JOHDANTO KURSSIN AIHEPIIRIIN

1. Näkemyksiä loogiseen päättelyyn
2. Logiikan soveluskohteita tietojenkäsittelyssä
3. Joitain esitietoja

1 Näkemyksiä loogiseen päättelyyn

- **Inhimillinen päättely:**
  Esim. ihmisen suorittama syiden ja seurausten analysointi.
  Keskeisiä piirteitä: päättelykyvyn ja käytettävissä olevien resurssien (aika ja muistin määrä) rajallisuus.
- **Formaali päättely:**
  Matemaattinen logiikka tekee loogisesta päättelystä formaalin:
  - väittämät esitetään formaalilla kielellä ja
  - johtopäätösten hyväksyttävyydelle annetaan eksaktit kriteerit.

  Malli on ideaalinen (vrt. inhimillinen päättely) ja abstrakti.
  Päättely voidaan palauttaa merkkijonojen käsittelyksi ao. kielessä.
- **Automaattinen/mekaaninen päättely:**
  Toteutetaan looginen päättely tietokoneohjelmana (päättelykone).

Esimerkki. Tarkastellaan yksinkertaisen hälyttimen kuvausta:

Järjestelmän kuvaus:
Jos savua, niin hälytys.
Jos ovi auki, niin hälytys.
Havainnot: savua tai ovi auki.

$\Longrightarrow$

Formaali esitys:
$A \to C$
$B \to C$
$A \lor B$

$\Downarrow$

Päättelykone

$\Downarrow$

Johtopäätös: C

$\Longleftarrow$

Tulkinta järjestelmässä:
Hälytys!

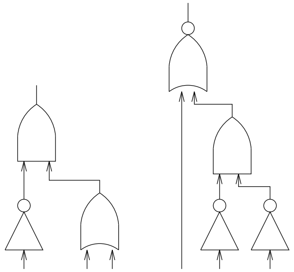
2 Logiikan soveluskohteita tietojenkäsittelyssä

Voidaan nähdä karkea kahtiajako:

- **Päättelykomponentti järjestelmän osana: esim.**
  - Ehtolausekeiden ajonaikainen evaluointi
  - Kyselyjen evaluointi (tietokannat, hakukoneet, . . .)
  - Logiikaohjelmointi (PROLOG)
  - Rajoiteohjelmointi
- **Järjestelmän ominaisuuksien määrittely ja analysointi: esim.**
  - Ehtolausekeiden muokaaminen ohjelmia kehitettäessä
  - Ohjelmien vaatimusmäärittelyt ja oikeellisuustarkastelut
  - Ohjelmien synteesi

Esimerkki. Kombinatoriset piirit

Laskevatko seuraavat kombinatoriset piirit samat funktiot?



Piirien kuvaukset lauselogiikalla: $\neg A \wedge (B \vee C)$ ja $\neg(A \vee (\neg B \wedge \neg C))$

Esimerkki. Ohjelmien ehtolausekeet

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. Ohjelmien esi- ja jälkiehdot oikeellisuustarkasteluissa

Olkoon P seuraava ohjelma:

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

Esiehto: $y \geq 0$

Jälkiehto: $z = x \cdot y$

Halutaan osoittaa, että $\{y \geq 0\}$ P $\{z = x \cdot y\}$ eli jos ohjelman P suoritus alkaa esiehdon vallitessa, niin jälkiehto on tosi suorituksen päättyessä.

Esimerkki. Deduktiiviset tietokannat

linkki(otaniemi, tapiola)

linkki(otaniemi, lehtisaari)

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

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

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

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

Esimerkki. Logiikaohjelmointi (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]
```

Vrt. yleinen näkemys: PROGRAM = LOGIC + CONTROL

Esimerkki. Järjestelmän määrittely ja ominaisuuksien analysointi

Matkakorttijärjestelmän kortinlukijan valot toimivat näin (`www.matkakortti.net`):

1. Vihreä valo: kausilippu voimassa **tai** arvolippu maksettu **tai** vaihto voimassa.
2. Vihreä ja keltainen valo: kautta jäljellä 3 täyttä päivää tai vähemmän **tai** arvoa jäljellä 10 euroa tai vähemmän.
3. Punainen valo: kausi tai vaihto ei voimassa **tai** muu virhe.

Onko mahdollista, että mikään valoista ei syty?

3 Joitain esitietoja

3.1 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 $\forall n \in \mathbb{N} : P(n)$.

Esimerkki. Todistetaan, että kaikille luonnollisille luvuille n pätee

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

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

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

Induktioaskel:
$$\begin{aligned} 2^0 + 2^1 + \ldots + 2^n &= (2^0 + 2^1 + \cdots + 2^{n-1}) + 2^n \\ &= (2^n - 1) + 2^n \\ &= 2 \times 2^n - 1 = 2^{n+1} - 1 \end{aligned}$$

Tuloksen tietojenkäsittelyllinen merkitys: täydellisessä binääripuussa on lehtisolmuja yksi enemmän kuin sisäsolmuja.

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

Induktioperiaatteesta käytetään usein tiettyjä muunnelmia.

- **Täydellinen induktio** (induktio-oletusta vahvennettu):
  Jos $P(0)$ ja $\forall n \in \mathbb{N} : (\forall m \in \mathbb{N} : (m < n+1 \rightarrow P(m)) \rightarrow P(n+1))$,
  niin $\forall n \in \mathbb{N} : P(n)$.

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

- **Yhtäaikainen induktio k:n eri ominaisuuden $P_1, \ldots, P_k$ suhteen:**
  Jos $P_1(0), \ldots, P_k(0)$ ja
  $\forall n \in \mathbb{N} : \forall i \in \{1, \ldots, k\} : (\forall j \in \{1, \ldots, k\} : P_j(n) \rightarrow P_i(n+1))$,
  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.

3.2 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 $|A| = |B| = 2$

Keskeisiä joukkojen välisiä operaatioita

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: $A - B = \{a\}$ ja $B - A = \{c\}$
- 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!

3.3 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 $A_1 = A, \ldots, A_n = A$, 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-paikainen *relaatio* R on karteesisen tulon $A_1 \times \cdots \times A_n$ osajoukko.

Esimerkki. Kaksi relaatiota luonnollisten lukujen välillä:

$\{n \mid n \in \mathbb{N} \text{ on pariton}\} \subseteq \mathbb{N}^1$

$\{\langle x, y, z\rangle \mid x^2 + y^2 = z^2,\ x \in \mathbb{N},\ y \in \mathbb{N},\ z \in \mathbb{N}\} \subseteq \mathbb{N}^3$

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.

3.4 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 $a_1 \in A_1, \ldots, a_n \in A_n$ 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 lausekeella $f(a_1, \ldots, a_n)$.

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.