

Boolean & Propotitional Logic

Matematika Diskrit RKA

Kenalan Dikit

Website : <https://abdanhafidz.com>

Linkedin : <https://www.linkedin.com/in/abdan-hafidz/>

Github : github.com/abdan_hafidz



List of Contents

- *Pengantar Logika Matematika*
- *Proposisi / Kalimat tertutup*
- *Tipe Data Boolean Python*
- *Proposisi Bertingkat / Kalimat Majemuk*
- *Negasi, Konjungsi, Disjungsi, Disjungsi Eksklusif*
- *Implikasi, Biimplikasi, dan Tautologi*
- *Clean Logical Nested If*

Disclaimer

- *Kalau aku kecepetan bilang*
- *Kalau **ga paham bertanya, jangan diam.***

01

Logika Matematika

Dasar dalam Berpikir di kehidupan kita sehari – hari.

Kenapa Anda harus punya Logika dalam Berpikir?

Dalam konteks kehidupan sehari – hari, logika menjadi dasar dalam pengambilan Keputusan, pembentukan perspektif, dan penilaian terhadap sesuatu yang objektif.

Kenapa Anda harus punya Logika dalam Berpikir?



Tenang aja bro, nanti gw nikah 🤔🤔 sama Hutao bakalan dimasakin tiap hari 🤔🤔🤔, tiap pagi gw dibangunin dan disemangatin 🤔🤔 sebelum pergi bekerja, nanti gw bakalan dipeluk dan dicium di kening 💖🤔🤔 sebelum berangkat

20.28

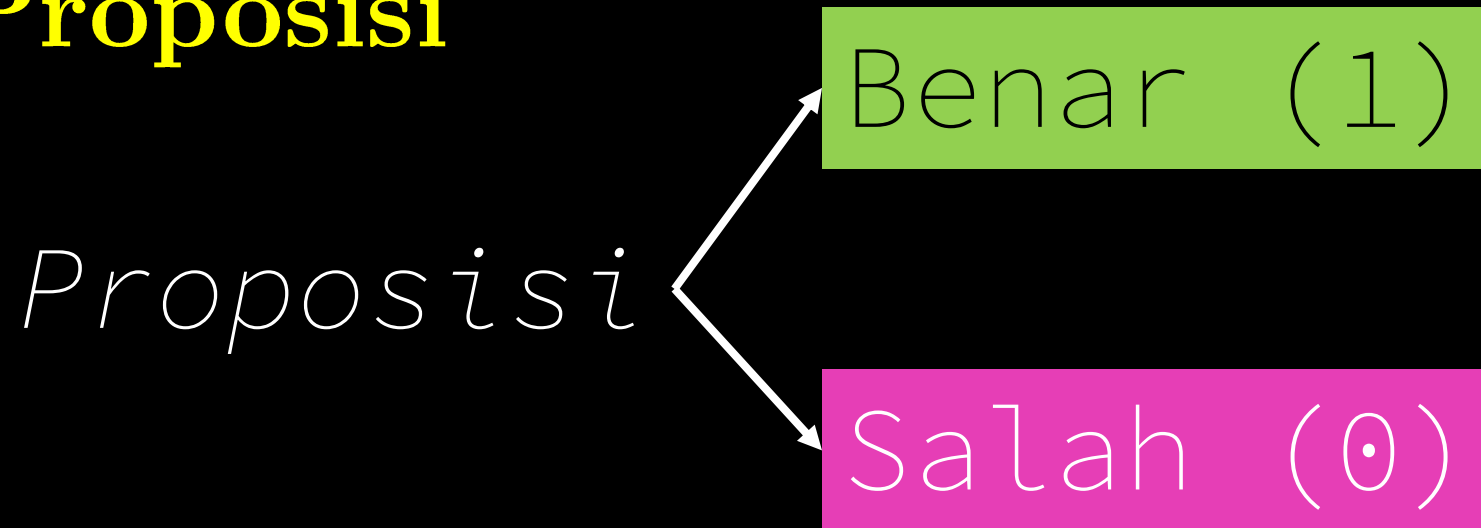
Bro itu kan cuman karakter kartun

20.28 ✓✓



Intinya Tanpa Logika Logila

Proposisi



Proposisi adalah suatu pernyataan yang dapat kita evaluasi keabsahannya apakah pernyataan tersebut benar atau salah (boolean).

Proposisi

Contoh yang merupakan Proposisi

- $1 + 1 = 2$ (pasti benar)
- 3 itu bilangan ganjil (pasti benar)
- Ada nilai x sedemikian sehingga $x + 1 = 2$ untuk x bilangan real (pasti benar)
- $1 + 3 = 5$ (pasti salah)
- $x > 9$ untuk $x = 1$ (pasti salah)

Proposisi

Tentukan kira – kira di mana kah di bawah ini yang merupakan proposisi?

- Rasya orang yang suka nonton Anime (bukan).
- Untuk semua x bilangan genap $x + 1$ selalu ganjil (proposisi)
- Besok adalah hari Kamis (bukan proposisi)
- Hari ini hari Rabu besok hari Kamis (proposisi)
- $x * x = 4$, Nilai x adalah 2 (bukan proposisi)
- Ibu Kota Indonesia adalah Jakarta (proposisi)

Negasi

Ketika kita diberikan proposi P maka kita mendapatkan negasinya adalah $\sim P \equiv \textit{not } P$. Sehingga kita mendapatkan

$$\sim(\textit{True}) \equiv \textit{False}$$

$$\sim(\textit{False}) \equiv \textit{True}$$

Negasi

Contoh Negasi :

$P = \text{“Aku suka kamu”}$

$\sim P = \text{“Aku tidak suka kamu”}$

Negasi

Kesalahan dalam Negasi :

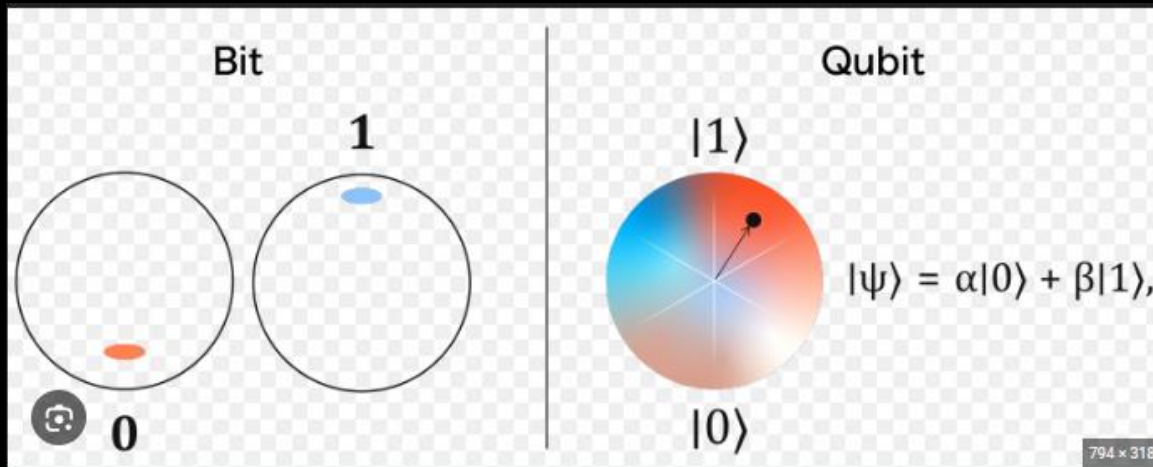
$P = \text{“Aku suka kamu”}$

$\sim P = \text{“Aku benci kamu” (salah)}$

Tidak suka belum tentu benci

Ambiguitas, Super Position, Qubit (Bonus aja)

- Menurut kabar burung Pak Camat Meninggal



Type Data Boolean Python

```
nama = "Abdan"  
jumlahPacar = 0  
jomblo_kah = bool(jumlahPacar > 0)
```

`jumlahPacar > 0` di sini merupakan sebuah proposisi

Type Data Boolean Python

```
nama = "Abdan"  
jumlahPacar = 99  
if(jumlahPacar > 0):  
    print("Lampu Merah")  
elif(jumlahPacar == 0):  
    print("Lampu hijau")  
else:  
    print("Lampu kuning")
```

If akan mengevaluasi apakah proposisi di dalam tanda kurung bernilai benar. Jika benar maka jalankan perintah di dalamnya. Jika salah jalankan perintah pengecualian "else"

Proposisi Bertingkat

*“Aku **dan** kamu harus Bersatu **atau** dunia ini kuhancurkan”*



*“Diriku mencintai dia **tetapi** dirinya tidak mencintai aku”*

*“**Jika** aku lapar, **maka** aku makan”*

Konjungsi (and, \wedge , &&)

Misalkan kita diberikan proposisi “ p dan q ”.

“Mahasiswa harus membawa STNK
dan membawa SIM”

Bawa STNK	Bawa SIM	Memenuhi Aturan
		
		
		
		

Konjungsi (and, ^, &&)

Misalkan kita diberikan proposisi “ p dan q ”.

p = membawa STNK

q = membawa SIM

Pernyataan p dan q
hanya akan dianggap
Benar jika dan hanya
jika p dan q
keduanya bernilai
benar

p	q	$p \wedge q$
B	B	B
B	S	S
S	S	S
S	B	S

Konjungsi (and, \wedge , &&)

Misalkan kita diberikan proposisi “ p dan q ”.

Pernyataan p dan q hanya akan dianggap Benar jika dan hanya jika p

dan q keduanya

bernilai benar



Disjungsi (or , \vee , $||$)

Misalkan kita diberikan proposisi “ p atau q ”.

“Untuk sebuah himpunan $A = \{2,6,8,9,10\}$. A adalah himpunan yang memuat anggotanya yaitu bilangan yang habis dibagi 2 atau 3”

Himpunan A mengandung bilangan habis dibagi 2 = $\{2,8,10\}$

Himpunan A mengandung bilangan habis dibagi 3 = $\{6,9\}$

Himpunan A mengandung bilangan habis dibagi 2 dan 3 = $\{6\}$

Disjungsi (or, \vee , $||$)

Misalkan kita diberikan proposisi “ p atau q ”.

p = bilangan habis dibagi 2

q = bilangan habis dibagi 3

Pernyataan p atau q
akan selalu benar
selama minimal ada
satu yang benar

p	q	$p \vee q$
B	B	B
B	S	B
S	B	B
S	S	S

Disjungsi (or, \vee , $||$)

Misalkan kita diberikan proposisi “ p atau q ”.

“Pilih aku **atau** dia”



*Kalau bisa dua
kenapa harus
satu?
~ Algoritma Greedy*

Disjungsi Eksklusif (xor , \oplus , \wedge)

Terkadang di dalam kehidupan kita diminta untuk membuat pilihan dan tidak diperkenankan untuk memilih keduanya

Kala itu Seif sedang makan malam bersama calon mertuanya

“Kamu mau saya masak Gulai atau Kari Ayam?”



Disjungsi Eksklusif (xor , \oplus , \wedge)

Terkadang di dalam kehidupan kita diminta untuk membuat pilihan dan tidak diperkenankan untuk memilih keduanya

Seif hanya bisa dibuatkan makanan salah satu dari Gulai atau Kari



Disjungsi Eksklusif (xor, \oplus , \wedge)

Apa yang akan terjadi jika Seif meminta untuk dimasakkan keduanya?

Pernyataan $p \text{ xor } q$
akan selalu benar
selama $p \neq q$

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

Implikasi (Jika ... ,Maka ...

Jika saya lapar, **maka** saya akan makan.

p	q	$p \rightarrow q$
B	B	B
B	S	S
S	B	B
S	S	B

Jika 3 itu bilangan ganjil, **maka** 2 itu bilangan genap.

Jika gatau lagi, **maka** 5 itu bilangan prima

Biimplikasi (... jika dan hanya jika ..)

Suatu bilangan dikatakan prima **jika dan hanya jika** bilangan itu habis dibagi dirinya sendiri dan 1

$P \leftrightarrow Q$ akan bernilai benar selama $P == Q$

p	q	$p \leftrightarrow q$
B	B	B
B	S	S
S	B	S
S	S	B

Tautologi

Tautologi adalah Ketika kita mempunyai beberapa proposisi missal P_1 dan P_2 . $P_1 \rightarrow P_2$ akan selalu bernilai benar. Dengan mengevaluasi secara implikatif dapat dinyatakan bahwa $P_1 \equiv P_2$ (Ekivalen)

TABLE 8**Precedence of
Logical Operators.**

<i>Operator</i>	<i>Precedence</i>
\neg	1
\wedge	2
\vee	3
\rightarrow	4
\leftrightarrow	5

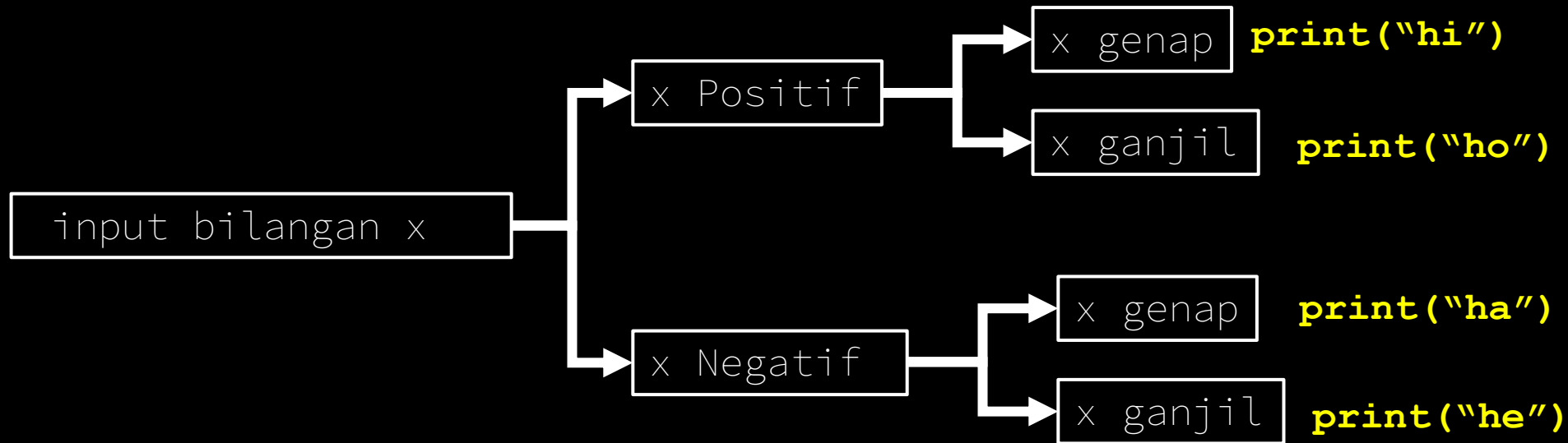
Clean Logical Nested If

Anda diminta untuk membuat program untuk memeriksa apakah x adalah bilangan positif atau negatif.

- Jika x bilangan positif , kemudian periksa apakah x
 - Bilangan genap, maka cetak “hi”
 - Bilangan ganjil, maka cetak “ho”
- Jika x bilangan negative maka periksa apakah x
 - Bilangan genap, maka cetak “ha”
 - Bilangan ganjil, maka cetak “he”

Clean Logical Nested If

Kita dapat menggambarkan flowchartnya



Not Clean

```
x = int(input())
if(x > 0):
    if(x%2 == 0):
        print("hi")
    else:
        print("ho")
elif(x < 0):
    if(not (x%2)):
        print("ha")
    elif((x%2)):
        print("he")
```

Mudah dibaca tapi proses program lebih banyak

Clean if

```
x = int(input())
if(x > 0 and not(x%2)):
    print("hi")
elif(x > 0 and (x%2)):
    print("ho")
elif(x < 0 and not(x%2)):
    print("ha")
else:
    print("he")
```

Clean Logical Nested If

Kita dapat menggambarkan flowchartnya

