in A')\,(x,u) \in L_{\mathfrak{M}}$.
- $L \in \Sigma_2^p$ ako $(\exists DTM^p\ \mathfrak{M})(\exists\, p(\cdot)\text{-polinom})$ tako da važi:
  $x \in L \quad \Leftrightarrow \quad (\exists\, u^{<p(|x|)} \in A')(\forall\, v^{<p(|x|)} \in A')\,(x,u,v) \in L_{\mathfrak{M}}$.

- $L \in \Sigma_k^p$ ako $(\exists DTM^p\ \mathfrak{M})(\exists\, p(\cdot)\text{-polinom})$ tako da važi:
  $x \in L \quad \Leftrightarrow$
  $(\exists\, u_1^{<p(|x|)} \in A')(\forall\, u_2^{<p(|x|)} \in A') \cdots (Q_i\, u_i^{<p(|x|)} \in A')$
  $(x, u_1, u_2, \ldots, u_i) \in L_{\mathfrak{M}}$, gdje je $Q_{2j} = \forall$, $Q_{2j+1} = \exists$.

- $\Pi_i^p = co\Sigma_i^p$ tj. u definiciji Σ_i^p treba zamjeniti mjesta $\forall, \exists$; specijalno $\Pi_1^p = coNP$.
- $\Sigma_i^p \subseteq \Pi_{i+1}^p \subseteq \Sigma_{i+2}^p$
- Polinomijalna hijerarhija je skup $PH = \bigcup_{i \in \mathbb{N}} \Sigma_i^p = \bigcup_{i \in \mathbb{N}} \Pi_i^p$.

- $\Sigma_i SAT = \{\psi \in TQBF : \psi = \exists x_1 \forall x_2 \cdots Q_i x_i \varphi(x_1, \ldots, x_i)\}$.
  φ je Bulova formula a $Q_{2j} = \forall$, $Q_{2j+1} = \exists$.
- Slično definišemo $\Pi_i SAT$.

- **Teorema**. a) Za svako $i \in \mathbb{N}$, ako je $\Sigma_i^p = \Pi_i^p$ onda je $PH = \Sigma_i^p$.
  b) Ako je $P = NP$ onda je $PH = P$.
  c) Jezik $\Sigma_i SAT$ je Σ_i^p-kompletan.
  d) Ako postoji jezik L koji je PH-kompletan onda postoji $k \in \mathbb{N}$ tako da je $PH = \Sigma_k^p$.
  e) Jezik $\Pi_i SAT$ je Π_i^p-kompletan.

- $PH \subseteq PSPACE$.

Naizmenična (Alternating) Tjuringova mašina - ATM

ATM je četvorka $\mathcal{M} = (Q, \Sigma, q_0, \delta)$, gdje je

1. $Q = Q_\exists \uplus Q_\forall \uplus \{q_a\} \uplus \{q_r\}$ konačan skup stanja sastoji se od stanja postojanja $Q_\exists$, univerzalnih stanja $Q_\forall$, jednog stanja prihvatanja q_a i stanja neprihvatanja q_r;
2. $\Sigma = \{a_0, a_1, \ldots, a_t\}$ -azbuka, gdje je a_0 prazan simbol;
3. $q_0 \in Q_\exists \uplus Q_\forall$ početno stanje;
4. δ funkcija tranzicije (daje nam dvije alternativne tranzicije)

$$\delta \colon (Q \setminus \{q_a, q_r\}) \times \Sigma \to (Q \times \Sigma \times \{l, r\})^2$$

- Za $\delta(q, a) = ((q_1, a_1, d_1), (q_2, a_2, d_2))$ pišemo $\delta_i(q, a) = (q_i, a_i, d_i)$, $i = 1, 2$.
- Konfiguracija ATM $\mathcal{M}$ je riječ wqw', gdje je $q \in Q$, $w, w' \in \Sigma^*$.
- Ako je $q \in \{q_a, q_r\}$ onda je wqw' završna konfiguracija.

- Izračunavanje ATM $\mathcal{M}$ na riječi $w \in \Sigma^*$ je niz konfiguracija $c_0, c_0, \ldots$ gdje je $c_0 = q_0 w$ (početna konfiguracija) a za svako $i \geq 0$, c_{i+1} je konfiguracija koja se dobija iz konfiguracije c_i primjenom funkcije tranzicije tj. c_{i+1} je sljedbenica od konfiguracije c_i.

- Završna konfiguracija je prihvatljiva ako je oblika wq_aw'.
  Za konfiguraciju $c = wqw'$ koja nije završna, definišemo (induktivno) da je c prihvatljiva ako je $q \in Q_\exists$ i bar jedna konfiguracija koja je sljedbenica od c je prihvatljiva, ili je $q \in Q_\forall$ i sve (obije) konfiguracije koje su sljedbenice od c su prihvatljive.

- Kažemo da ATM $\mathcal{M}$ prihvata ulaz w ako je polazna konfiguracija $q_0 w$ prihvatljiva.

- Sa $L_\mathcal{M}$ označavamo jezik koji prihvata $\mathcal{M}$,
  tj. $L_\mathcal{M} = \{w \in \Sigma^* \mid \mathcal{M} \text{ prihvata } w\}$.

- Kažemo da je K vremenska složenost ATM $\mathcal{M}$ na riječi w ako sva izračunavanja ATM $\mathcal{M}$ na riječi w dostižu završnu konfiguraciju u najviše K koraka.

- Slično kao i ranije možemo definistai vremensku i prostornu složenost (najgori i prosječni slučaj) na ulazu veličine n.

- $t : \mathbb{N} \to \mathbb{N}, \quad \text{ATIME}(t(\cdot)) =$
  $\{L : (\exists ATM\ \mathcal{M})(\exists c \in \mathbb{R}^+) L = L_{\mathcal{M}} \text{ i } T(\mathcal{M}, n) \leq c \cdot t(n)\}$

- $AP = \bigcup_{c \in \mathbb{N}} \text{ATIME}(n^c)$

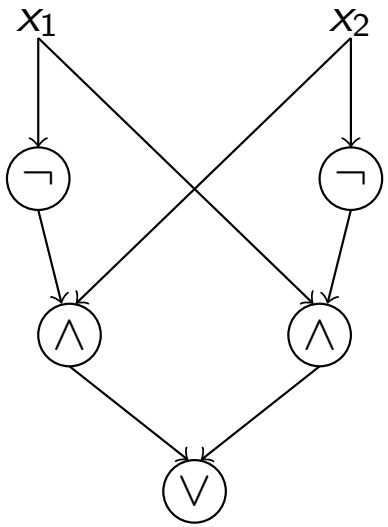
- $\Sigma_i \text{TIME}(t(\cdot))$ je skup jezika iz klase $\text{ATIME}(t(\cdot))$ i kod kojih $ATM\ \mathcal{M}$ ima početno stanje iz $Q_\exists$ i u svakom izračunavanju izvrši najviše $i-1$ promjenu stanja iz skupa $Q_\exists$ u $Q_\forall$ ili obrnuto.

- $\Sigma_i^p = \bigcup_{c \in \mathbb{N}} \Sigma_i \text{TIME}(n^c)$; $AP = PSPACE$; $APSPACE = EXPTIME$.

Logička kola

- Logičko kolo C_n sa n-ulaza i jednim izlazom je usmjereni aciklični graf sa n-izvora i jednim ušćem. Svi čvorovi sem izvora su označeni sa $\wedge, \vee, \neg$. Veličina od C_n, u oznaci $|C_n|$, je broj čvorova u kolu.
- Neka je $x \in \{0,1\}^n$. Izlaz kola C_n na ulazu x, u oznaci $C_n(x)$ se definiše na prirodan način.
- $t : \mathbb{N} \to \mathbb{N}$,
  $SIZE(t(\cdot)) = \left\{ L : (\exists \{C_n\}_{n\in\mathbb{N}}) \left((x \in L \Leftrightarrow C_n(x) = 1) \text{ i } |C_n| \leq t(n) \right) \right\}$
- $P_{/poly} = \bigcup_{c\in\mathbb{N}} SIZE(n^c)$; $\quad P \subseteq P_{/poly}$
-

C_2 :



$C_2(1,0) = 1;\ C_2(1,1) = 0$

$C_2(x_1, x_2) = (\neg x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)$