

## Kotitehtävät

- Kotitehtävät ovat henkilökohtaisia.
- Tehtävien aiheet ovat seuraavat:
  1. BDD-kirjaston käyttäminen boolean lausekkeiden ekvivalenssin tarkastamiseen
  2. Suunnitteluongelman esittäminen toteutuvuusongelmana
  3. Suunnitteluongelman esittäminen logiikkaohjelmana (stabiilien mallien semantiikka)
- Tehtävien ratkaisemisessa käytetään erilaisia ohjelmistoja.
- Tarvittavat ohjelmat asennetaan myös laskentakeskuksen koneisiin.

## 1. Kotitehtävä: tehtävänanto

Selvitä, ovatko annetut Boolean funktiot ekvivalentit. Jos eivät, anna vastaesimerkki (totuusarvojakelu, jolla funktiot antavat eri arvot).

- (a) Konjunkttiivisessa normaalimuodossa olevat funktiot  $f_1$  ja  $g_1$ .
- (b) If-then-else-muodossa olevat funktiot  $f_2$  ja  $g_2$ .

Vihje: käytä ratkaisussa ohjelmia boole ja translate (lähdekoodit ja ohjeita kurssin kotisivulla).

☞ Henkilökohtaiset kotisivut (salasana lähetetään erikseen):

<http://www.tcs.hut.fi/Studies/T-79.154/students/12345X/>

- Arvostelu: hyväksytty/hylätty
- Hyväksytty suoritus: **lyhyt** työselostus, jossa
  1. käytetty käänös (ratkaisumalli)
  2. löydetty ratkaisut
  3. ajoesimerkit
- Alustava aikataulu kotitehtäville:

	Tehtävänanto	Palautuspäivä
1.	18.9.	9.10.
2.	23.10.	20.11.
3.	13.11.	4.12.

## Boole

- David Longin toteuttama BDD-kirjasto ja komentotulkki (boole).
- Käyttöesimerkki: `boole < input.txt > output.txt`
- Interaktiivinen käyttö komentotulkkina: `boole -i`
- Ohjelman suoritus päättyy syötetiedoston loppuessa (EOF).
- Lähdekoodi saatavilla verkosta osoitteella

<http://www-2.cs.cmu.edu/~modelcheck/bdd.html>

## Keskeisiä Boolean toimintoja

- Muuttujien luominen:

```
vars <id> <id> ... ;
```

- Lauseketta vastaavan BDD:n muodostaminen ja sijoitus muuttujan arvoksi:

```
<id> := <expr> ;
```

- Muuttujien poistaminen:

```
clear all ; tai clear <id> <id> ... ;
```

- Lausekkeiden tulostaminen:

```
print <expr> ; tai sop <expr> ;
```

## Erikoistoimintoja

- Lausekkeen toteuttavien totuusjakeluiden tulostaminen:

```
satisfy <expr> ;
```

- Solmujen lukumäärän rajoittaminen:

```
limit <number> ; tai limit none ;
```

- Muuttujajärjestyksen vaihtaminen:

```
reorder [sift|window|hybrid|none] ;
```

- Roskankeruu (käyttämättömien solmujen poistaminen):

```
gc ;
```

## Statistiikkaa

- BDD-esitysten solmujen lukumäärät:

```
size <expr> <expr> ... ;
```

- Lausekkeen toteuttavien totuusjakeluiden suhteellinen määrä:

```
satfrac <expr> ;
```

- Ajankäytön seuranta:

```
timer ; ja timer off ;
```

- BDD-kirjaston sisäinen tilastointi:

```
stats ;
```

## Lausekkeiden syntaksi

- Lauseke <expr> on muodoltaan jokin seuraavista:

<id>	Boolean muuttuja
0	Epätosi
1	Tosi
! <expr>	Negaatio; myös ~
<expr1> & <expr2>	Konjunktio; myös *
<expr1>   <expr2>	Disjunktio; myös +
<expr1> ^ <expr2>	Poissulkeva disjunktio
<expr1> = <expr2>	Ekvivalenssi
( <expr> )	



► Tai seuraavista:

<code>&lt;expr1&gt; ? &lt;expr2&gt; : &lt;expr3&gt;</code>	If-Then-Else
<code>&lt;expr1&gt; [ &lt;id&gt; := &lt;expr2&gt; ]</code>	Kompositio; myös rajoitus
<code>exists [ &lt;id&gt; ... ] ( &lt;expr&gt; )</code>	Kvantifiointi
<code>forall [ &lt;id&gt; ... ] ( &lt;expr&gt; )</code>	Kvantifiointi
<code>subst [ &lt;id&gt; := &lt;expr1&gt; ... ] ( &lt;expr2&gt; )</code>	Sijoitus

► Yksinkertainen esimerkki:

```
vars f x1 x2 x3;  
f := (x1 & x2) | (x2 & x3);  
print f;
```