

MODUL 7

FIRST ORDER LOGIC

7.1. Tujuan

Memperkenalkan kepada mahasiswa dasar-dasar bahasa pemrograman logis (SWI PROLOG). Mahasiswa diharapkan mampu menerjemahkan dan mereresentasikan kasus-kasus order logic ke dalam program komputer; mampu memahami konsep logika proposional (*Propositional Logic*) dalam menyelesaikan suatu permasalahan logika.

7.2. Dasar Teori

7.2.1. Logika Proposisi (*Propositional Logic*)

Logika Proposisi (*Propositional Logic*) menawarkan logika dalam bentuk sederhana sehingga mudah dipahami. Meskipun begitu, Logika Proposisi sudah mampu membantu menarik kesimpulan. Namun, banyak kasus yang muncul akan menjadi terlihat panjang dan rumit saat diwujudkan dalam bentuk Logika Proposisi. Dan itu bisa lebih panjang dan rumit dibandingkan problem itu sendiri.

Saya ambil contoh berikut ini. Di sebuah kelas II SD, terdapat 35 siswa. Setiap hari Senin sampai dengan Kamis, mereka mengenakan seragam merah-putih. Sedangkan hari lain, mereka mengenakan seragam pramuka. Anak tetanggaku yang bernama Amin, ada salah satu siswa kelas II SD tersebut. Hari Rabu pagi kami bertemu saat dia berangkat sekolah. Seragam apa yang dia kenakan?

Bagaimana menyelesaikan contoh tersebut dengan menggunakan Logika Proposisi?

Solusi:

Misalkan:

p: amin adalah siswa kelas II SD

q: amin mengenakan seragam merah putih

r : hari rabu

Kalimat yang bisa kita nyatakan dari cerita tersebut adalah

1 : $p \wedge r \rightarrow q$

2 : p

3 : r

Dengan ekpresi seperti itu, kita sudah bisa menarik kesimpulan tentang Amin. Tetapi banyak informasi yang tidak dinyatakan dan terlewatkan. Akibatnya, ekspresi tersebut tidak bisa digunakan untuk membuat kesimpulan tentang seragam yang dipakai Ali pada hari Rabu jika diketahui bahwa Ali juga seorang siswa kelas SD tersebut. Agar bisa membuat kesimpulan tentang Ali, kita bisa mengubahnya menjadi seperti di bawah ini:

1 : $p_1 \wedge r \rightarrow q$

2 : p_1

3 : r

4 : $p_2 \wedge r \rightarrow q$

5 : p_2

dengan p_1 berarti “amin adalah anak kelas II SD” dan p_2 berarti “ali adalah anak kelas II SD”. Bagaimana jika untuk semua siswa? Kita harus menambahkan lagi kalimat nomor 1 dan 2 dengan sebelumnya mengubah p_1 menjadi p_3 . Demikian seterusnya sampai p_{35} . Maka akan diperoleh 71 kalimat. Padahal, solusi ini hanya untuk hari Rabu saja, belum hari-hari yang lain.

Predicate: Simbol dengan Parameter

First order Logic menawarkan penggunaan simbol dengan parameter. Simbol ini dikenal sebagai predikat. Sebuah predikat didefinisikan sebagai atribut(sifat) sebuah obyek atau relasi antar obyek. Obyek-obyek tersebutlah yang dijadikan sebagai parameter predikat tersebut.

Sebagai contoh, kita kembali ke contoh sebelumnya. Untuk menyelesaikan contoh tersebut, kita menggunakan simbol p untuk menyatakan atribut **seorang siswa kelas II SD**, r untuk menyatakan atribut **nama hari**, dan q untuk menyatakan relasi **mengenakan seragam**. Definisi lengkap setiap simbol, termasuk parameternya, adalah sebagai berikut:

$p(x)$: x adalah seorang siswa kelas II SD

$r(x)$: x adalah nama hari

$q(x,y)$: x mengenakan seragam y .

Dengan definisi tersebut, jika kita ingin mengungkapkan kalimat **amin adalah seorang siswa kelas II SD, hari rabu**, dan **amin mengenakan seragam pramuka** maka dapat dinyatakan sebagai berikut:

$p(\text{amin})$

$r(\text{rabu})$

$q(\text{amin}, \text{pramuka})$

Quantifier

Selain penggunaan predikat, First Order Logic juga menawarkan *quantifier* untuk membuat kalimat logika yang lebih sederhana. Ada 2 jenis *quantifier*, yaitu *universal* dan *existential*. *Quantifier* ini berlaku terhadap parameter yang muncul di sebuah kalimat masih dalam bentuk variabel. *Universal quantifier* terhadap sebuah variabel x (disimbolkan dengan $\forall x$) berarti bahwa kalimat

tersebut berlaku untuk setiap obyek x , sedangkan *existential quantifier* (disimbolkan dengan $\exists x$) berarti berlaku untuk sebagian obyek saja.

Contoh: Menggunakan definisi untuk $p(x)$, $r(x)$, dan $q(x,y)$, berikut adalah kalimat-kalimat logika dengan menggunakan *quantifier* dan artinya:

$\forall x(p(x) \wedge r(\text{rabu}) \rightarrow q(x,\text{merah-putih}))$: untuk setiap x , jika x adalah seorang siswa kelas II SD dan pada hari Rabu maka x akan mengenakan seragam merah-putih.

$\exists x(p(x) \rightarrow \neg q(x,\text{merah-putih}))$: ada x , jika x adalah seorang siswa kelas II SD maka x tidak mengenakan seragam merah putih.

7.2.2. Contoh First Order Logic dan Penarikan Kesimpulan

Lihat kembali contoh seragam Amin di atas. Solusi untuk problem di atas adalah sebagai berikut.

Solusi:

Misalkan:

$p(x)$: x adalah seorang siswa kelas II SD

$r(x)$: x adalah nama hari

$q(x,y)$: x mengenakan seragam y .

Kalimat yang bisa kita nyatakan dari cerita tersebut adalah

1 : $\forall x(p(x) \wedge r(\text{senin}) \rightarrow q(x,\text{merah-putih}))$

2 : $\forall x(p(x) \wedge r(\text{selasa}) \rightarrow q(x,\text{merah-putih}))$

3 : $\forall x(p(x) \wedge r(\text{rabu}) \rightarrow q(x,\text{merah-putih}))$

4 : $\forall x(p(x) \wedge r(\text{kamis}) \rightarrow q(x,\text{merah-putih}))$

5 : $\forall x(p(x) \wedge r(\text{jumat}) \rightarrow q(x,\text{pramuka}))$

6 : $\forall x(p(x) \wedge r(\text{jumat}) \rightarrow q(x,\text{pramuka}))$

Jika diketahui bahwa Amin adalah seorang siswa kelas II SD dan hari rabu, maka ditambahkan kalimat berikut:

7 : $p(\text{amin}) \wedge r(\text{rabu})$

Proses penarikan kesimpulan untuk menjawab pertanyaan **apa seragam yang dipakai oleh Amin pada hari Rabu** adalah sebagai berikut:

8 : $p(\text{amin}) \wedge r(\text{rabu}) \rightarrow q(\text{amin,merah-putih})$ {Instansiasi x dengan Amin pada kalimat 3}

9 : $q(\text{amin,merah-putih})$ {Modus Ponens antara 7 dan 8}

Arti kalimat 9 adalah Amin mengenakan seragam merah-putih.

————— □

Instansiasi: membuang *quantifier* dan mengganti kemunculan setiap variabel yang terkait dengan *quantifier* tersebut dengan sebuah obyek.

Contoh yang lain: Menggunakan contoh seragam siswa kelas II SD di atas, tetapi yang ditanyakan adalah apakah Taufiq seorang siswa kelas II SD jika diketahui dia tidak mengenakan seragam pramuka pada hari Jumat.

Solusi:

Menggunakan definisi sebelumnya, kita tetap memperoleh kalimat logika sebagai berikut:

1 : $\forall x(p(x) \wedge r(\text{senin}) \rightarrow q(x,\text{merah-putih}))$

2 : $\forall x(p(x) \wedge r(\text{selasa}) \rightarrow q(x,\text{merah-putih}))$

3 : $\forall x(p(x) \wedge r(\text{rabu}) \rightarrow q(x,\text{merah-putih}))$

4 : $\forall x(p(x) \wedge r(\text{kamis}) \rightarrow q(x,\text{merah-putih}))$

5 : $\forall x(p(x) \wedge r(\text{jumat}) \rightarrow q(x,\text{pramuka}))$

6 : $\forall x(p(x) \wedge r(\text{jumat}) \rightarrow q(x,\text{pramuka}))$

Diketahui bahwa taufiq tidak mengenakan seragam pramuka pada hari Jumat. Ditambahkan kalimat-kalimat berikut:

7 : $\neg q(\text{taufiq,pramuka})$

8 : $r(\text{jumat})$

Proses penarikan kesimpulan untuk menjawab pertanyaan **apa Taufiq seorang siswa kelas II SD** adalah sebagai berikut:

9 : $p(\text{taufiq}) \wedge r(\text{jumat}) \rightarrow q(\text{taufiq,pramuka})$ {Instansiasi x dengan taufiq pada kalimat 5}

10 : $\neg(p(\text{taufiq}) \wedge r(\text{jumat}))$ {Modus Tollens antara 7 dan 9}

11 : $\neg p(\text{taufiq}) \vee \neg r(\text{jumat})$ {Hukum de Morgan untuk 10}

12 : $p(\text{taufiq}) \rightarrow \neg r(\text{jumat})$ {Ekuivalensi implikasi dengan 11}

13 : $\neg p(\text{taufiq})$ {Modus Tollens antara 8 dan 12}

Arti kalimat 14 adalah Taufiq bukan seorang siswa kelas II SD.

7.3. Penggunaan SWI-PROLOG (under MacOS atau LinuX) untuk Kasus First Order Logic

Kasus 1 :

- Buka salah satu teks editor (misalnya vi, vim, pico, gedit, notepad, etc), kemudia ketikkan program berikut dan berikan nama file **OrderLogic1.pl** pada folder local, (misalnya /Document/Data/FOL).

```
p(amin).
r(rabu).

q(X,merah-putih) :- p(X), r(senin).
q(X,merah-putih) :- p(X), r(selasa).
q(X,merah-putih) :- p(X), r(rabu).
q(X,merah-putih) :- p(X), r(kamis).
q(X,pramuka) :- p(X), r(jumat).
q(X,pramuka) :- p(X), r(sabtu).
```

- Jalankan program SWI-PROLOG melalui “terminal” dengan menuliskan perintah :

```
IZamanhuri$ /opt/local/bin/swipl <enter>

% library(swi_hooks) compiled into pce_swi_hooks 0.00 sec,
2,284 bytes
Welcome to SWI-Prolog (Multi-threaded, 32 bits, Version
5.10.4)
Copyright (c) 1990-2011 University of Amsterdam, VU Amsterdam
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free
software,
and you are welcome to redistribute it under certain
conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

?- <kursor>
```

- Setelah masuk ke dalam SWI-PROLOG, jalankan file OrderLogic1.pl dengan menuliskan perintah dibawah ini pada bagian <kursor>.

```
?- [ 'OrderLogic1.pl' ].
% OrderLogic1.pl compiled 0.00 sec, 1,944 bytes
true.

?- q(amin,X) .
X = merah-putih .
```

KESIMPULAN: Si Amin memakai baju merah-putih.

- Untuk KELUAR dari program SWI-PROLOH ketikkan.

?- **halt**.

Kasus 2 :

- Buka salah satu teks editor (misalnya vi, vim, pico, gedit, notepad, etc), kemudia ketikkan program berikut dan berikan nama file **OrderLogic2.pl** pada folder local, (misalnya /Document/Data/FOL).

```
p(amin).
r(rabu).

q(X,merah-putih) :- p(X), r(senin).
q(X,merah-putih) :- p(X), r(selasa).
q(X,merah-putih) :- p(X), r(rabu).
q(X,merah-putih) :- p(X), r(kamis).
q(X,pramuka) :- p(X), r(jumat).
q(X,pramuka) :- p(X), r(sabtu).

not q(taufiq,pramuka).
r(jumat).
```

- Jalankan program SWI-PROLOG melalui “terminal” dengan menuliskan perintah :

```
IZamanhuri$ /opt/local/bin/swipl <enter>

% library(swi_hooks) compiled into pce_swi_hooks 0.00 sec,
2,284 bytes
Welcome to SWI-Prolog (Multi-threaded, 32 bits, Version
5.10.4)
Copyright (c) 1990-2011 University of Amsterdam, VU Amsterdam
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free
software,
and you are welcome to redistribute it under certain
conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

?- <kursor>
```

- Setelah masuk ke dalam SWI-PROLOG, jalankan file OrderLogic2.pl dengan menuliskan perintah dibawah ini pada bagian <kursor>.

```
?- [ 'OrderLogic2.pl' ].
% OrderLogic2.pl compiled 0.00 sec, 1,944 bytes
true.

?- p(taufiq).
false .
```

Artinya: Si Taufiq BUKAN siswa kelas II SD.

```
?- not(p(taufiq)).  
true .
```

Artinya: Si Taufiq BUKAN siswa kelas II SD.

- Untuk KELUAR dari program SWI-PROLOH ketikkan.

```
?- halt.
```

Tugas:

- Buat dan carilah contoh First Order Logic yang lainnya, kemudian implementasikan kasus tersebut dengan menggunakan SWI-PROLOG.

Implementasikan Kasus “Hukum Pernikahan” (Lihat slide-07) menggunakan program SWI-PROLOG.