JOHDANTO KURSSIN AIHEPIIRIIN

1. Näkemyksiä loogiseen päättelyyn
2. Sovelluksia tietojenkäsittelyssä
3. Joitain esitietoja

1. NÄKEMYKSIÄ LOOGISEEN PÄÄTTELYYN

- Ihmisen suorittama päättely
- Formaali päättely
- Automaattinen/mekaaninen päättely

Ihmisen suorittama päättely

Päättely on usein erilaisten syiden ja seurausten analysointia.

Esimerkki.

Hemmo tulee töihin omalla autollaan ainoastaan silloin, kun sataa tai bussinkuljettajat ovat lakossa. Muutoin hän kulkee töihin bussilla.

Hemmo tuli tänään töihin omalla autollaan.

Bussinkuljettajat ovat tänään normaalisti töissä.

Tänään sataa.

Huomio. Ihmisen päättelykyky ja päättelyyn käytettävissä olevat resurssit (aika ja muisti) ovat usein rajalliset.

Formaali päättely

Matemaattinen logiikka tekee loogisesta päättelystä formaalin:

- väittämät esitetään formaalilla kielellä ja
- johtopäätösten hyväksyttävyydelle annetaan eksaktit kriteerit.

Malli on ideaalinen verrattuna ihmisen suorittamaan päättelyyn.

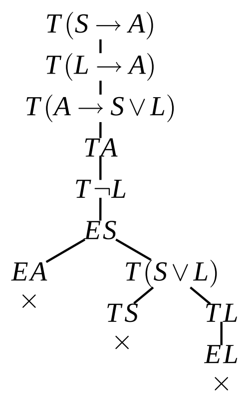
Esimerkki. Formalisoidaan edellisen esimerkin väittämät:

$$\{ \quad S \rightarrow A, \quad L \rightarrow A, \quad A \rightarrow S \vee L, \quad A, \quad \neg L \quad \}.$$

Näistä voidaan päätellä S vaikkapa semanttisten taulujen menetelmällä.

$\Longrightarrow$ Päättely voidaan nähdä merkkijonojen käsittelynä ao. kielessä.

Esimerkki todistuksesta (semanttinen taulu)



Huomio. Todistuksen muoto ei riipu atomisille väittämille S, A ja L annetuista merkityksistä (suoritetut päätelmät ovat abstrakteja).

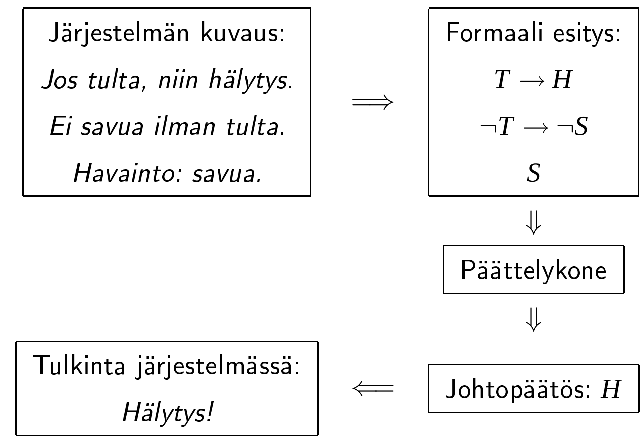
Automaattinen/mekaaninen päättely

Toteutetaan looginen päättely tietokoneohjelmana (päättelykone).

Esimerkki. Suoritetaan vastaava päättely eräällä automaattisella teoreemantodistimella (`http://www.cs.unm.edu/~mccune/otter/`):

Syötetiedosto:	Osa tulosteesta:
`set(auto).`	`------------ PROOF ------------`
`formula_list(usable).`	`3 [] -A|S|L.`
`S -> A.`	`4 [] -L.`
`L -> A.`	`5 [] -S.`
`A -> S|L.`	`6 [] A.`
`A. -L. -S.`	`8 [hyper,6,3,unit_del,5,4] $F.`
`end(list).`	`-------- end of proof ---------`

Esimerkki. Tarkastellaan yksinkertaisen hälyttimen periaatteita:



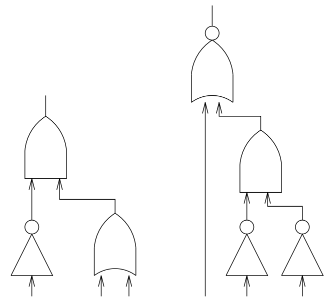
2. SOVELLUKSIA TIETOJENKÄSITTELYSSÄ

Voidaan nähdä karkea kahtiajako:

1. **Päättelykomponentti järjestelmän osana: esim.**
   – Ehtolausekkeiden ajonaikainen evaluointi
   – Kyselyjen evaluointi (tietokannat, hakukoneet, . . .)
   – Logiikkaohjelmointi (PROLOG)
   – Rajoiteohjelmointi
2. **Järjestelmän ominaisuuksien määrittely ja analysointi: esim.**
   – Ehtolausekkeiden muokkaaminen ohjelmia kehitettäessä
   – Ohjelmien vaatimusmäärittelyt ja oikeellisuustarkastelut
   – Ohjelmien synteesi

Esimerkki: Kombinatoriset piirit

Laskevatko seuraavat kombinatoriset piirit samat funktiot?



$\Longrightarrow$ Ovatko seuraavat lauselogiikan lauseet ekvivalentit?

$\neg A \wedge (B \vee C)$ ja $\neg(A \vee (\neg B \wedge \neg C))$

Esimerkki: Ohjelmien esi- ja jälkiehdot

Olkoon P seuraava ohjelma:

```
a = 0;
z = 0;
while (a != y) {
  z = z + x;
  a = a + 1;
}
```

Esiehto: `y>=0`

Jälkiehto: `z==x*y`

Halutaan osoittaa, että `[y>=0] P [z==x*y]` eli jos ohjelman P suoritus alkaa esiehdon vallitessa, niin jälkiehto on tosi suorituksen päättyessä.

Esimerkki: Ohjelmien ehtolausekkeet

Olkoon `myptr` tyyppiä "`char *`" C-kielessä.

```
/* TAPA 1 */
if(myptr != NULL && myptr[0] == '/') dothis(myptr);
if(!(myptr == NULL || myptr[0] == '.')) dothat(myptr);

/* TAPA 2 */
if(myptr != NULL) {
  if(myptr[0] == '/') dothis(myptr);
  if(myptr[0] != '.') dothat(myptr);
}
```

Kutsutaanko funktioita `dothis(myptr)` ja `dothat(myptr)` täsmälleen samoilla ehdoilla?

Esimerkki: Deduktiiviset tietokannat

linkki(otaniemi, tapiola)

linkki(otaniemi, lehtisaari)

$\forall x \forall y(\text{linkki}(x,y) \rightarrow \text{linkki}(y,x))$

$\forall x \forall y(\text{linkki}(x,y) \rightarrow \text{yhteys}(x,y))$

$\forall x \forall y \forall z(\text{yhteys}(x,y) \wedge \text{yhteys}(y,z) \rightarrow \text{yhteys}(x,z))$

Kysymyksiä:

- Kuinka selvitetään vastaus kyselyyn yhteys(tapiola, lehtisaari)?
- Entä perinteisellä relaatiotietokannalla (SQL-kysely)?

Esimerkki: Logiikkaohjelmointi (PROLOG)

```
append([],L,L).
append([A|T],L,[A|S]) :- append(T,L,S).

?- append([1,2,3],[4,5,6],X).
X = [1,2,3,4,5,6]

?- append([1,X,3],Y,[1,2,3,4]).
X = 2, Y=[4]
```

Robert Kowalski (1979): "Algorithm = Logic + Control"

Esimerkki: Järjestelmien määrittely

Matkakorttijärjestelmän kortinlukijan valoilla on seuraavat merkitykset:

1. Vihreä valo: kausilippu voimassa / arvolippu maksettu / vaihto voimassa.
2. Vihreä ja keltainen valo: kautta jäljellä 3 täyttä päivää tai vähemmän / arvoa jäljellä 5 euroa tai vähemmän.
3. Punainen valo: kausi / vaihto ei voimassa, muu virhe.

Onko mahdollista, että kaikki valot palavat yhtäaikaisesti?

Esimerkki: Rajoiteohjelmointi

Mitä vikaa on oheisessa SuDoku-tehtävässä?

1	9	3	8	6	7	4	2	**5**
4	6	8	5	**3**	2	9	1	7
7	5	**2**	1	**4**	9	6	8	3
6	2	1	4	7	3	5	9	8
5	**3**	**4**	9	1	8	**7**	6	2
9	8	7	**2**	5	**6**	3	4	**1**
2	1	6	3	9	**5**	8	7	4
8	**7**	5	6	2	4	1	**3**	9
3	4	9	7	8	**1**	2	5	6

3. JOITAIN ESITIETOJA

- Induktioperiaate luonnollisille luvuille
- Induktioperiaatteen muunnelmia
- Joukko-opin peruskäsitteet
- Joukkojen väliset perusoperaatiot
- Relaatiot
- Funktiot

Vrt. Pekka Orposen luentomoniste: "*Tietojenkäsittelyteorian perusteet*" ja erityisesti kappaleet 1.1–1.3 ja 1.5.

Induktioperiaate luonnollisille luvuille

Luonnollisten lukujen joukko määritellään induktiivisesti:

- 0 on luonnollinen luku ja
- jos n on luonnollinen luku, niin myös $n+1$ on luonnollinen luku.

Jos halutaan osoittaa, että kaikilla luonnollisilla luvuilla n on jokin ominaisuus P, sovelletaan seuraavaa periaattetta:

Määritelmä. Induktioperiaate.

Olkoon $P(n)$ luonnolliselle luvulle n määritelty ominaisuus.

Jos $P(0)$ ja $\forall n \in \mathbb{N} : (P(n) \to P(n+1))$, niin $\forall n \in \mathbb{N} : P(n)$.

Induktioperiatteen muunnelmia

- **Täydellinen induktio** (induktio-oletusta vahvennettu):

  Jos $P(0)$ ja $\forall n \in \mathbb{N} : (\forall m \in \mathbb{N} : (m < n+1 \to P(m)) \to P(n+1))$, niin $\forall n \in \mathbb{N} : P(n)$.

  Esimerkki. Jokainen luonnollinen luku $n > 1$ voidaan kirjoittaa alkulukujen tuloksi.

- **Yhtäaikainen induktio k:n eri ominaisuuden $P_1, \ldots, P_k$ suhteen:**

  Jos $P_1(0), \ldots, P_k(0)$ ja
  $\forall n \in \mathbb{N} : \forall i \in \{1, \ldots, k\} : (\forall j \in \{1, \ldots, k\} : P_j(n) \to P_i(n+1))$,
  niin $\forall n \in \mathbb{N} : P_1(n), \ldots, \forall n \in \mathbb{N} : P_k(n)$.

  Käyttökelpoinen tilanteissa, joissa $P_1, \ldots, P_k$ riippuvat toisistaan.

Esimerkki

Todistetaan, että kaikille luonnollisille luvuille n pätee

$$2^0 + 2^1 + \ldots + 2^n = 2^{n+1} - 1.$$

Perustapaus: $2^0 = 1$ ja $2^1 - 1 = 2 - 1 = 1$.

Induktiohypoteesi: $2^0 + 2^1 + \ldots + 2^{n-1} = 2^n - 1$

Induktioaskel: $2^0 + 2^1 + \ldots + 2^n = (2^0 + 2^1 + \ldots + 2^{n-1}) + 2^n =$
$(2^n - 1) + 2^n = 2 \times 2^n - 1 = 2^{n+1} - 1.$

Tuloksen sovellus tietojenkäsittelyssä: täydellisessä binääripuussa on lehtisolmuja yksi enemmän kuin sisäsolmuja.

- Jos kysymyksessä on hakupuu, niin puun syvyyden n kasvattaminen $n+1$:een johtaa työmäärän kaksinkertaistumiseen.

Joukko-opin peruskäsitteet

Tarkastellaan joukkoja $A = \{a, b\}$ ja $B = \{b, c\}$.

- Joukon jäsenyys: $a \in A$, $b \in A$ ja $c \notin A$
- Joukkojen yhtäsuuruus: $A \neq B$
  (koska A:lla ja B:llä ei ole samat alkiot).
- Tyhjä joukko: $\emptyset$ (tai vaihtoehtoisesti $\{\}$)
- Osajoukko: $\emptyset \subseteq A$, $A \not\subseteq B$ ja $B \not\subseteq A$.
- Aito osajoukko: $\emptyset \subset A$
- Joukkojen kardinaliteetti: $|\emptyset| = 0$ ja $|A| = |B| = 2$

Joukkojen väliset perusoperaatiot

Tarkastellaan edelleen joukkoja $A = \{a, b\}$ ja $B = \{b, c\}$:

- Joukkojen unioni: $A \cup B = \{a, b, c\}$
- Joukkojen leikkaus: $A \cap B = \{b\}$
- Joukkojen erotus: $A \setminus B = \{a\}$ ja $B \setminus A = \{c\}$
- Karteesinen tulo: $A \times B = \{\langle a,b\rangle, \langle a,c\rangle, \langle b,b\rangle, \langle b,c\rangle\}$
- Potenssijoukko (eli joukon kaikki osajoukot):

$$\mathbf{2}^A = \{\emptyset, \{a\}, \{b\}, \{a,b\}\} \text{ ja } |\mathbf{2}^A| = 2^{|A|} = 2^2 = 4.$$

Huomio. Joukko $\mathbf{2}^{\emptyset} = \{\emptyset\}$ eli joukko, jonka alkiona on tyhjä joukko!

Määritelmä. Binäärirelaatio $R \subseteq A \times A$ on

- *refleksiivinen*, joss kaikille $a \in A$ pätee $\langle a,a\rangle \in R$.
- *irrefleksiivinen*, joss kaikille $a \in A$ pätee $\langle a,a\rangle \notin R$.
- *symmetrinen*, joss kaikille $a \in A$ ja $b \in A$ pätee:

  jos $\langle a,b\rangle \in R$, niin $\langle b,a\rangle \in R$.
- *asymmetrinen*, joss kaikille $a \in A$ ja $b \in A$ pätee:

  jos $\langle a,b\rangle \in R$, niin $\langle b,a\rangle \notin R$.
- *transitiivinen*, joss kaikille $a \in A$, $b \in A$ ja $c \in A$ pätee:

  jos $\langle a,b\rangle \in R$ ja $\langle b,c\rangle \in R$, niin $\langle a,c\rangle \in R$.
- *ekvivalenssirelaatio*, joss R on refleksiivinen, symmetrinen ja transitiivinen.

Relaatiot

- Joukkojen $A_1, \ldots, A_n$ *karteesinen tulo*
  $A_1 \times \cdots \times A_n = \{\langle a_1, \ldots, a_n\rangle \mid a_1 \in A_1, \ldots, a_n \in A_n\}$.
- Jos $A_1 = A$, ..., $A_n = A$, saadaan
  $A^n = \underbrace{A \times \cdots \times A}_{n \text{ kpl}} = \{\langle a_1, \ldots, a_n\rangle \mid a_1 \in A, \ldots, a_n \in A\}$.

  Erikoistapaukset: $A^1 = \{\langle a\rangle \mid a \in A\} = A$ ja $A^0 = \{\langle\rangle\}$.
- Joukkojen $A_1, \ldots, A_n$ välinen n-paikkainen *relaatio* R on karteesisen tulon $A_1 \times \cdots \times A_n$ osajoukko.

Esimerkki. Kaksi relaatiota luonnollisten lukujen välillä:

$\{n \in \mathbb{N} \mid n \text{ on pariton}\} \subseteq \mathbb{N}^1$ ja $\{\langle x,y,z\rangle \in \mathbb{N}^3 \mid x^2 + y^2 = z^2\} \subseteq \mathbb{N}^3$.

Funktiot

- *Funktio* $f : A_1 \times \cdots \times A_n \to A$ on relaatio $f \subseteq A_1 \times \cdots \times A_n \times A$, joka toteuttaa *funktionaalisuusehdon*:

  kaikille $a_1 \in A_1$, ..., $a_n \in A_n$ on olemassa täsmälleen yksi $a \in A$ siten, että $\langle a_1, \ldots, a_n, a\rangle \in f$ (eli f:n arvo pisteessä $\langle a_1, \ldots, a_n\rangle$).
- Kyseistä alkiota merkitään lausekkeella $f(a_1, \ldots, a_n)$.

Määritelmä. Funktio $f : A \to B$ on

- *injektio*, joss kaikille $a \in A$ ja $b \in A$ pätee $f(a) \neq f(b)$.
- *surjektio*, joss kaikille $b \in B$ on olemassa $a \in A$ siten, että $f(a) = b$.
- *bijektio*, joss f on sekä injektio että surjektio.