Maatriksite korrutamise kaudu programmianalüüsi

Näited analüüsidest

Reaching Definitions

Ülesanne

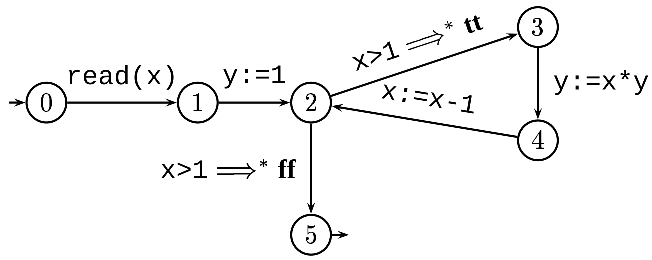
Leida iga programmipunkti kohta, millistes programmipunktides saab iga nähtav muutuja olla viimati üle kirjutatud.

Näide

Olgu meil programm

`read(x); y:=1; while x>1 do (y:=x*y; x:=x-1).`

Elementaaroperatsioonideks jagunemine võiks meie analüüsi seisukohalt käia järgmiselt:

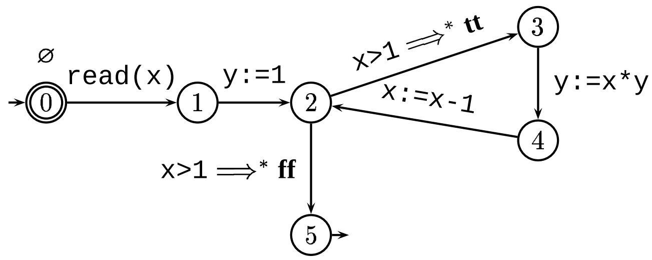


Siis RD analüüsi tulemus on

$$
\begin{array}{c|l}
0 & \{\} \\
1 & \{\mathsf{x} \mapsto \{0\}\} \\
2 & \{\mathsf{x} \mapsto \{0,4\}, \mathsf{y} \mapsto \{1,3\}\} \\
3 & \{\mathsf{x} \mapsto \{0,4\}, \mathsf{y} \mapsto \{1,3\}\} \\
4 & \{\mathsf{x} \mapsto \{0,4\}, \mathsf{y} \mapsto \{3\}\} \\
5 & \{\mathsf{x} \mapsto \{0,4\}, \mathsf{y} \mapsto \{1,3\}\}
\end{array}
$$

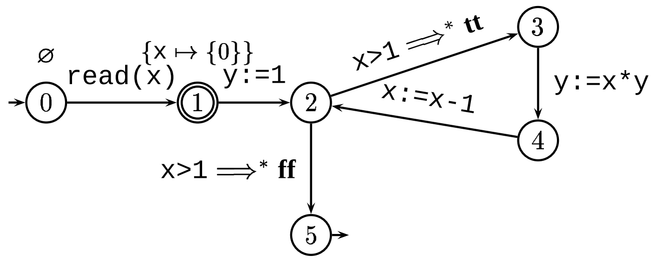
Süstemaatiline leidmine

Tee tsükli 0-kordse läbimisega:



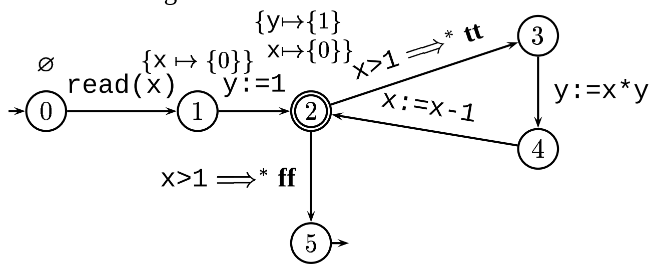
Süstemaatiline leidmine

Tee tsükli 0-kordse läbimisega:



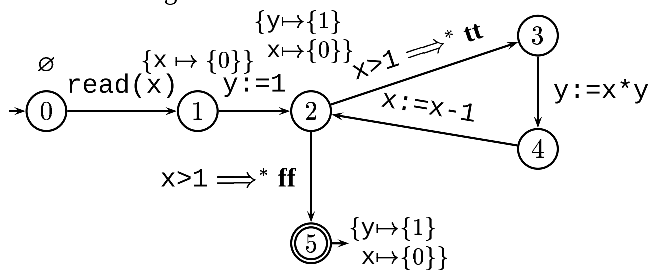
Süstemaatiline leidmine

Tee tsükli 0-kordse läbimisega:



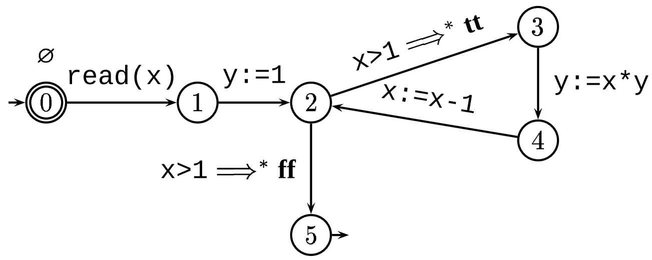
Süstemaatiline leidmine

Tee tsükli 0-kordse läbimisega:



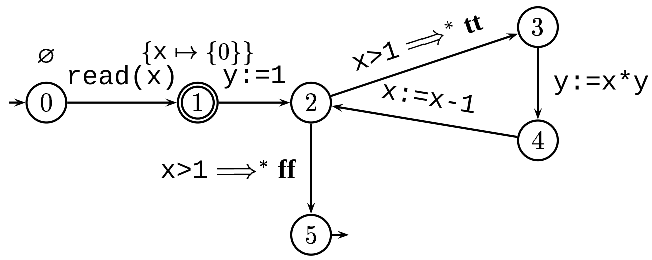
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



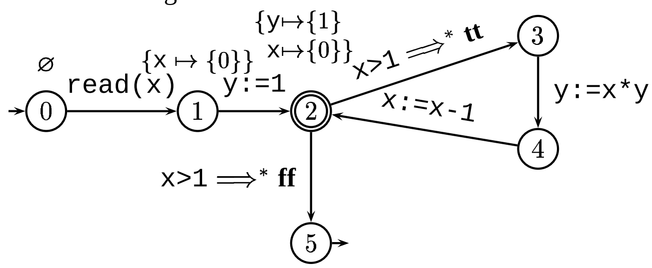
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



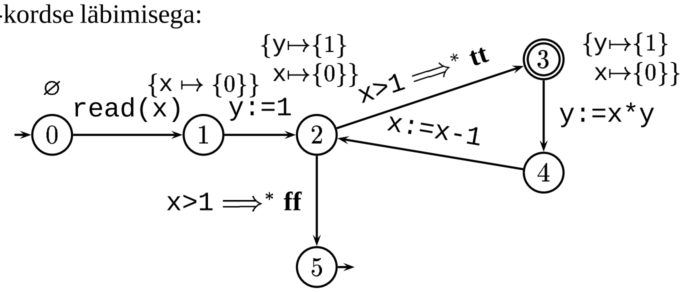
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



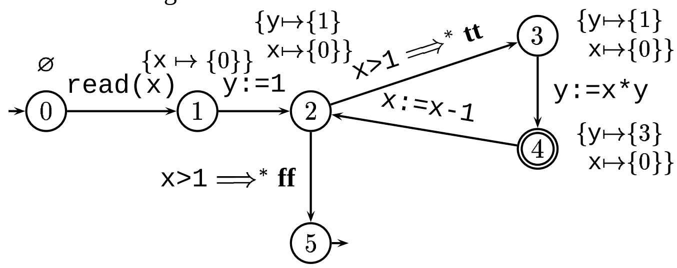
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



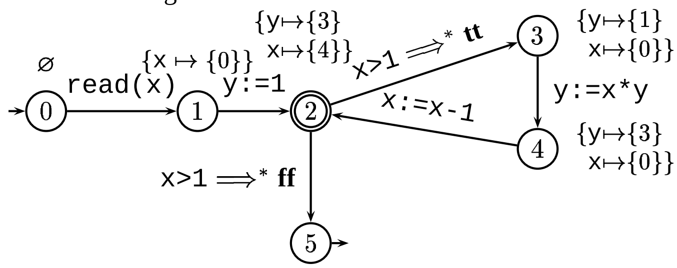
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



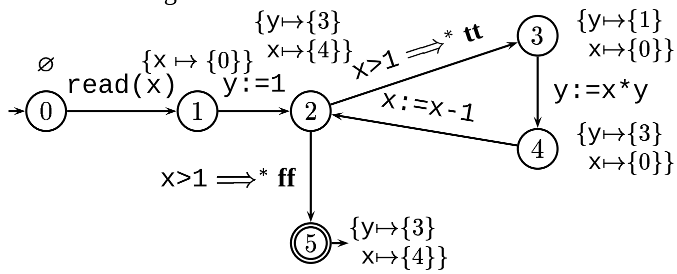
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



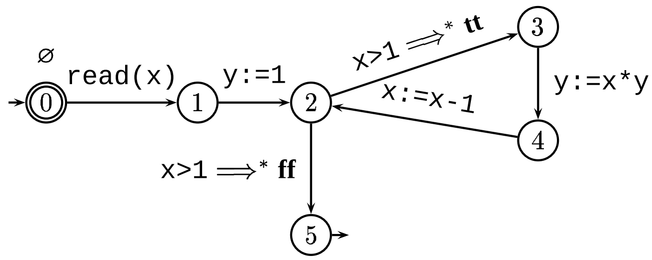
Süstemaatiline leidmine

Tee tsükli 1-kordse läbimisega:



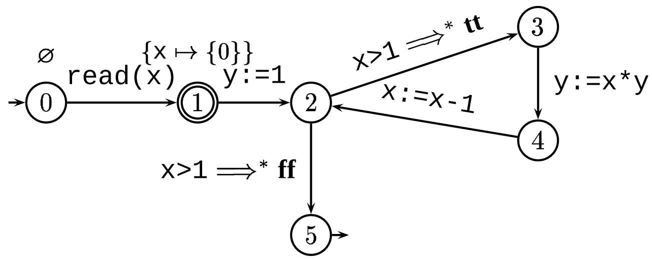
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



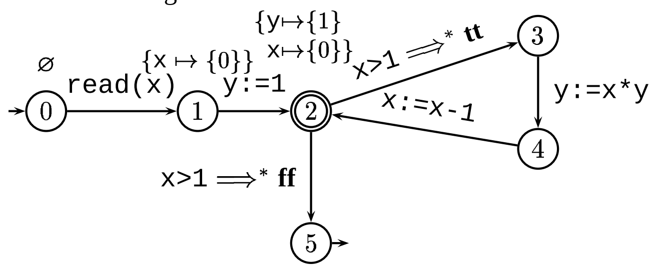
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



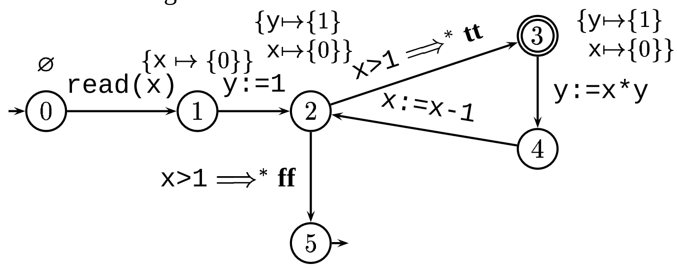
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



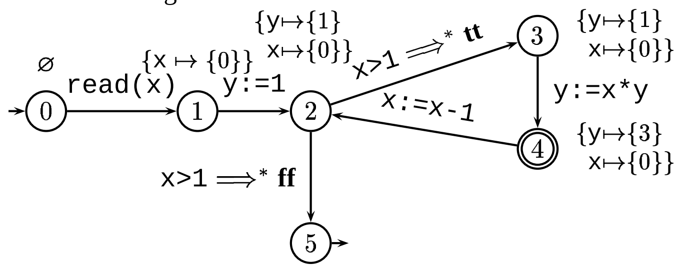
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



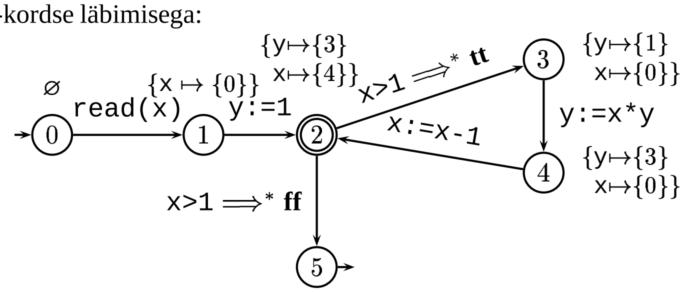
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



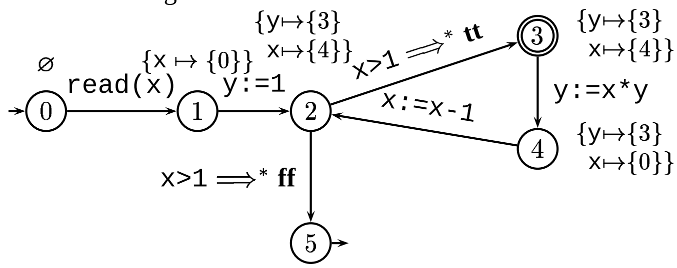
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



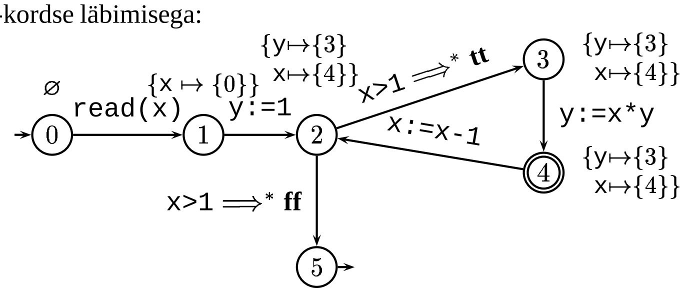
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



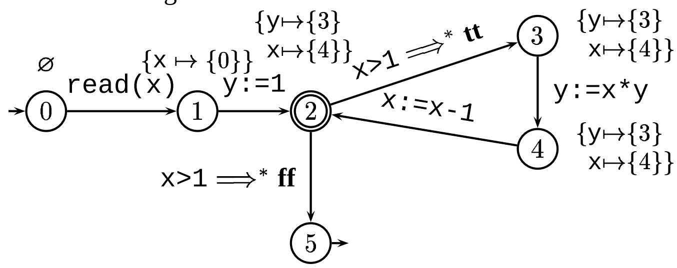
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



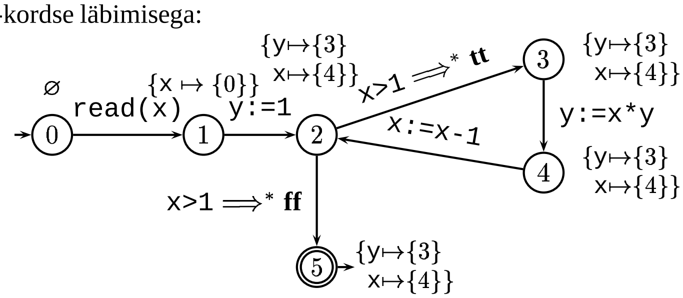
Süstemaatiline leidmine

Tee tsükli 2-kordse läbimisega:



Süstemaatiline leidmine

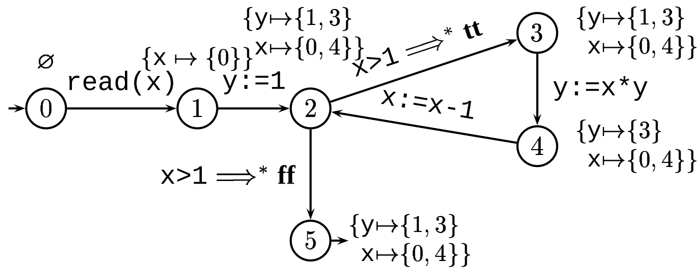
Tee tsükli 2-kordse läbimisega:



Süstemaatiline leidmine

Sellega on juhud ammendatud, sest punkti 2 väärtus, millest lähtudes tsükli väärtused arvutatakse, ei muutunud.

Kokkuvõttes saame



Live Variables

Elav

Muutujat x nimetatakse **elavaks programmipunktis** p, kui juhtimisvoograafis leidub selline ahel $v_0, \ldots, v_l$, kus $v_0 = p$, et tippudes v_i, $i < l$, muutujat x üle ei kirjutata ja tipus v_l toimub muutuja x lugemine.

Ülesanne

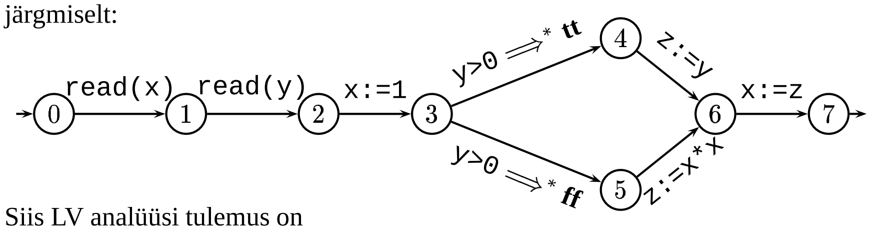
Leida iga programmipunkti kohta, millised muutujad on seal elavad.

Näide

Olgu meil programm

```
read(x); read(y); x:=1;
(if y>0 then z:=y else z:=x*x);
x:=z.
```

Elementaaroperatsioonideks jagunemine võiks meie analüüsi seisukohalt käia järgmiselt:

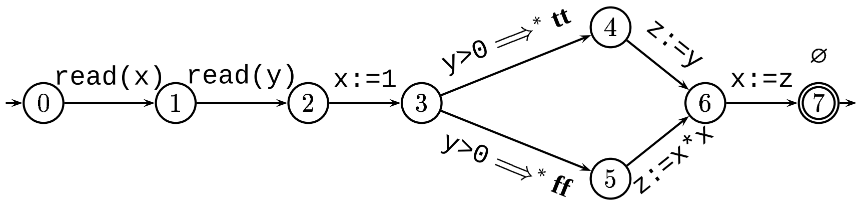


Siis LV analüüsi tulemus on

0	{}
1	{}
2	{y}
3	{x, y}
4	{y}
5	{x}
6	{z}
7	{}

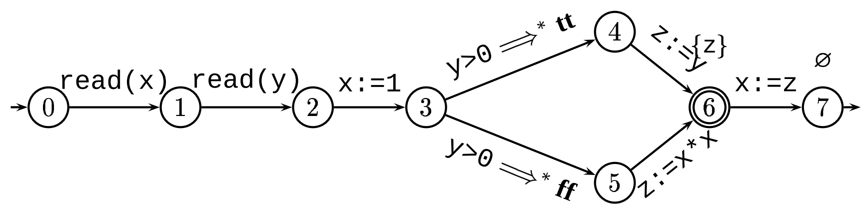
Süstemaatiline leidmine

Ülemine tee:



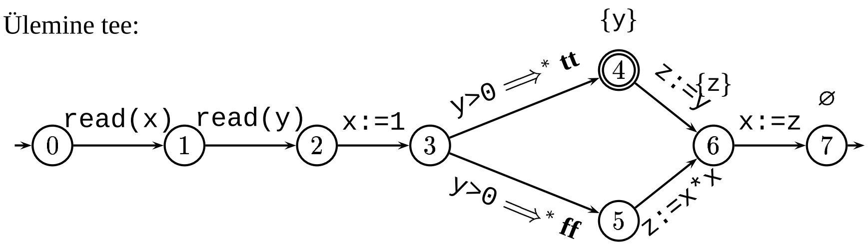
Süstemaatiline leidmine

Ülemine tee:



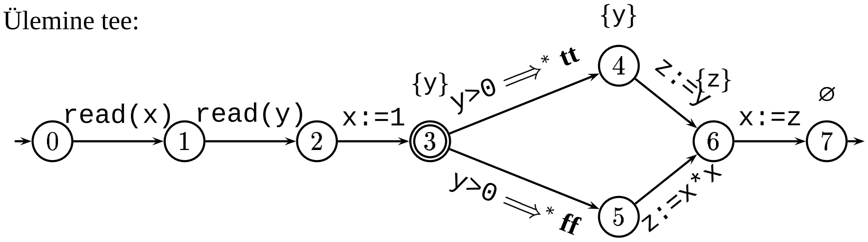
Süstemaatiline leidmine

Ülemine tee:



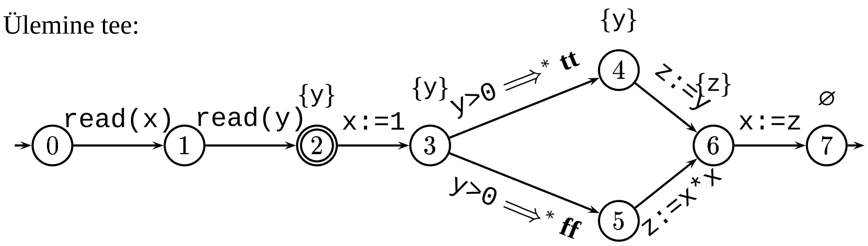
Süstemaatiline leidmine

Ülemine tee:



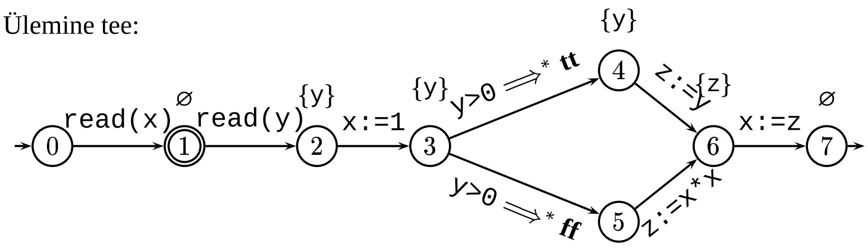
Süstemaatiline leidmine

Ülemine tee:



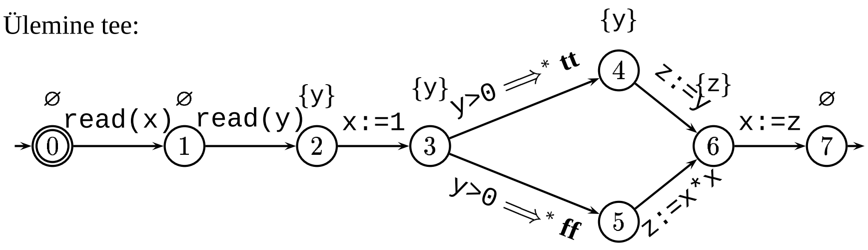
Süstemaatiline leidmine

Ülemine tee:



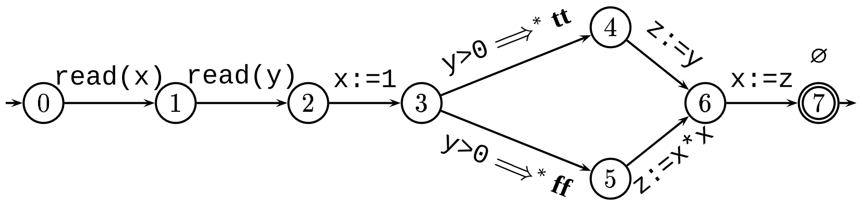
Süstemaatiline leidmine

Ülemine tee:



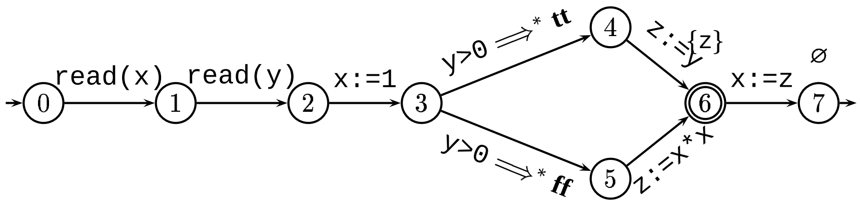
Süstemaatiline leidmine

Alumine tee:



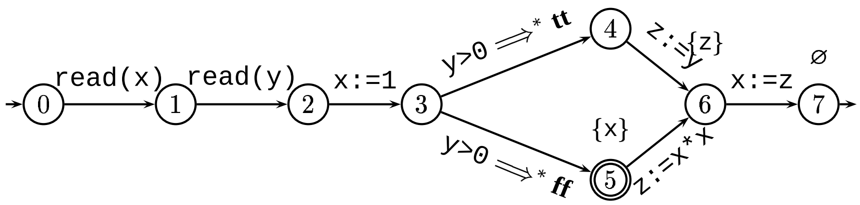
Süstemaatiline leidmine

Alumine tee:



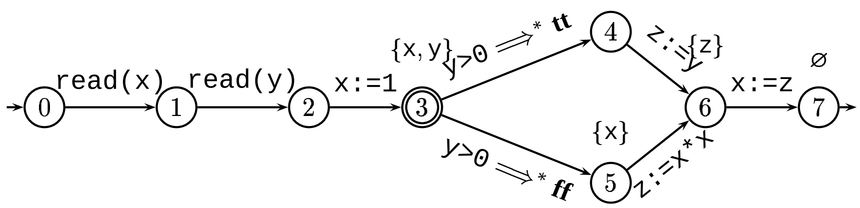
Süstemaatiline leidmine

Alumine tee:



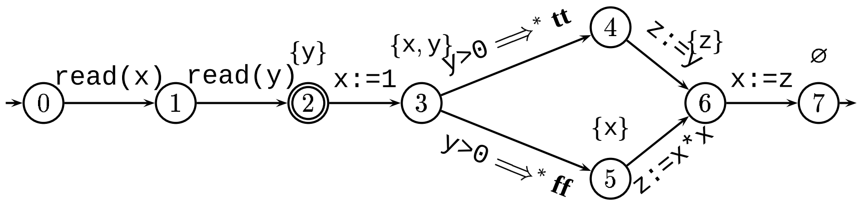
Süstemaatiline leidmine

Alumine tee:



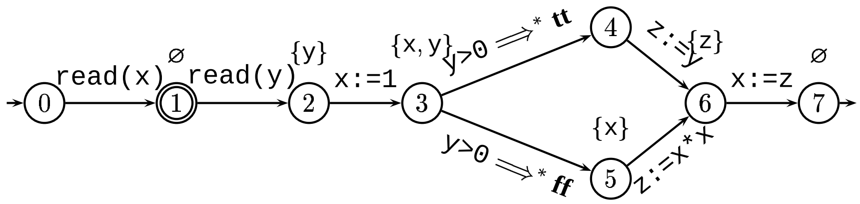
Süstemaatiline leidmine

Alumine tee:



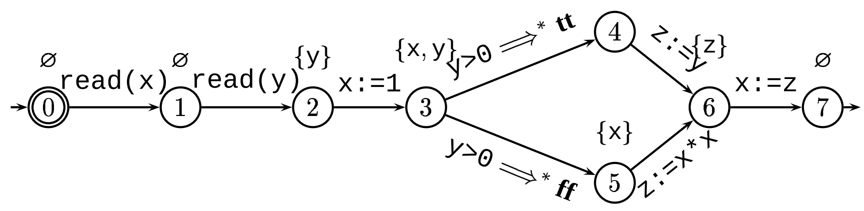
Süstemaatiline leidmine

Alumine tee:



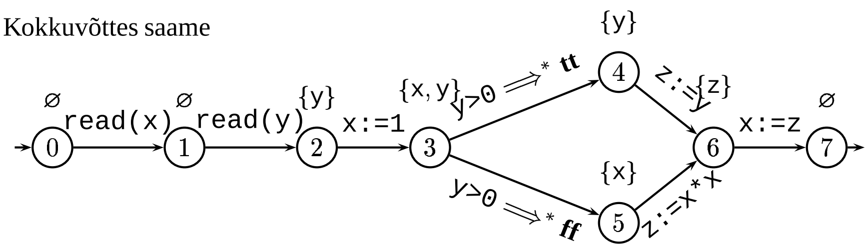
Süstemaatiline leidmine

Alumine tee:



Süstemaatiline leidmine

Kokkuvõttes saame



Constant Propagation

Ülesanne

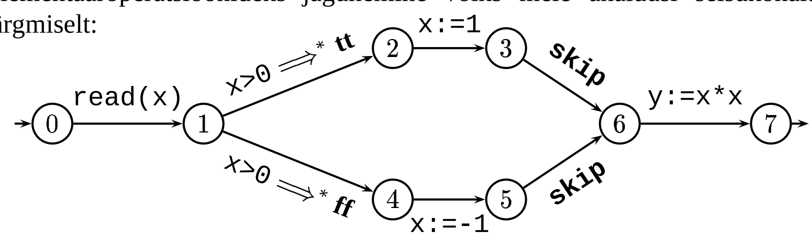
Leida iga programmipunkti kohta kõigi muutujate väärtused, millistel see on võimalik (st ei sõltu sellest, mil viisil on programmi täitmine sinna programmipunkti jõudnud).

Näide

Olgu meil programm

```
read(x);
(if x>0 then (x:=1; skip) else (x:=-1; skip));
y:=x*x.
```

Elementaaroperatsioonideks jagunemine võiks meie analüüsi seisukohalt käia järgmiselt:

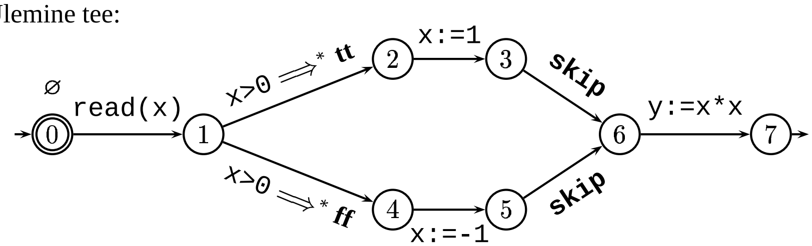


Siis CP analüüsi tulemus on

0	$\{\}$
1	$\{\}$
2	$\{\}$
3	$\{x = 1\}$
4	$\{\}$
5	$\{x = -1\}$
6	$\{\}$
7	$\{y = 1\}$

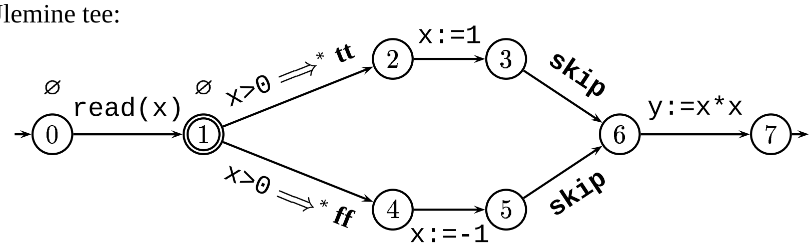
Süstemaatiline leidmine

Ülemine tee:



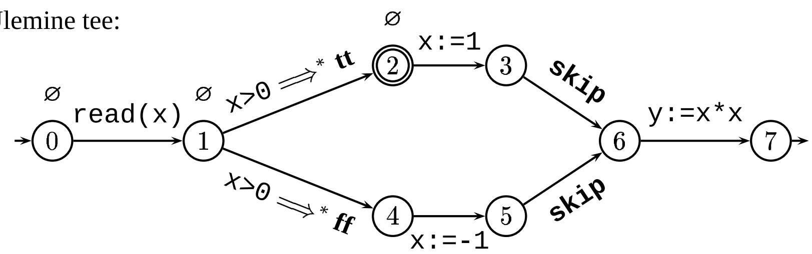
Süstemaatiline leidmine

Ülemine tee:



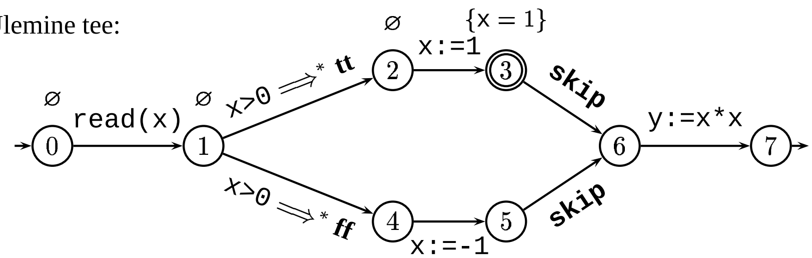
Süstemaatiline leidmine

Ülemine tee:



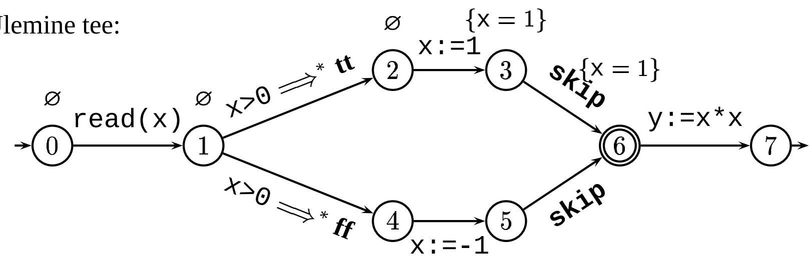
Süstemaatiline leidmine

Ülemine tee:



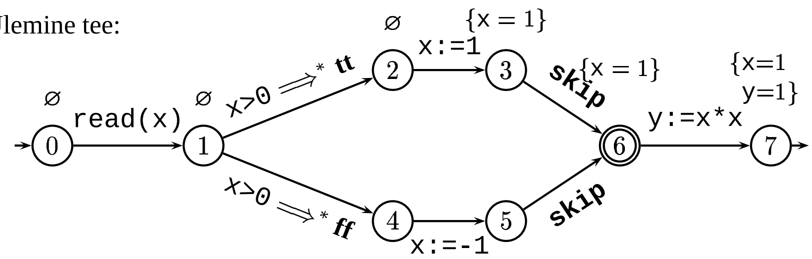
Süstemaatiline leidmine

Ülemine tee:



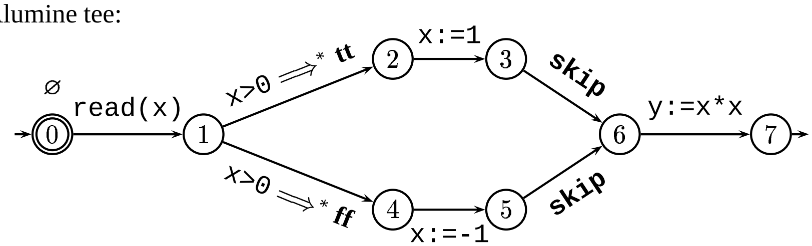
Süstemaatiline leidmine

Ülemine tee:



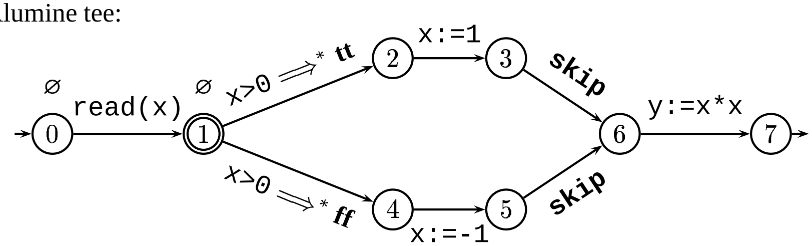
Süstemaatiline leidmine

Alumine tee:



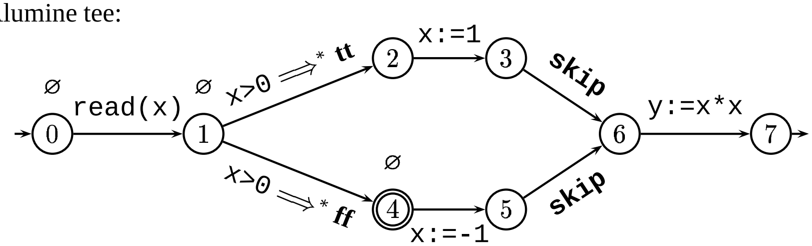
Süstemaatiline leidmine

Alumine tee:



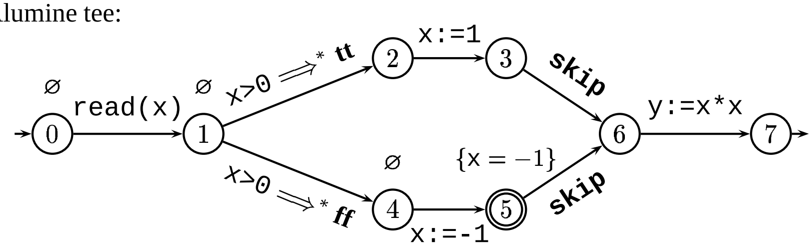
Süstemaatiline leidmine

Alumine tee:



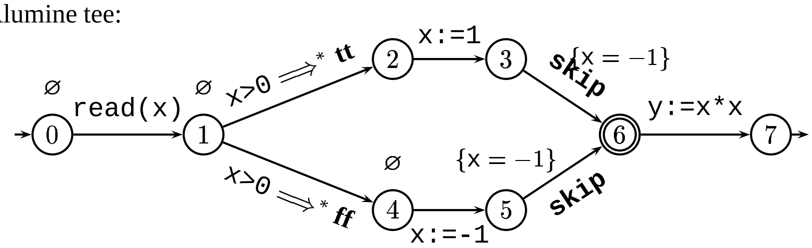
Süstemaatiline leidmine

Alumine tee:



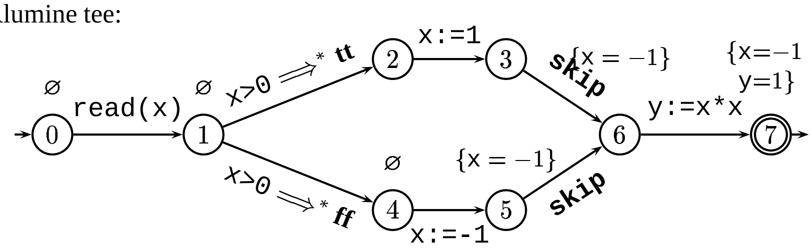
Süstemaatiline leidmine

Alumine tee:



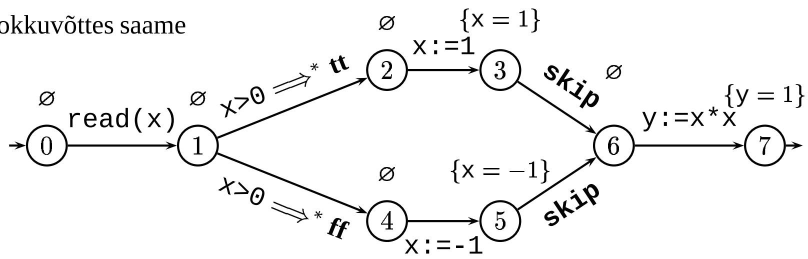
Süstemaatiline leidmine

Alumine tee:



Süstemaatiline leidmine

Kokkuvõttes saame



Üldine analüüsikäsitlus

Intraprotseduraalanalüüs

Olemus

Intraprotseduraalanalüüs uurib üksikuid protseduure.

Analüüsi spetsifitseerimine

- Määratakse võimalike olekute täisvõre $(S; \bigvee)$.
- Määratakse suund: **päripidi** või **tagurpidi**.
- Iga võimaliku juhtimisvoograafi kaare e jaoks määratakse üleminek, monotoonne olekuteisendus f_e.
  - Sõltub üldjuhul nii kaare märgendiks olevast programmeerimiskeele elementaarlausest kui kaare otstippudest.
  - Pere f üldistub loomulikul viisil võimalikele juhtimisvoograafi ahelatele:

$$f_\lambda = \mathrm{id},$$
$$f_{pe} = f_p \ ; f_e, \text{ (kui } p \text{ ja } pe \text{ on graafi ahelad).}$$

- Määratakse algolek ι.

Analüüsi läbiviimine

Olgu meil fikseeritud analüüs ning konkreetne protseduur.

- Leitakse analüüsi aluseks olev graaf.
  - Päripidianalüüsi puhul juhtimisvoograaf.
  - Tagurpidianalüüsi puhul juhtimisvoograafi pöördgraaf (st kaared on ümber pööratud ning alg- ja lõpptipud on vahetatud).
- Analüüsi oodatav tulemus väljendub seosega

$$a_w = \bigvee_{p: v_0 \to w} f_p(\iota),$$

(ülemraja leidmine toimub üle graafi ahelate, mis algavad graafi algtipust v_0 ja lõpevad tipus w).

Reaching Definitions

- Võre:
  - $S = (\mathbf{Var} \to \mathcal{P}(V \cup \{?\}))$ ($? \notin V$ tähistab initsialiseerimata olemist);
  - $\bigvee$ tuleb loomulikul hulgateoreetilisel viisil.
- Päripidianalüüs.
- Üleminekud:

$$f_{vw}(s) = \begin{cases} s[x \mapsto \{v\}], & \text{kui } vw \text{ on omistamine } x\,\texttt{:=}\,e, \\ s, & \text{kui } vw \text{ on testserv,} \\ s, & \text{kui } vw \text{ on } \textbf{\texttt{skip}}. \end{cases}$$

- $\iota(x) = \{?\}$ iga $x \in \mathbf{Var}$ korral.

Live Variables

- Võre:
  - $S = \mathcal{P}(\mathbf{Var})$;
  - $\bigvee$ tuleb loomulikul hulgateoreetilisel viisil.
- Tagurpidianalüüs.
- Üleminekud:

$$f_{vw}(s) = \begin{cases} (s \setminus \{x\}) \cup \mathrm{fv}(e), & \text{kui } vw \text{ on omistamine } x\,\texttt{:=}e, \\ s \cup \mathrm{fv}(e), & \text{kui } vw \text{ on testserv testavaldisega } e, \\ s, & \text{kui } vw \text{ on } \texttt{skip}. \end{cases}$$

- $\iota = \varnothing$.

Constant Propagation

- Võre:
  - $S = \{\bot\} \cup (\mathbf{Var} \to \mathbf{Data} \cup \{\top\})$ ($\mathbf{Data}$ tähistab muutujate võimalikke väärtusi ning $\top \notin \mathbf{Data}$ märgib, et muutuja väärtus pole ühene);
  - $\mathbf{Data} \cup \{\top\}$ peal defineeritakse ülemraja reegliga

    $$x \vee y = \begin{cases} x, & \text{kui } x = y, \\ \top & \text{ülejäänud juhtudel.} \end{cases}$$

    Loomulikul viisil üldistatakse see funktsioonidele $\mathbf{Var} \to \mathbf{Data} \cup \{\top\}$.
  - Iga funktsioon $s \in S \setminus \{\bot\}$ üldistub avaldistele loomulikul viisil (kui $s(y) \in \mathbf{Data}$ iga avaldises e sisalduva muutuja y korral, siis $s(e)$ on väärtus, mille programmeerimiskeele semantika sellele avaldisele nende muutujaväärtuste korral defineerib, vastasel korral $s(e) = \top$).
- Päripidianalüüs.
- Üleminekud:

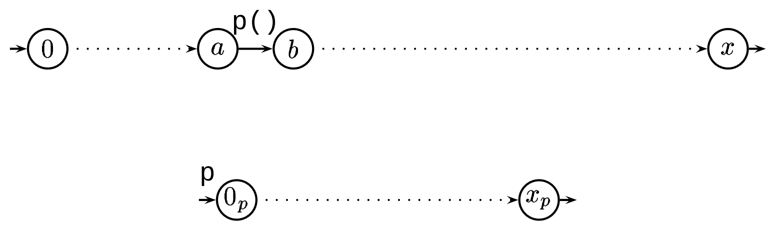
$$f_{vw}(s) = \begin{cases} s[x \mapsto s(e)], & \text{kui } vw \text{ on omistamine } x\texttt{:=}e, \\ s, & \text{kui } vw \text{ on testserv,} \\ s, & \text{kui } vw \text{ on } \texttt{skip}. \end{cases}$$

- $\iota(x) = \top$ iga $x \in \mathbf{Var}$ korral.

Interprotseduraalanalüüs

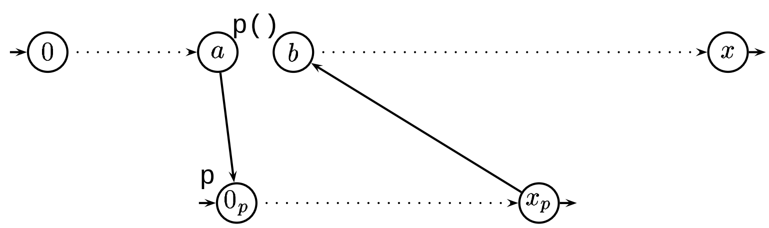
Naiivne lähenemine

Olgu meil järgmine juhtimisvoograafi fragment.



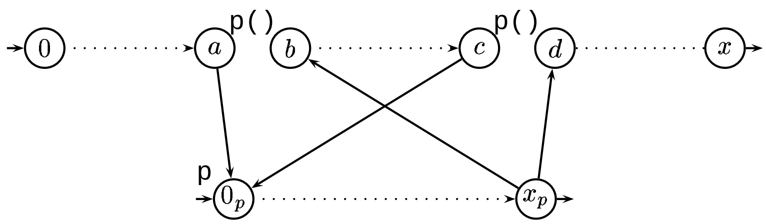
Naiivne lähenemine

Kirjeldame alamprogrammi kutseid ja tagasipöördumisi tavaliste kaartena tavalises juhtimisvoograafis.



Naiivne lähenemine

Kui sama protseduuri kutsutakse mitmest kohast,



tekivad graafi teed, mida reaalselt kunagi ei kasutata!

- $0, \ldots, a, 0_p, \ldots, x_p, d, \ldots, x.$
- $0, \ldots, c, 0_p, \ldots, x_p, b, \ldots, x.$

Funktsionaalne lähenemine

- Iga protseduur analüüsitakse omaette.
  - Algolek sõltub olekust väljakutsel.
  - Võib olla vaja analüüsida sama protseduuri mitme algolekuga.
- Fikseeritakse funktsioonid **entry** ja **combine**:
  - **entry** arvutab oleku järgi protseduuri väljakutsel selle protseduuri analüüsi algoleku;
  - **combine** arvutab protseduuri väljakutse aegse oleku ja protseduuri analüüsi lõppoleku järgi oleku protseduurist tagasipöördumise järel.
- Iga protseduurikutsega kaar omandab olekufunktsioonilise väärtuse.
  - Olgu
    - vw protseduuri p väljakutse,
    - p olekuteisendus, mis seab protseduuri p analüüsi suvalisele algolekule vastavusse oleku tema lõpppunktis, kui analüüs algas antud algolekust.

    Siis

    $$f_{vw}(s) = \textbf{combine}(s, p(\textbf{entry}(s))).$$
- Rekursiivse protseduuri korral võib juhtuda, et teda tuleb analüüsida lõpmata paljude erinevate algolekute jaoks!

Kutsestringide lähenemine

- Sisuliselt tehakse iga protseduuri juhtimisvoograafist koopia iga võimaliku kutsestringi jaoks.
- Rekursiooni korral paisub analüüs lõpmatuks!
  - Määratakse maksimumsügavus: kui kutsestring läheb pikemaks, arvestatakse ainult lõpuosa.

Näited maatriksite korrutamisest

Maatriksioperatsioonid

Kõik me teame, et...

- ...kui M ja N on arvuliste väärtustega maatriksid, kusjuures nende mõõtmed langevad kokku, siis $M+N$ defineeritakse samamõõtmelise maatriksina reegliga
$$(M+N)_{i,j} = M_{i,j} + N_{i,j}.$$
- ...$m \times n$-maatriks θ, mis antakse reegliga $\theta_{i,j} = 0$, käitub $m \times n$-maatriksite liitmise mõlemapoolse ühikuna.
- ...kui M ja N on arvuliste väärtustega maatriksid, kusjuures M mõõtmed on $m \times s$ ja N mõõtmed on $s \times n$, siis $M \cdot N$ defineeritakse $m \times n$-maatriksina reegliga
$$(M \cdot N)_{i,j} = \sum_{k=1}^{s} M_{i,k} \cdot N_{k,j}.$$
- ...ruutmaatriks E, mis antud reegliga
$$E_{i,j} = \begin{cases} 1, & \text{kui } i = j, \\ 0 & \text{ülejäänud juhtudel,} \end{cases}$$
käitub maatriksite korrutamise mõlemapoolse ühikuna.

Kuhu suundume?

- Arvude asemel võib võtta midagi muud.
- Elementide liitmise ja korrutamise kohale võib võtta mingid muud operatsioonid.
- Ühikuteks osutuvad mingid muud objektid.
- Lõpmatute summade olemasolu korral on mõtet rääkida täisiteratsioonist

$$M^* = \sum_{i=0}^{\infty} M^i.$$

Relatsioonid tõeväärtusmaatriksina

Süsteem

- Algebraliseks struktuuriks olgu $(\mathbf{T}; \vee, \wedge, 0, 1, \bigvee)$, kus $\mathbf{T} = \{0, 1\}$ on tõeväärtuste hulk.
- Olgu A, B hulgad, mille vahel antud relatsioon $\rho \subseteq A \times B$. Esitame ta oma karakteristliku funktsiooni $M_\rho : A \times B \to \mathbf{T}$ abil, mis sisuliselt kujutab endast maatriksit.

Maatriksioperatsioonide tähendus

- Olgu A, B hulgad, mille vahel relatsioonid $\rho, \sigma \subseteq A \times B$. Siis

$$M_\rho + M_\sigma = M_{\rho \cup \sigma}.$$

- Olgu A, B, C hulgad, mille vahel relatsioonid $\rho \subseteq A \times B$ ja $\sigma \subseteq B \times C$. Siis

$$M_\rho \cdot M_\sigma = M_{\rho;\sigma}.$$

- Olgu A hulk, millel relatsioon $\rho \subseteq A \times A$. Siis

$$M_\rho^* = M_{\rho^{rt}}$$

  (ρ maatriksi täisiteratsioon võrdub ρ refleksiivse transitiivse sulundi maatriksiga).

Graafide naabrusmaatriksid

Süsteem

- Algebraliseks struktuuriks olgu $(\mathbb{N} \cup \{\infty\}; +, \cdot, 0, 1, \sum)$, kus lõpmata paljude positiivsete liidetavatega summa väärtus on ∞.
- Olgu $G = (V, E)$ orienteeritud graaf. Esitame ta naabrusmaatriksina

  $$(M_G)_{v,w} = |\{e \in E : \iota(e) = v \ \wedge \ \tau(e) = w\}|,$$

  st $(M_G)_{v,w}$ on kaarte arv tipust v tippu w.

Maatriksioperatsioonide tähendus

- Olgu $G = (V, E)$ orienteeritud graaf. Siis

  $(M_G^i)_{v,w}$ on pikkusega i marsruutide arv tipust v tippu w graafis G;
  $(M_G^*)_{v,w}$ on marsruutide arv tipust v tippu w graafis G.

Odavaima tee leidmine

Süsteem

- Algebraliseks struktuuriks olgu $(\mathbb{R}^{\geqslant 0} \cup \{\infty\}\,; \min, +, \infty, 0, \inf)$.
- Olgu $G = (V, \omega)$ kaalutud servadega graaf, st $\omega : V \times V \to \mathbb{R}^{\geqslant 0} \cup \{\infty\}$ ($\omega(v, w) = \infty$ tähendab, et tipust v tippu w ei saa). Maatriksisse paneme kaalud:

$$(M_G)_{v,w} = \omega(v, w) \quad \text{ehk} \quad M_G = \omega.$$

Maatriksioperatsioonide tähendus

- Olgu $G = (V, \omega)$ kaalutud servadega graaf. Siis

$(M_G^i)_{v,w}$ on pikkusega i teedest tipust v tippu w kõige odavama hind;
$(M_G^*)_{v,w}$ on tipust v tippu w kõige odavama tee hind.

Lõplikud automaadid maatriksina

Süsteem

Olgu X fikseeritud tähestik.

- Algebraliseks struktuuriks olgu $(\mathcal{P}(X^*); \cup, \cdot, 0, 1, \bigcup)$, kus
  * suvaliste sõnahulkade W_1, W_2 jaoks
  $$W_1 \cdot W_2 = \{w_1 w_2 : w_1 \in W_1, w_2 \in W_2\},$$
  * $0 = \varnothing$,
  * $1 = \{\lambda\}$.
- Olgu $A = (Q, X, \phi, I, T)$ lõplik (mittedetermineeritud) automaat (Q on olekute hulk, $\phi : Q \times X \to \mathcal{P}(Q)$ üleminekufunktsioon, $I \subseteq Q$ ja $T \subseteq Q$ vastavalt alg- ja lõppolekute hulgad). Esitame ta maatriksina
  $$(M_A)_{q_1, q_2} = \{x \in X : q_2 \in \phi(q_1, x)\}.$$
- Eelmises punktis defineeritud maatriks ei sõltu hulkadest I ja T. Nendele defineerime eraldi vastavalt karakteristliku rea $R_I : \{\bullet\} \times Q \to \mathcal{P}(X^*)$ ja veeru $V_T : Q \times \{\bullet\} \to \mathcal{P}(X^*)$:
  $$(R_I)_{\bullet, q} = \begin{cases} 1, & \text{kui } q \in I, \\ 0 & \text{ülejäänud juhtudel;} \end{cases}$$
  $$(V_T)_{q, \bullet} = \begin{cases} 1, & \text{kui } q \in T, \\ 0 & \text{ülejäänud juhtudel.} \end{cases}$$

Maatriksioperatsioonide tähendus

- Olgu $A = (Q, X, \phi, I, T)$ automaat. Siis

  $(M_A^i)_{q_1,q_2}$ on pikkusega i sõnade hulk, millega saab olekust q_1 olekusse q_2;
  $(M_A^*)_{q_1,q_2}$ on sõnade hulk, millega saab olekust q_1 olekusse q_2;
  $(R_I \cdot M_A^* \cdot V_T)_{\bullet,\bullet} = \mathcal{L}(A)$.

Maatriksite korrutamine — graafi teede analüüs

Poolring

- Algebralist struktuuri $\mathbf{R} = (R; +, \cdot, 0, 1)$ nimetatakse **poolringiks**, kui
  - $(R; +, 0)$ on kommutatiivne monoid,
  - $(R; \cdot, 1)$ on monoid,
  - 0 on $\cdot$ nullelement,
  - $\cdot$ on $+$ suhtes mõlemat pidi distributiivne.
- Kõigi toodud maatriksinäidete puhul oli algebraliseks struktuuriks mingi poolring.

Teoreem

Olgu V mingi tippude, olekute vms hulk. Olgu antud algebraline struktuur $(R; +, \cdot, 0, 1, \sum)$.

1. Kui $(R; +, \cdot, 0, 1)$ on poolring, siis iga maatriksi $M : V \times V \to R$, $i \in \mathbb{N}$ ja $v, w \in V$ korral

$$M^i_{v,w} = \sum_{v=v_0,\ldots,v_i=w\in V} M_{v_0,v_1} \cdot \ldots \cdot M_{v_{i-1},v_i}. \qquad (1)$$

2. Kui lisaks on $\sum$ ositi leitav, st

$$\text{kui } A = \bigcup_{i=0}^{\infty} B_i,\ B_i \perp B_j \Leftarrow i \neq j, \text{ siis } \sum_{a\in A} a = \sum_{i=0}^{\infty} \sum_{b\in B_i} b,$$

siis iga maatriksi M ja $v, w \in V$ korral

$$M^*_{v,w} = \sum_{v=v_0,\ldots,v_i=w} M_{v_0,v_1} \cdot \ldots \cdot M_{v_{i-1},v_i}, \qquad (2)$$

kus summeerimine toimub üle kõigi lõplike korteežide, mille esimene komponent on v ja viimane w.

Järeldus

Olgu lisaks antud kaarehulk E, st $G = (V, E)$ on orienteeritud graaf. Kui iga $(v, w) \in V \times V, M_{v,w} \neq 0$ korral leidub $e \in E$, nii et $\iota(e) = v$ ja $\tau(e) = w$, siis võib valemites (1) ja (2) summeerida ainult üle graafi marsruutide.

Programmianalüüs maatriksite korrutamise ülesandena

Püstitus

Süsteem: esmane versioon

Olgu fikseeritud analüüs A, st meil on

- täisvõre $(S; \bigvee)$,
- orienteeritud graaf $G = (P, E)$ ja monotoonsed üleminekufunktsioonid $f_e : S \to S$.

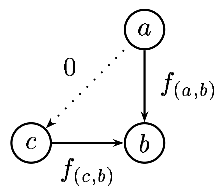
- Algebraliseks struktuuriks olgu $((S; \leq) \to (S; \leq); \vee, ;, 0, \mathrm{id}, \bigvee)$, kus
  - $(S; \leq) \to (S; \leq)$ tähistab võre $(S; \bigvee)$ monotoonsete teisenduste hulka,
  - $\vee$ on S binaarse ülemrajaoperatsiooni loomulik üldistus teisendustele,
  - analoogselt saame $\bigvee$,
  - $0(s) = \bot$ iga $s \in S$ korral.
- Analüüsi A maatriks on

$$(M_A)_{p,q} = \begin{cases} f_{(p,q)}, & \text{kui } (p,q) \in E, \\ 0 & \text{ülejäänud juhtudel.} \end{cases}$$

Väike probleem

Maatriksi element, mis vastab kaare puudumisele, ei tohiks arvutust mõjutada.

- Paraku kuna üleminekud f_e ei pea olema agarad (st ei pea säilitama vähimat elementi $\perp$), siis üldjuhul $0 ; f_e = 0$ ei kehti.
  - Seega "läbi seina minek" võib anda kõrvalefekti, nt kui joonisel toodud graafis ükski tee tipust a tippu b ei läbi tippu c.



Lahendus

- Et 0 oleks siiski nullelement, nagu teoreemi eeldus nõuab, laiendame hulka S "superbottomiga" $\underline{\perp}$, mis on S kõigist elementidest väiksem, ja defineerime
  - $0(s) = \underline{\perp}$ iga $s \in S$ korral,
  - $f_e(\underline{\perp}) = \underline{\perp}$ iga $e \in E$ korral.
- Muutus ei mõjuta analüüsi tulemust, sest analüüs olematuid kaari ega superbottomit ei kasuta.
- Veel rahustuseks: kui graafis on täpselt üks algustipp, nagu normaalne, siis analüüsi tulemust "läbi seina minekud" ei mõjuta ka ilma superbottomita:

$$\perp \leq s \;\Rightarrow\; f_e(\perp) \leq f_e(s) \;\Rightarrow\; (0\,;f_e)(x) \leq f_e(s).$$

Väide

Väide

Algebraline struktuur $((S;\leq,\underline{\perp}) \to (S;\leq,\underline{\perp});\vee,;,\mathrm{const}\,\underline{\perp},\mathrm{id},\bigvee)$ on poolring, milles suur summa on ositi leitav.

- Järeldus: analüüsi A tulemus avaldub maatriksiga IM_A^*, kus

$$I_{\bullet,p} = \begin{cases} \bullet \mapsto \iota, & \text{kui } p \text{ on algtipp } (\iota \text{ on algolek}), \\ \bullet \mapsto \underline{\perp} & \text{ülejäänud juhtudel.} \end{cases}$$

Teame

Struktuur $((S; \leq, \perp) \rightarrow (S; \leq, \perp); \bigvee)$ on täisvõre, kuna ta on ehitatud S teisendustest, kasutades komponentviisil täisvõre $(S; \bigvee)$ tehet.

Väite tõestuse etapp: liitmise assotsiatiivsus

- Iga $f, g, h : (S; \leq, \perp) \to (S; \leq, \perp)$ korral $f \vee (g \vee h) = (f \vee g) \vee h$.

Väite tõestuse etapp: liitmise assotsiatiivsus

- Iga $f, g, h : (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $f \vee (g \vee h) = (f \vee g) \vee h$.
  - TÕESTUS. Tuleneb sellest, et $((S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp}); \bigvee)$ on täisvõre.

Väite tõestuse etapp: liitmise ühik

- Iga $f : (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $f \vee \mathrm{const}\,\underline{\perp} = f$ ja $\mathrm{const}\,\underline{\perp} \vee f = f$.

Väite tõestuse etapp: liitmise ühik

- Iga $f : (S; \leq, \perp) \to (S; \leq, \perp)$ korral $f \vee \mathrm{const}\,\underline{\perp} = f$ ja $\mathrm{const}\,\underline{\perp} \vee f = f$.
  - TÕESTUS. Suvalise $s \in S$ korral

$$
\begin{aligned}
(f \vee \mathrm{const}\,\underline{\perp})(s) &= f(s) \vee (\mathrm{const}\,\underline{\perp})(s) = f(s) \vee \underline{\perp} = f(s),\\
(\mathrm{const}\,\underline{\perp} \vee f)(s) &= (\mathrm{const}\,\underline{\perp})(s) \vee f(s) = \underline{\perp} \vee f(s) = f(s).
\end{aligned}
$$

Väite tõestuse etapp: liitmise kommutatiivsus

- Iga $f, g : (S; \leq, \perp) \to (S; \leq, \perp)$ korral $f \vee g = g \vee f$.

Väite tõestuse etapp: liitmise kommutatiivsus

- Iga $f, g : (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $f \vee g = g \vee f$.
  - TÕESTUS. Tuleneb sellest, et $((S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp}); \bigvee)$ on täisvõre.

Väite tõestuse etapp: liitmise ositi leitavus

- Olgu $F = \bigcup_{i=0}^{\infty} G_i$, kus G_i on paarikaupa lõikumatud. Siis $\bigvee F = \bigvee_{i=0}^{\infty} \bigvee G_i$.

Väite tõestuse etapp: liitmise ositi leitavus

- Olgu $F = \bigcup_{i=0}^{\infty} G_i$, kus G_i on paarikaupa lõikumatud. Siis $\bigvee F = \bigvee_{i=0}^{\infty} \bigvee G_i$.
  - TÕESTUS. Täisvõre ülemrajaoperatsioon on ositi leitav ja $((S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp}); \bigvee)$ on täisvõre.

Väite tõestuse etapp: korrutamise assotsiatiivsus

- Iga $f, g, h : (S; \leq, \perp) \to (S; \leq, \perp)$ korral $(f \; ; \; g) \; ; \; h = f \; ; \; (g \; ; \; h)$.

Väite tõestuse etapp: korrutamise ühik

- Iga $f : (S; \leq, \perp) \to (S; \leq, \perp)$ korral $f \,;\, \mathrm{id} = f$ ja $\mathrm{id} \,;\, f = f$.

Väite tõestuse etapp: korrutamise null

- Iga $f : (S; \leq, \underline{\perp}) \rightarrow (S; \leq, \underline{\perp})$ korral $f\ ;\ \mathrm{const}\,\underline{\perp} = \mathrm{const}\,\underline{\perp}$ ja $\mathrm{const}\,\underline{\perp}\ ;\ f = \mathrm{const}\,\underline{\perp}$.

Väite tõestuse etapp: korrutamise null

- Iga $f : (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $f \,;\, \mathrm{const}\,\underline{\perp} = \mathrm{const}\,\underline{\perp}$ ja $\mathrm{const}\,\underline{\perp} \,;\, f = \mathrm{const}\,\underline{\perp}$.
  - TÕESTUS. Iga $s \in S$ korral

$$(f \,;\, \mathrm{const}\,\underline{\perp})(s) = (\mathrm{const}\,\underline{\perp})(f(s)) = \underline{\perp} = (\mathrm{const}\,\underline{\perp})(s),$$
$$(\mathrm{const}\,\underline{\perp} \,;\, f)(s) = f((\mathrm{const}\,\underline{\perp})(s)) = f(\underline{\perp}) = \underline{\perp} = (\mathrm{const}\,\underline{\perp})(s)$$

Väite tõestuse etapp: distributiivsus vasakult

- Iga $f, g, h \in (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $f \,;\, (g \vee h) = f \,;\, g \vee f \,;\, h$.

Väite tõestuse etapp: distributiivsus vasakult

- Iga $f, g, h \in (S; \leq, \perp) \to (S; \leq, \perp)$ korral $f\ ;(g \vee h) = f\ ; g \vee f\ ; h$.
  - TÕESTUS. Suvalise $s \in S$ korral

$$\begin{aligned}(f\ ;(g \vee h))(s) &= (g \vee h)(f(s)) = g(f(s)) \vee h(f(s)) = \\ &= (f\ ; g)(s) \vee (f\ ; h)(s) = (f\ ; g \vee f\ ; h)(s).\end{aligned}$$

Väite tõestuse etapp: distributiivsus paremalt

- Iga $f, g, h \in (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $(f \vee g)\ ;\ h = f\ ;\ h \vee g\ ;\ h$.

Väite tõestuse etapp: distributiivsus paremalt

- Iga $f, g, h \in (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $(f \vee g)\ ;\ h = f\ ;\ h \vee g\ ;\ h$.
  - TÕESTUS. Suvalise $s \in S$ korral

$$((f \vee g)\ ;\ h)(s) = h((f \vee g)(s)) = h(f(s) \vee g(s)) =?$$

Väite tõestuse etapp: distributiivsus paremalt

- Iga $f, g, h \in (S; \leq, \underline{\perp}) \to (S; \leq, \underline{\perp})$ korral $(f \vee g) \mathbin{;} h = f \mathbin{;} h \vee g \mathbin{;} h$.
  - TÕESTUS. Suvalise $s \in S$ korral

$$((f \vee g) \mathbin{;} h)(s) = h((f \vee g)(s)) = h(f(s) \vee g(s)) =?$$
$$(f \mathbin{;} h \vee g \mathbin{;} h)(s) = (f \mathbin{;} h)(s) \vee (g \mathbin{;} h)(s) = h(f(s)) \vee h(g(s)) =?$$

Väite tõestuse etapp: distributiivsus paremalt

- Iga $f, g, h \in (S; \leq, \perp) \to (S; \leq, \perp)$ korral $(f \vee g) \mathbin{;} h = f \mathbin{;} h \vee g \mathbin{;} h$.
  - TÕESTUS. Suvalise $s \in S$ korral

$$((f \vee g) \mathbin{;} h)(s) = h((f \vee g)(s)) = h(f(s) \vee g(s)) =?$$
$$(f \mathbin{;} h \vee g \mathbin{;} h)(s) = (f \mathbin{;} h)(s) \vee (g \mathbin{;} h)(s) = h(f(s)) \vee h(g(s)) =?$$

Ei tulegi välja!

Tegelik teoreem

Distributiivne analüüs

Analüüsi olekuhulgaga S ja teisenduste perega f nimetatakse **distributiivseks**, kui

> iga $s_1, s_2 \in S$ ja e (potentsiaalne kaar) korral $f(s_1 \vee s_2) = f(s_1) \vee f(s_2)$,

st

> iga potentsiaalne olekuteisendus säilitab binaarse ülemraja.

Teoreem

Distributiivsele analüüsile vastav algebraline struktuur

$$((S;\leq,\perp)\to(S;\leq,\perp);\vee,;,\mathrm{const}\,\perp,\mathrm{id},\bigvee)$$

on poolring, milles suur summa on ositi leitav.

– Järeldus: distributiivse analüüsi A tulemus avaldub maatriksiga IM_A^*, kus

$$I_{\bullet,p}=\begin{cases}\bullet\mapsto\iota, & \text{kui } p \text{ on algtipp } (\iota \text{ on algolek}),\\ \bullet\mapsto\perp & \text{ülejäänud juhtudel.}\end{cases}$$

Veel seoseid

Püsipunktivõrrandid

Olgu A distributiivne analüüs.

- IM_A^* on võrrandi $I + XM_A = X$ vähim lahend ehk

  kujutuse $F(X) = I + XM_A$ vähim püsipunkt.

- Aga programmianalüüsis püsipunkte arvutataksegi. Maatrikskujul oleks see

  kujutuse $F(X) = X + XM_A$ vähim püsipunkt X,
  mis rahuldab tingimust $I \leq X$.

- Viimane tingimus on samaväärne

  funktsiooni $F(X) = I + XM_A$ vähima eelpüsipunkti leidmisega,

  mis on teadaolevalt samaväärne

  sama F vähima püsipunkti leidmisega.

- Seega programmianalüüsi klassikaline püsipunktialgoritm sisuliselt leiab IM_A^*.

Näide mittedistributiivsest analüüsist

- Võtame
  - Constant Propagation analüüsi,
  - üleminekuks elementaarlausele `y:=x*x` vastava ülemineku f,
  - olekuteks $s_1 = \{\mathsf{x} = 1\}$, $s_2 = \{\mathsf{x} = -1\}$.
- Siis

$$f(s_1) \vee f(s_2) = \{\mathsf{x} = 1, \mathsf{y} = 1\} \vee \{\mathsf{x} = -1, \mathsf{y} = 1\} = \{\mathsf{y} = 1\}\,,$$
$$f(s_1 \vee s_2) = f(\{\}) = \{\}\,.$$

Mittelahenduvus

Constant Propagation on koguni mittelahenduv!

Mis on maatriksitest kasu?

- Funktsionaalsel lähenemisel tuleb sama protseduuri analüüsida paljude algtingimustega.

  St IM_P^* tuleb leida paljude ridade I jaoks.

  Arvutame õige M_P^* ära ja siis iga I-ga lihtsalt korrutame, pole vaja iga kord pikalt itereerida!?

  – M_P^* on funktsioonide maatriks. Et sihukest iteratsiooniga kätte saada, tuleb igal sammul kontrollida, kas uue maatriksi kõik funktsioonid võrduvad vana maatriksi vastavate funktsioonidega. Selleks tuleb rakendada igaüht neist igale võimalikule argumendile $\Rightarrow$ haa-haa. . .

- Aga teoreetilisteks targutusteks võivad nad mugavad olla küll :)