Loeng 4: Loogilise programmeerimise alused

J.Vain

Loengu kava

- Loogika valemid Horni lause kujul
  - Fakt
  - Reegel
  - Päring
- Horni lausete tõestmine resolutsiooni meetodiga
- Termide unifitseerimine kui resolutsiooni eeldus
- Näided
- *Takeaway*
  - Teadmised Prologi tõestusmehhanismist

Tagasi algusesse - Horni lause

- *Loogiline programm* koosneb *Horni lausetest* (edaspidi HL)
- Horni lause üldkujul (loogikas):

$$p \leftarrow q_1, \dots, q_n,$$

kus

p – Horni *lause päis*

$q_1, \dots, q_n$ – Horni *lause keha*

Horni lause

- Horni lause $p \leftarrow q_1,\ldots, q_n$ esituskujud:
  - *implikatiivne* kuju:

$$p \leftarrow q_1 \wedge \ldots \wedge q_n$$

    - kus p ja $q_1,\ldots, q_n$ tähistavad *literaale*
    - *Literaal* – atomaarne valem või selle eitus
    - *Atomaarne valem*: $P(t_1,\ldots, t_n)$, kus P – *predikaadi sümbol*
      $t_1, \ldots, t_n$ – termid
  - *disjunktiivne* kuju (sisaldab ≤1 positiivse literaali):

$$p \vee \neg\, q_1 \vee \ldots \vee \neg\, q_n$$

- Implikatiivne ja disjunktiivne kuju on *semantiliselt ekvivalentsed*!

Kuidas mõista Horni lauset $p \leftarrow q_1,\ldots, q_n$?

- Operatsiooniline semantika:

  "Programmi p täitmine seisneb alamprogrammide $q_1, \ldots, q_n$ täitmises"
- Deklaratiivne semantika:

  "Lause p tõesus järeldub lausete $q_1,\ldots, q_n$ tõesusest" ehk

  "*Lause p kehtib siis, kui kehtivad laused $q_1,\ldots, q_n$*"

- Muutujate tõlgendamine reeglis:

  Kõik muutujad Horni lauses on seotud kvantoritega:

  $\forall x_1 \ldots x_n\ \exists y_1 \ldots y_m, \ldots, z_1 \ldots z_l : P(x_1 \ldots x_n) \leftarrow Q_1(y_1 \ldots y_m) \wedge \ldots \wedge Q_n(z_1 \ldots z_l)$ — Predikaat-arvutus (PA)

  `p(X1,...,Xn):- q1(Y1,...,Ym),...,qn(Z1,...,Zl)` — Prolog

Muutujate $x_1 \ldots x_n$ mistahes väärtustuse korral on predikaat P tõene, kui leidub muutujate $y_1 \ldots y_m, \ldots, z_1 \ldots z_l$ väärtustus, mille korral on predikaadid $Q_1, \ldots, Q_n$ tõesed.

Horni lause tüübid

- *Fakt*

  lause, millel puudub keha (disjunktiivse kuju puhul ainsast positiivsest literaalist koosnev disjunktsioon).

  Näide: $Q(a, b) \leftarrow$ ehk $Q(a, b)$ % tingimusteta tõene lause (PA)

  `isa(jaan, peeter).` (Prolog)

- *Reegel*

  HL, millel on pea ja keha (disjunktiivse kuju puhul ühest positiivsest ja vähemalt ühest negatiivsest literaalist koosnev disjunktsioon).

  Näide: $Q(a, b) \leftarrow R(a, v), \ldots, P(b, b, w)$ % Implikatsioon (PA)

  `vanaisa(X,Y):-isa(X,Z),...,ema(Z, Y).` (Prolog)

Horni lause tüübid

- *Päring* – HL, millel on ainult keha (disjunktiivse kuju puhul vähemalt ühest negatiivsest literaalist koosnev disjunktsioon).

- Prologi päring käivitab lahendi leidmiseks otsingumootori

  Näide: `?- isa(juku,X).`

Suletud maailma eeldus

Suletud maailma eeldus:

Tõene on ainult see väide, mille tõesuse saab tuletada loogilises programmis ja teadmusbaasis olevatest Horni lausetest

- NB!
  - Faktide puudumine teadmusbaasist ei tähenda määramatust
  - Fakti puudumine teadmusbaasist tähendab fakti eitust!

Resolutsioon

- *Resolutsioon* - Horni lausete kujul oleva deduktiivse süsteemi tõestusreegel.
- Kuidas mõista resolutsiooni?
  - Prolog kasutab resolutsiooni vastuväitelises tõestusskeemis:

Vastuväiteline tõestamine:

Olgu $D \leftarrow \Delta$ Horni lause, kus *D* on reegli päis ja Δ on reegli keha.

Valem $D \leftarrow \Delta$ kehtib parajasti siis, kui tema eitus $\neg(D \leftarrow \Delta)$ on väär.

- Valem on väär, kui sellest saab tuletada ***vastuolu***.
- Vastuolu Prologi mõttes tähendab, et teadmusbaasis ei leidu niisugusi fakte, millest saaks tuletada tõestatava väite.
- Seega valemi tõestamiseks püüame tuletada valemi eitusest tühja valemi
- *Resolutsiooni reegel* (RR) võimaldab tuletada tõestatava valemi eitusest *tühja valemi* kasutades vahe-eelduste „väljalõikamist“

Resolutsiooni reegel

Olgu Horni lause implikatsiooni kujul

- HL päring: $\leftarrow A_1, B_2, \dots, B_m$, milles esineb literaal A_1
- HL reegel: $A_1 \leftarrow A_2, \dots, A_n$, kus literaal A_1 on järelduses,

siis rakendades resolutsiooni reeglit

$$\frac{A_1 \leftarrow A_2, \dots, A_n \qquad \leftarrow A_1, B_2, \dots, B_m}{\leftarrow A_2, \dots, A_n, B_2, \dots, B_m}$$

asendame päringus literaali A_1 reegli $A_1 \leftarrow A_2, \dots, A_n$ eeldustega $A_2, \dots, A_n$.
Tulemusena omandab algne päring $\leftarrow A_1, B_2, \dots, B_m$ kuju:

$$\leftarrow A_2, \dots, A_n, B_2, \dots, B_m$$

Kui päringu kõigi literaalide asendamisel on jõutud faktideni, siis neil puuduvad eelduslaused ja faktide asendamine annab tühja literaali.

Resolutsiooni reegel disjunktsiooni kujul olevate Horni lausetega

Nagu eelnevalt nägime, saab HL kirjutada ka disjunktsiooni kujul. Sel juhul omandab RR järgmise kuju:

$$\frac{\neg A_1 \vee A_2 \vee \ldots \vee A_n \qquad A_1 \vee B_2 \vee \ldots \vee B_m}{A_2 \vee \ldots \vee A_n \ \vee \ \ B_2 \vee \ldots \vee B_m} \qquad \text{(RR)}$$

Tõestus:

Olgu $\Gamma \equiv \neg A_1 \vee A_2 \vee \ldots \vee A_n$ ja $\Lambda \equiv A_1 \vee B_2 \vee \ldots \vee B_m$

Kehtib $\Gamma \wedge \Lambda \Rightarrow \Gamma \vee \Lambda$. (*disjunktsioon järeldub konjunktsioonist*)

Kuna $\Gamma \vee \Lambda$ sisaldab disjunktsiooni $\neg A_1 \vee A_1$ ja kehtivad samasused

$\neg A_1 \vee A_1 \Leftrightarrow true$ ning $true \vee A \Leftrightarrow A$,

saame $A_2 \vee \ldots \vee A_n \ \vee \ \ B_2 \vee \ldots \vee B_m$ $\square$

Faktoriseerimisreegel (abireegel)

- Kui päringus on kaks ühesugust literaali, võib neist ühe eemaldada ilma valemi tõeväärtust muutmata.

$$\frac{\leftarrow A_1, A_1, A_2, \ldots, A_n}{\leftarrow A_1, A_2, \ldots, A_n}$$

Resolutsiooni rakendusnäide

Olgu antud Horni laused disjunktsiooni kujul:

1. `isa(jaan, martin)`
2. `¬ isa(riivo, veiko)`
3. `isa(martin, veiko) ∨ isa(riivo, veiko)`
4. `¬ isa(martin, veiko) ∨ vanaisa(jaan,veiko)`

Kas lausetest 1. – 4. saab tuletada lause `vanaisa(jaan,veiko)`?

s.t. kas laused 1. – 4. koos negatiivse literaaliga

5. `¬ vanaisa(jaan,veiko)`

on vastuolulised ehk kas neist saab tuletada tühja disjunkti?

Resolutsiooni rakendusnäide

Olgu antud Horni laused disjunktsiooni kujul:

1. `isa(jaan, martin)`
2. `¬ isa(riivo, veiko)`
3. `isa(martin, veiko) ∨ isa(riivo, veiko)`
4. `¬ isa(martin, veiko) ∨ vanaisa(jaan,veiko)`

Kas lausetest 1. – 4. saab tuletada lause `vanaisa(jaan,veiko)`?

s.t. kas laused 1. – 4. koos negatiivse literaaliga

5. `¬ vanaisa(jaan,veiko)`

on vastuolulised ehk kas neist saab tuletada tühja disjunkti?

Laused **2** ja **3** annavad uue lause **6.** `isa(martin, veiko)`

Resolutsiooni rakendusnäide

Peale esimest RR rakendamist on teadmusbaasis laused:

1. isa(jaan, martin)
2. ¬ isa(riivo, veiko)
3. isa(martin, veiko) ∨ isa(riivo, veiko)
4. ¬ isa(martin, veiko) ∨ vanaisa(jaan,veiko)
5. ¬ vanaisa(jaan,veiko)
6. isa(martin, veiko)

Rakendame resolutsiooni lausetele **4** ja **6**, saame uue lause

7. vanaisa(jaan,veiko)

Resolutsiooni rakendusnäide

Peale teist RR rakendamist on teadmusbaasis laused:

1. `isa(jaan, martin)`
2. `¬isa(riivo, veiko)`
3. `isa(martin, veiko) ∨ isa(riivo, veiko)`
4. `¬ isa(martin, veiko) ∨ vanaisa(jaan,veiko)`
5. `¬ vanaisa(jaan,veiko)`
6. `isa(martin, veiko)`
7. `vanaisa(jaan,veiko)`

Rakendades resolutsiooni lausetele **7** ja **5**, saame tühja disjunkti!

Järelikult kehtib lause `vanaisa(jaan,veiko)`

Resolutsioon muutujaid sisaldavate Horni lausetega

<u>Näide</u>

Olgu lausete hulk Γ:

```
isa(jaan, peeter).
isa(jaan, martin).
isa(martin, veiko).
isa(riivo, leo).
ema(leena, leo) ∨ isa(leena, leo).
¬ isa(leena, leo).
¬ isa(X, Y) ∨ ¬ isa(Y, Z) ∨ vanaisa(X, Z).
¬ isa(X, Y) ∨ ¬ ema(Y, Z) ∨ vanaisa(X, Z).
```

ja valem ***A***, mida tahame tõestada hulgal Γ**,** kus

A**:** `vanaisa(jaan, veiko)`

Seega tõestame, et kehtib $\Gamma \rightarrow$ ***A*****:**

Resolutsioon muutujaid sisaldavate Horni lausetega

- Eelnevast teame, et kehtib samasus $\boldsymbol{\Gamma} \rightarrow \boldsymbol{A} \equiv \neg\, \boldsymbol{\Gamma} \vee \boldsymbol{A}$,
- ehk valem $\boldsymbol{\Gamma} \rightarrow \boldsymbol{A}$ kehtib parajasti siis, kui tema eitus $\neg(\boldsymbol{\Gamma} \rightarrow \boldsymbol{A}) \equiv$

  $\equiv \neg\,(\neg\, \boldsymbol{\Gamma} \vee \boldsymbol{A}) \equiv \boldsymbol{\Gamma} \wedge \neg\, \boldsymbol{A}$ on vasturääkiv.
- Seega vasturääkivuse näitamiseks laiendame hulka $\boldsymbol{\Gamma}$, lisades valemi $\neg\, \boldsymbol{A}$, saame lausete hulga $\boldsymbol{\Gamma}' = \boldsymbol{\Gamma} \cup \neg\, \boldsymbol{A}$:

  Näite puhul: $\boldsymbol{\Gamma}' = \boldsymbol{\Gamma} \cup \neg$ `vanaisa(jaan, veiko)`
- Tuletame $\boldsymbol{\Gamma}'$-st vastuolu (e. tühja disjunkti) kasutades selleks resolutsiooni.

- Kuidas aga rakendada resolutsiooni, kui Horni laused sisaldavad *muutujaid*?

Resolutsioon muutujaid sisaldavate Horni lausetega

- Muutujat V sisaldav HL tähistab *kõigi* lausete hulka, milles V on asendatud konkretiseeriva termiga.

<u>Näide (muutujate X, Y, Z konkretiseerimine):</u>

Lause `¬ isa(X, Y) ∨ ¬ isa(Y, Z) ∨ vanaisa(X, Z).`

konkretiseeringud on:

1) `¬isa(leo,jaan) ∨ ¬isa(jaan,martin) ∨ vanaisa(leo, martin).`

2) `¬isa(leo,peeter) ∨ ¬isa(peeter,martin) ∨ vanaisa(leo,martin).`

kui `X∈{leo}, Y∈{jaan, peeter}, Y∈{martin}`

Resolutsioon muutujaid sisaldavate Horni lausetega

- Kuidas leida HL konkretiseering, mida kasutades saab rakendada resolutsiooni?

- Olgu reeglis

$$\frac{\neg A_1 \vee A_2 \vee \ldots \vee A_n \qquad A'_1 \vee B_2 \vee \ldots \vee B_m}{A_2 \vee \ldots \vee A_n \vee\ B_2 \vee \ldots \vee B_m}$$

- literaalid A_1 ja A_1', mis erinevad ainult nendes esinevate muutujate nimede poolest.
- Tuletame meelde muutujate interpretatsiooni reeglis:

$$\forall x_1 \ldots x_n \,\exists y_1 \ldots y_m\, z_1 \ldots z_l :\ P(x_1 \ldots x_n) \leftarrow Q_1(y_1 \ldots y_m) \wedge \ldots \wedge Q_n(z_1 \ldots z_l)$$

- Et rakendada resolutsiooni, peab muutujad literaalides A_1 ja A_1' konkretiseerima nii, et literaalid oleksid ühesugusel kujul ehk *unifitseeritud.*

Resolutsioon muutujaid sisaldavate Horni lausetega

Näide (järg - unifitseerimine):

Et rakendada resolutsioonireeglit lausetele

`isa(jaan, martin)`

`¬ isa(X, Y) ∨ ¬ isa(Y, Z) ∨ vanaisa(X, Z),`

tuleb nende literaalid unifitseerida. Unifitseerimine toimub *muutujate asendamise* teel konkretiseerivate termidega.

Muutujate asendus e. substitutsioon: $\{x_1/t_1, \ldots , x_n/t_n\}$, kus

x_i – asendatav muutuja

t_i – asendav term, kus t_i on erinev x_i –st

Selgitus: Näites kasutame asendust {X/jaan, Y/martin}

Lause `isa(jaan, martin)` on juba konkreetne, sest konstante ei saa asendada,
Rakendades seda asendust teisele lausele, saame

`¬isa(jaan,martin)∨ ¬isa(martin,Z)∨ vanaisa(jaan,Z).`

Termide unifitseerimine

- Asendus on funktsioon:

$$(term_1,\ term_2) \rightarrow asenduste\ \ vektor$$

<u>Definitsioon</u> (Termide unifitseeruvus):

Termid t_1 ja t_2 on unifitseeruvad, kui leidub asendus σ, nii et $\sigma(t_1) = \sigma(t_2)$

Termide unifitseerimine

<u>Näide (järg)</u>: kasutame unifitseerijat `{X/jaan, Y/martin}`

Peale `X` ja `Y` unifitseerimist ja RR reegli rakendamist, saame lause

7''. ¬ `isa(martin, Z)` ∨ `vanaisa(jaan, Z)`.

Püüame unifitseerida 3 ja 7'', kus

3. `isa(martin,veiko)`.

7''. ¬ `isa(martin, Z)` ∨ `vanaisa(jaan, Z)`.

Unifitseerija `{Z/veiko}` annab uue lause

7'''. ¬ `isa(martin, veiko)` ∨ `vanaisa(jaan, veiko)`

ning RR reegli abil tuletame väidetest 3. ja 7'''.

9. ¬ `vanaisa(jaan, veiko)` % tõestatava väite eitus

10. `vanaisa(jaan, veiko)`

9. ja 10. annavad RR põhjal tühja disjunkti e. vastuolu.

Termide unifitseerimine

- Kuidas unifitseerida, kui unifitseeritavad termid sisaldavad muutujaid?

Minimaalse substitutsiooni reegel:

Asendatakse ainult need muutujad, mille asendamine võimaldab otseselt rakendada resolutsiooni.

Näide:

Unifitseerime $\underline{P(a, X, Y)} \vee S(X,Y)$ ja $\underline{\neg P(U, V, b)} \vee R(U,V)$

kasutades asendust $\{U/a, Y/b, X/V\}$.

saame $\underline{P(a, V, b)} \vee S(V, b)$ ja $\underline{\neg P(a, V, b)} \vee R(a,V)$

RR rakendamine annab $S(V, b) \vee R(a,V)$

Kõige üldisem unifitseerija - mgu

NB! Asendused tehakse ühesuguse funktori ja aarsusega termide sees vasakult paremale ja asendus tehakse teadmusbaasi **kõikides** Horni lausetes.

Definitsioon (**Kõige üldisem unifitseerija** - mgu):

Asendus ρ on termide t_1 ja t_2 kõige üldisem unifitseerija, kui

1. ρ on termide t_1 ja t_2 unifitseerija
2. Termide t_1 ja t_2 iga teise unifitseerija $\sigma \neq \rho$ korral leidub veel asendus τ, nii et $\sigma = \tau \bullet \rho$

st. iga termi t korral $\sigma(t) = \tau(\rho(t))$.

$\bullet$ – tähistab unifitseerijate kompositsiooni (rakendamist paremalt vasakule)

Kõige üldisem unifitseerija - mgu

<u>Märkus:</u>

mgu annab kõige vähem konkretiseeritud, kuid siiski veel unifitseeruvad termid:

Seega

mgu on funktsioon: $(term_1, term_2) \rightarrow \min(asenduste\ \ vektor)$

Unifitseerimise näiteid

- mgu(P(a, X), P(Y, b)) = {Y/a, X/b}
- mgu(P(X, f(X)), P(f(Y), U)) = {X/f(Y), U/f(f(Y))}
- mgu(L(g(X,X)), L(g(f(a), f(a)))) = {X/f(a)}
- mgu(R(a, b), R(a, b)) = {} % Siin puuduvad muutujad, mida asendada

Resolutsioon unifitseeritavatel literalidel

Üldistame RR reegli Horni lausetele, kus literaalid sisaldavad muutujaid:

Unifitseerimisega resolutsiooni reegel:

$$\frac{\neg A_1 \vee A_2 \vee \ldots \vee A_n \qquad B_1 \vee B_2 \vee \ldots \vee B_m}{(A_2 \vee \ldots \vee A_n \ \vee \ B_2 \vee \ldots \vee B_m)\sigma} \quad \sigma = \mathrm{mgu}(A_1, B_1)$$

Horni lausetes, millele rakendatakse resolutsiooni reeglit, tuleb eelnevalt unifitseerida mgu-ga kõik termid

Kõige üldisem unifitseerija - mgu

Lause:

Leidub algoritm mgu, mis arvutab literalide ja termide *x* ja *y* kõige üldisema unifitseerija.

- Lause tõestuseks esitame järgnevalt mgu-algoritmi

Algoritm mgu leidmiseks (1)

```
substitution mgu(x,y)
term x,  y;                                          % x, y - unifitseeritavad termid
{ int i;                                             %  i - termi argumentide indeksmuutuja
 Substitutions list g;                               %  g - asenduste list
% Case1: termideks on muutujad, konstandid või termid on erineva pikkusega

if x==y return EMPTYSUBST;                           %  kui x ja y on ühesugused
else if variable(x) return mgu_var(x,y);             % Kui x on muutuja
else if variable(y) return mgu_var(y,x);             % Kui y on muutuja
else if (constant(x)&& constant(y)) return NOSUBST;  % Kui  x ja y
                                                     % on erinevad konstandid
else if (length(x) != length(y))                     % Kui aarsus erinev,
return NOSUBST;                                      % siis ei ole unifitseeritavad
```

Algoritm mgu leidmiseks (2)

% Case 2: ühepikkuste mitmekohaliste termide unifitseerimine

```
i=0;                       % Algväärtustame alamtermide indeksmuutuja
g=EMPTYSUBST;              % Algväärtustame asenduste listi
while (i<=length(x))       % iteratsioon üle x ja y alamtermide
{
   s= mgu(subterm(x,i), subterm(y,i)); % x ja x i-nda alamtermi võrdlus
   if s==NOSUBST return s;             % Kui puudub asendus, siis s={}
   else                                % Kui leidub asendus
      g=compose(g,s);            % Täiendada unifitseerijat g asendusega s
      x=substitute(x,g);         % Rakendada unifitseerijat g termile x
      y=substitute(y,g);         % Rakendada unifitseerijat g termile y
i= i+1;
}
return g;
}
```

Case 3: Muutujate asendatavuse kontroll

```
substitution mgu var(x,y)
term x,y;
{
        if occurs_in(x,y) return NOSUBST;
                % Muutuja ei tohi esineda temaga unifitseeritavas termis, sest siis
                % tekib "ringasendus"
        else return makesubstitution(x,y);
}
```

Näide:

$\mathrm{mgu}(\mathrm{P}(a, X, X),\ \ \mathrm{P}(Y, b, Y))$ - siin ei eksisteeri unifitseerivat asendust

$\{X \backslash b\}$ $\{Y \backslash a\}$

$\mathrm{P}(a, b, \mathbf{b}) \neq \mathrm{P}(a, b, \mathbf{a}))$

Unifitseerimise harjutusi

Missugused alltoodud asendustest on lubatud ja otstarbekad?

- $\{x/y,\ y/x\}$
- $\{x/x,\ y/x\}$
- $\{x/y,\ x/z\}$
- $\{x/f(x,y),\ y/\mathrm{g}(z)\}$
- $\{x/a,\ y/x\}$
- $\{z/a,\ b/c\}$

Rakendada asendust $\{x/h(y,\ a),\ y/b,\ z/g(c)\}$ järgmistele termidele

- $f(x,y,z)$
- $f(a,b,c)$
- $h(x,x)$
- $f(g(h(x),v),\ g(x,y),\ g(z,\ f(x,y,u)))$

Märkus: Alfabeedi lõpuosa sümbolid tähistavad muutujaid ja algusosa sümbolid konstante!

Unifitseerimise harjutusi (vastused)

Missugused alltoodud asendustest on lubatud ja otstarbekad?

- $\{x/y, y/x\}$ - ei anna reaalset efekti ja võib tekitada vigu, kui termid sisaldavad *y*
- $\{x/x, y/x\}$ - x/x on kasutu
- $\{x/y, x/z\}$ - x/z on kasutu peale asendust x/y
- $\{x/f(x,y), y/g(z)\}$ – ringasendus, sest $f(x,y)$ sisaldab asendatavat muutujat
- $\{x/a, y/x\}$ – ainult asendus x/a annab efekti
- $\{z/a, b/c\}$ – b/c on keelatud asendus (konstandi asendamine konstandiga)

Asenduse $\sigma=\{x/h(y, a), y/b, z/g(c)\}$ rakendamine termidele

- $f(x, y, z) \rightarrow_\sigma f(h(b, a), b, g(c))$
- $f(a, b, c) \rightarrow_\sigma f(a,b,c)$ % puuduvad asendatavad muutujad
- $h(x, x) \rightarrow_\sigma h(h(b, a), h(b, a))$
- $f(g(h(x),v), g(x,y), g(z, f(x,y,u))) \rightarrow_\sigma f(g(h(h(b,a),v), g(h(b, a),b), g(g(c)\},$
  $f(h(b, a),b,u)))$

Unifitseerimise harjutusi

Leida mgu termidele:

$$\text{mgu}(p(f(y),w,\ g(z)),\ \ p(z,\ h(z,w),\ f(v))) = ?$$

$$\text{mgu}(p(a,\ x,\ f(g(y))),\ p(u,\ u,\ v)) = ?$$

Unifitseerimise harjutusi (vastus)

Leida mgu termidele $p(f(y),w, g(z))$ ja $p(z, h(z,w), f(v))$

Lahenduskäik:

$$p(f(y),w, g(z)) \qquad p(z, h(z,w), f(v))$$

$$\downarrow z\backslash f(y)$$

$$(p(f(y),w, g(f(y))) \qquad p(f(y), h(f(y),w), f(v)))$$

$$\downarrow \cancel{w\backslash h(z,w)} \quad \leftarrow \text{ringasendus}$$

mgu puudub

Unifitseerimise harjutusi (vastus)

- Leida mgu termidele $p(a, x, f(g(y)))$ ja $p(u, u, v)$
- Lahenduskäik:

$$
\begin{array}{ccc}
p(a, x, f(g(y))) & & p(u, u, v) \\
& \big\downarrow u\backslash a & \\
p(a, x, f(g(y))) & & p(a, a, v) \\
& \big\downarrow x\backslash a & \\
p(a, a, f(g(y))) & & p(a, a, v) \\
& \big\downarrow v\backslash f(g(y)) & \\
p(a, a, f(g(y))) & & p(a, a, f(g(y)))
\end{array}
$$

Vastus:

$$\mathrm{mgu}(p(a, x, f(g(y))), p(u, u, v)) = \{u\backslash a, x\backslash a, v\backslash f(g(y))\}$$