



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 ovat käytettävissä laskentakeskuksen alfoissa hakemistossa `~tssyrjan/T-79.154/bin/`.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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

	Tehtävänanto	Palautuspäivä
1.	27.9.	25.10.
2.	18.10.	22.11.
3.	15.11.	13.12.

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio




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/~ttj/LTE/12345X/>

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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>

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



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> ;
```



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 ;
```



Erikoistoimintoja

- Lausekkeen toteuttavien totuusjaketuiden tulostaminen:
`satisfy <expr> ;`
- Solmujen lukumäärän rajoittaminen:
`limit <number> ;` tai `limit none ;`
- Muuttujajärjestyksen vaihtaminen:
`reorder [sift|window|hybrid|none];`
- Roskankero (käyttämättömien solmujen poistaminen):
`gc ;`

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



Lausekkeiden syntaksi

- Lauseke `<expr>` on muodoltaan jokin seuraavista:

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

© 2001 Teknillinen korkeakoulu, Tietojenkäsittelyteorian laboratorio



► Tai seuraavista:

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

► Yksinkertainen esimerkki:

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