

# Soal Pra Praktikum Logika Komputasional– IF2121

## PETUNJUK PRA PRAKTIKUM:

1. Praktikum IF2121 - Logika Komputasional adalah praktikum yang bersifat **mandiri**.
2. Format file jawaban yang dikumpulkan adalah **PP01\_[NIM].pl** untuk file prolog, dan **PP01\_[NIM].txt** untuk query beserta hasilnya.
3. Sebelum menjawab tiap butir soal, **tandai** terlebih dahulu untuk **setiap poin**.

```
/*Untuk File .pl*/

/* Bagian <X> */
/* Deklarasi Fakta */
<fakta>

/* Deklarasi Rules */
predikat1(X) :- predikat2(X)
```

```
/*Untuk File .txt*/

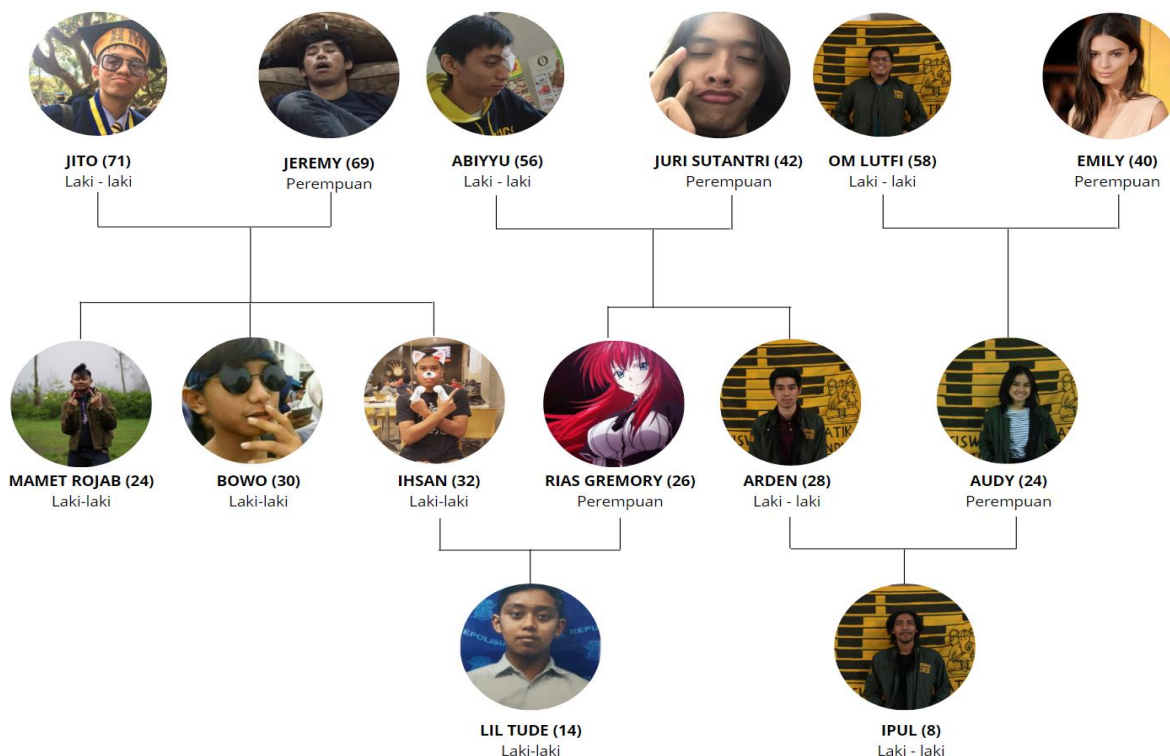
Bagian <X>

Query: <query>
<hasil query>
```

4. Semua deliverable files jawaban praktikum **dikompres** ke dalam arsip dengan ekstensi **.zip**, lalu diunggah ke website **stei.kuliah.itb.ac.id**. Format penamaan file arsip praktikum adalah **PP01\_ [NIM].zip**.
5. **Batas Waktu** pengumpulan adalah hari **Senin 4 November 2019**, pukul **12.00** waktu server.

## SOAL PRA PRAKTIKUM:

Berikut ini merupakan pohon keluarga dalam keluarga verde:



## BAGIAN I : INTRODUCTION TO PROLOG

1. Buatlah fakta-fakta dari pohon keluarga di atas dengan menggunakan **HANYA** aturan fakta di bawah ini. Tulislah dalam bahasa pemrograman prolog lalu simpan dalam file **.pl** sesuai aturan! :
  - a. pria(X) : X adalah pria
  - b. wanita(X) : X adalah wanita
  - c. usia(X,Y) : X berusia Y
  - d. menikah(X,Y) : X menikah dengan Y
  - e. anak(X,Y) : X adalah anak Y
  - f. saudara(X,Y) : X adalah saudara kandung Y
  
2. Buatlah rule/aturan di bawah ini **TANPA** membuat rule/fakta tambahan. Tulislah dalam bahasa pemrograman prolog lalu simpan dalam file **.pl** sesuai aturan! (**Boleh menggunakan rule yang sudah didefinisikan butir soal lain**):
  - a. kakak(X,Y) : X adalah kakak dari Y (baik perempuan maupun lelaki)
  - b. keponakan(X,Y) : X adalah keponakan dari Y
  - c. suami(X,Y) : X adalah suami dari Y
  - d. sepupu(X,Y) : X adalah sepupu dari Y
  - e. mertua(X,Y) : X adalah mertua dari Y
  - f. bibi(X,Y) : X adalah bibi dari Y
  - g. cucu(X,Y) : X adalah cucu dari Y
  - h. anaksulung(X) : X adalah anak paling tua
  - i. anakbungsu(X) : X adalah anak paling muda
  
3. Implementasi kalimat di bawah ini ke bentuk query prolog, kemudian tulis query dan hasilnya dalam file **.txt** sesuai aturan! (**dilarang membuat rule tambahan selain dari soal 2**)
  - a. Suami dari Rias Gremory
  - b. Paman dari Lil Tude
  - c. Menantu dari Juri Sutantri
  - d. Nenek dari Ipul
  - e. Cucu dari Om Lutfi
  - f. Ipar dari Ihsan (ipar: saudara kandung dari suami/istri)
  - g. Sepupu dari Ipul
  - h. Wanita yang merupakan anak tunggal
  - i. Pria yang belum menikah

## BAGIAN II : REKURSIF

*Magical Math Recursion* : Implementasikan *rules* berikut dalam bahasa prolog. Tulislah dalam file **.pl** sesuai dengan ketentuan :

- a. **fibbonaci(N,X)** : bilangan fibbonaci ke N adalah X dengan ketentuan  $N \geq 1$ .

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* tersebut:

```
| ?- fibbonaci(2,1) .  
true?  
yes  
| ?- fibbonaci(4,4) .  
no  
| ?- fibbonaci(4,X) .  
X = 3 ?  
yes
```

- b. **gcd(A,B,X)** : X adalah *greatest common divider* dari A dan B.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- gcd(8,12,X) .  
X= 4 ?  
yes
```

- c. **power(A,B,X)** : X adalah hasil dari A dipangkatkan dengan B.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- power(2,4,X) .  
X= 16 ?  
yes
```

### BAGIAN III : LIST

Buatlah *rule* untuk beberapa operasi berikut ini:

- a. **inverse(A,B)** : A dan B merupakan sebuah *list* yang berisi beberapa angka nilai *rules* akan bernilai yes jika A merupakan kebalikan dari *list* B dan no jika tidak.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- inverse([1,2,3,4,5],[5,4,3,2,1]).  
yes  
  
| ?- inverse([1,2,3,4,5],[5,4,3,2,2]).  
no
```

- b. **insert(X,Y,Z)** : terdapat sebuah *list* X dan sebuah angka Y. Z merupakan sebuah *list* yang bernilai sama dengan X jika angka Y sudah terdapat pada *list* X. Jika angka Y tidak terdapat di *list* X maka Z bernilai sama dengan *list* X yang telah di insert angka Y.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- insert([1,2,3],4,Z).  
Z = [1,2,3,4]  
yes  
| ?- insert([1,2,3],2,Z).  
Z = [1,2,3] ?  
yes
```

- c. **dotProduct(X,Y,Result)** : Implementasikan *rule* untuk *dot product*. Antar *list* X dan Y dengan ukuran *list* X dan Y pasti sama. Hasilnya disimpan pada Result.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- dotProduct([1,2,3],[4,5,-10],Result).  
Result = -16
```

```
yes
```

- d. **countX(L,X,N)** : *Rule* ini menghitung jumlah elemen *list* L yang bernilai X. Hasilnya akan disimpan di N.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- countX([1,3,2,5,3,2,7],2,N) .  
N = 2 ?  
yes  
| ?- countX([1,2,1,5,1,6,3,1,1,1],1,N) .  
N = 6 ?  
yes
```

- e. **palindrom(L)** : *Rule* digunakan untuk mengecek apakah sebuah *list* yaitu *list* L palindrom atau tidak.

Berikut ini adalah contoh *query* dan hasil *query* untuk *rule* berikut

```
| ?- palindrom([1,2,3,2,1]) .  
yes  
| ?- palindrom([]) .  
yes  
| ?- palindrom([1,2,2,3,2,1]) .  
no
```

## Cara memakai prolog

Instalasi GNU Prolog pada OS Linux,

```
sudo apt-get install gprolog
```

Untuk Windows dan Mac OS, dapat dilihat pada [tautan ini](#) dengan konfigurasi yang sesuai.

Prolog merupakan bahasa pemrograman untuk merepresentasikan *logic* dan bersifat *case sensitive*.

Pengetahuan atau pernyataan(*statement*) dapat dituliskan ke dalam suatu file berekstensi .pl. Pada contoh di bawah misalnya dalam **db.pl**

```
% statements
parent(X,Y) :-
    child(Y,X).
child(aby,ariel).
```

Untuk menjalankan **db.pl**, jalankan *gprolog* di direktori *file* tersebut berada.

```
| ?- [db].
```

**[db]** disesuaikan dengan nama *file* yang sesuai. Apabila *file* berhasil dikompilasi, akan menghasilkan **yes**.

Beberapa pertanyaan yang disampaikan beserta hasil yang didapatkan.

```
| ?- parent(ariel,aby).
yes
| ?- parent(aby,ariel).
no
```