

TUGAS INDIVIDU
FUZZY LOGIC
FUNGSI KEANGGOTAAN TRAPESIUM



DISUSUN OLEH :

Reksi Hendra Pratama (G1A022032)

DOSEN PENGAMPU :

Dr. Endina Putri Purwandari, S.T, M.Kom.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS BENGKULU
2024

1.1 Soal Individu

Suatu penelitian dilakukan untuk mencari jumlah produksi berdasarkan pengaruh faktor suhu, kebisingan, dan pencahayaan. Dalam penelitian ini ada 30 pekerja, yang masing-masing melakukan 27 kali percobaan dengan kombinasi suhu ($^{\circ}\text{C}$), kebisingan (dB), dan pencahayaan (lux) yang berbeda untuk menghasilkan sejumlah produk. Banyaknya data diperoleh sejumlah 810 data. Dari ketigapuluh data untuk setiap kombinasi diambil nilai rata-ratanya, sehingga data yang akan diolah tinggal 27 data sebagai berikut :

No	Suhu ($^{\circ}\text{C}$)	Kebisingan (dB)	Pencahayaan (lux)	Rata-rata jumlah produk	Standar deviasi
1	22	55	150	148,00	4,71
2	22	55	300	150,90	4,78
3	22	55	500	146,50	4,90
4	22	75	150	143,10	4,90
5	22	75	300	146,53	4,58
6	22	75	500	142,73	5,42
7	22	90	150	136,73	4,49
8	22	90	300	140,77	4,49
9	22	90	500	135,97	4,75
10	26	55	150	149,73	4,43
11	26	55	300	153,27	5,59
12	26	55	500	152,13	5,04
13	26	75	150	148,00	5,15
14	26	75	300	150,63	5,06
15	26	75	500	147,63	4,84
16	26	90	150	141,47	5,69
17	26	90	300	145,67	4,81
18	26	90	500	140,20	4,76
19	32	55	150	142,10	4,28
20	32	55	300	146,53	5,38
21	32	55	500	142,17	4,53
22	32	75	150	138,70	4,84
23	32	75	300	141,40	4,95
24	32	75	500	138,30	5,12
25	32	90	150	133,33	4,71
26	32	90	300	138,53	4,51
27	32	90	500	137,77	4,83

Tentukan :

- Fungsi Keanggotaan beserta gambarnya
- 27 aturan Fuzzy
- Derajat keanggotaan nilai tiap variable dalam setiap himpunan
- α -predikat untuk setiap aturan
- Rata-rata jumlah produk (gunakan metode defuzzy weighted average)

NPM saya genap, jadi saya menggunakan fungsi keanggotaan trapesium.

Tentukan

- Fungsi Keanggotaan beserta gambarnya

Jawab:

Pada soal nomor 1 ini saya membuat fungsi keanggotaan trapesium

```
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
import matplotlib.pyplot as plt

# Definisikan variabel fuzzy
suhu = ctrl.Antecedent(np.arange(22, 33, 1), 'suhu')
kebisingan = ctrl.Antecedent(np.arange(55, 91, 1), 'kebisingan')
pencahayaan = ctrl.Antecedent(np.arange(150, 501, 1), 'pencahayaan')
produksi = ctrl.Consequent(np.arange(120, 161, 1), 'produksi')

# Fungsi keanggotaan suhu
suhu['dingin'] = fuzz.trapmf(suhu.universe, [22, 22, 23, 25])
suhu['normal'] = fuzz.trapmf(suhu.universe, [23, 26, 29, 30])
suhu['panas'] = fuzz.trapmf(suhu.universe, [29, 32, 32, 32])
```

```

# Fungsi keanggotaan kebisingan
kebisingan['rendah'] = fuzz.trapmf(kebisingan.universe, [55, 55, 60, 65])
kebisingan['sedang'] = fuzz.trapmf(kebisingan.universe, [60, 70, 80, 90])
kebisingan['tinggi'] = fuzz.trapmf(kebisingan.universe, [85, 90, 90, 90])

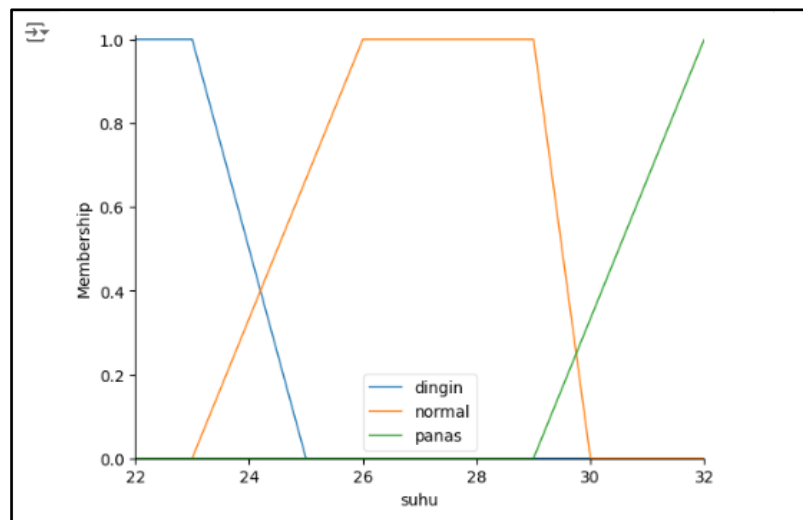
# Fungsi keanggotaan pencahayaan
pencahayaan['redup'] = fuzz.trapmf(pencahayaan.universe, [150, 150, 200, 250])
pencahayaan['normal'] = fuzz.trapmf(pencahayaan.universe, [200, 300, 400, 500])
pencahayaan['terang'] = fuzz.trapmf(pencahayaan.universe, [450, 500, 500, 500])

# Fungsi keanggotaan produksi
produksi['rendah'] = fuzz.trapmf(produksi.universe, [120, 130, 135, 140])
produksi['sedang'] = fuzz.trapmf(produksi.universe, [135, 145, 150, 155])
produksi['tinggi'] = fuzz.trapmf(produksi.universe, [150, 155, 160, 160])

# Gambar fungsi keanggotaan
suhu.view()
kebisingan.view()
pencahayaan.view()
produksi.view()
plt.show()

```

Gambar 2.1 Kode Fungsi keanggotaan



Gambar 2.2 Fungsi keanggotaan suhu

Source Code:

```

import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
import matplotlib.pyplot as plt

# Definisikan variabel fuzzy
suhu = ctrl.Antecedent(np.arange(22, 33, 1), 'suhu')
kebisingan = ctrl.Antecedent(np.arange(55, 91, 1), 'kebisingan')
pencahayaan = ctrl.Antecedent(np.arange(150, 501, 1), 'pencahayaan')

```

```

produksi = ctrl.Consequent(np.arange(120, 161, 1), 'produksi')

# Fungsi keanggotaan suhu
suhu['dingin'] = fuzz.trapmf(suhu.universe, [22, 22, 23, 25])
suhu['normal'] = fuzz.trapmf(suhu.universe, [23, 26, 29, 30])
suhu['panas'] = fuzz.trapmf(suhu.universe, [29, 32, 32, 32])

# Fungsi keanggotaan kebisingan
kebisingan['rendah'] = fuzz.trapmf(kebisingan.universe, [55, 55, 60, 65])
kebisingan['sedang'] = fuzz.trapmf(kebisingan.universe, [60, 70, 80, 90])
kebisingan['tinggi'] = fuzz.trapmf(kebisingan.universe, [85, 90, 90, 90])

# Fungsi keanggotaan pencahayaan
pencahayaan['redup'] = fuzz.trapmf(pencahayaan.universe, [150, 150, 200, 250])
pencahayaan['normal'] = fuzz.trapmf(pencahayaan.universe, [200, 300, 400, 500])
pencahayaan['terang'] = fuzz.trapmf(pencahayaan.universe, [450, 500, 500, 500])

# Fungsi keanggotaan produksi
produksi['rendah'] = fuzz.trapmf(produksi.universe, [120, 130, 135, 140])
produksi['sedang'] = fuzz.trapmf(produksi.universe, [135, 145, 150, 155])
produksi['tinggi'] = fuzz.trapmf(produksi.universe, [150, 155, 160, 160])

# Gambar fungsi keanggotaan
suhu.view()
kebisingan.view()
pencahayaan.view()
produksi.view()
plt.show()

```

Penjelasan:

- 1) Fungsi keanggotaan dingin (garis biru)

$$\mu_{\text{dingin}}(x) = \begin{cases} 1, & \text{jika } x \leq 23 \\ \frac{25-x}{25-23}, & \text{jika } 23 < x \leq 25 \\ 0, & \text{jika } x > 25 \end{cases}$$

Fungsi keanggotaan ini berbentuk trapesium, dengan keanggotaan penuh (nilai 1) pada suhu dari 22°C hingga 23°C. Artinya, suhu pada rentang tersebut dianggap sepenuhnya "dingin".

Setelah 23°C, fungsi mulai turun, dan pada suhu di atas 25°C, nilai keanggotaan mencapai 0, artinya suhu di atas 25°C tidak dianggap dingin sama sekali.

2) Fungsi keanggotaan normal (garis oranye)

$$\mu_{\text{normal}}(x) = \begin{cases} 0, & \text{jika } x \leq 23 \text{ atau } x \geq 30 \\ \frac{x-23}{26-23}, & \text{jika } 23 < x \leq 26 \\ 1, & \text{jika } 26 < x \leq 29 \\ \frac{30-x}{30-29}, & \text{jika } 29 < x \leq 30 \end{cases}$$

Fungsi ini juga berbentuk trapesium, dengan keanggotaan penuh (nilai 1) untuk suhu antara 26°C hingga 29°C, yang berarti suhu dalam rentang tersebut sepenuhnya normal.

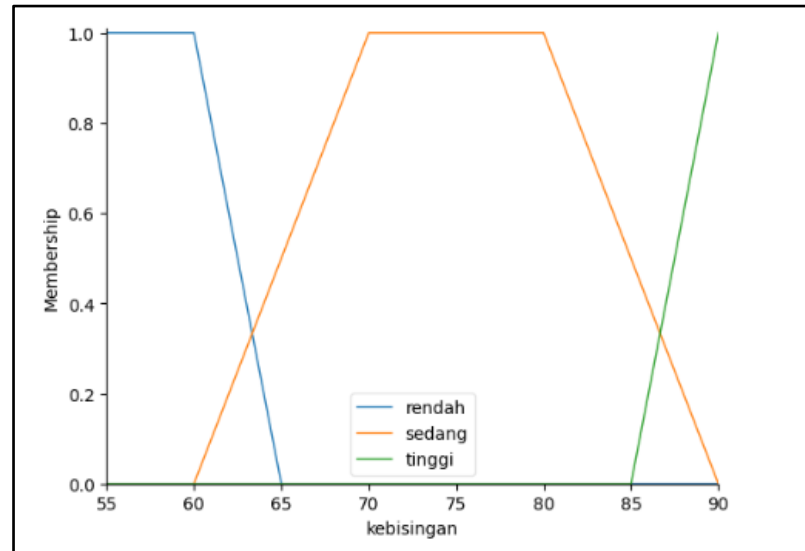
Fungsi ini memiliki transisi dari suhu 23°C hingga 26°C, di mana nilai keanggotaan naik dari 0 menjadi 1, dan setelah 29°C hingga 30°C, nilai keanggotaan turun lagi hingga 0. Suhu di luar rentang 23°C-30°C tidak dianggap normal.

3) Fungsi keanggotaan panas (garis hijau)

$$\mu_{\text{panas}}(x) = \begin{cases} 0, & \text{jika } x \leq 29 \\ \frac{x-29}{32-29}, & \text{jika } 29 < x \leq 32 \\ 1, & \text{jika } x > 32 \end{cases}$$

Fungsi keanggotaan untuk suhu panas dimulai pada suhu 29°C, dan mencapai nilai keanggotaan penuh (1) pada suhu 32°C. Ini berarti bahwa suhu 32°C atau lebih dianggap sepenuhnya "panas".

Sebelum suhu 29°C, nilai keanggotaan adalah 0, menunjukkan bahwa suhu di bawah 29°C tidak dianggap panas.



Gambar 2.3 Fungsi keanggotaan kebisingan

Penjelasan :

1) Fungsi keanggotaan rendah (garis biru)

$$\mu_{\text{rendah}}(x) = \begin{cases} 1, & \text{jika } x \leq 60 \\ \frac{65-x}{65-60}, & \text{jika } 60 < x \leq 65 \\ 0, & \text{jika } x > 65 \end{cases}$$

Fungsi keanggotaan rendah terjadi pada rentang kebisingan antara 55 dB hingga 60 dB, di mana kebisingan di bawah atau sama dengan 60 dB sepenuhnya dianggap "rendah". Dari 60 dB hingga 65 dB, ada transisi di mana nilai keanggotaan turun dari 1 menjadi 0, sehingga kebisingan di atas 65 dB tidak dianggap rendah.

2) Fungsi keanggotaan sedang (garis oranye)

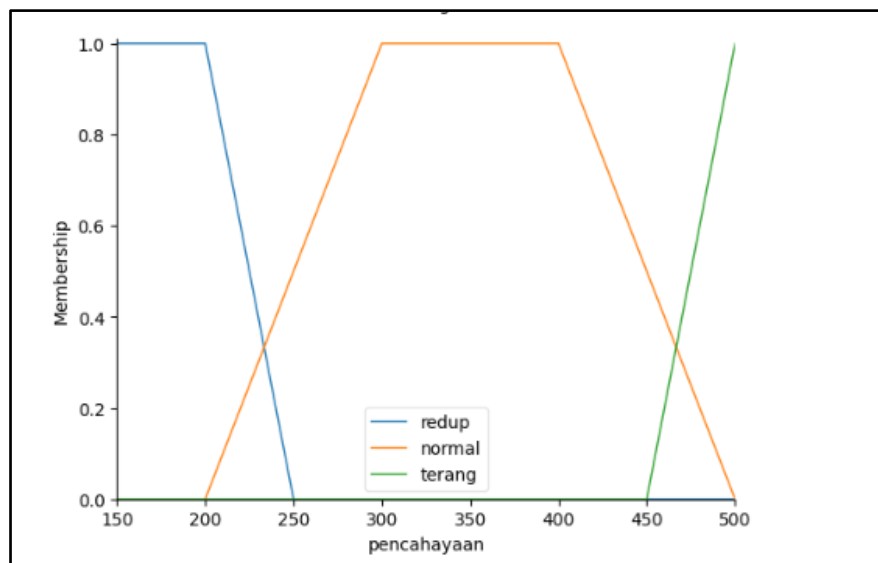
$$\mu_{\text{sedang}}(x) = \begin{cases} 0, & \text{jika } x \leq 60 \text{ atau } x \geq 85 \\ \frac{x-60}{65-60}, & \text{jika } 60 < x \leq 65 \\ 1, & \text{jika } 65 < x \leq 80 \\ \frac{85-x}{85-80}, & \text{jika } 80 < x \leq 85 \end{cases}$$

Fungsi keanggotaan sedang terjadi pada kebisingan antara 65 dB hingga 80 dB, keanggotaan dimulai dari 60 dB hingga 65 dB (naik dari 0 menjadi 1) dan dari 80 dB hingga 85 dB (turun dari 1 menjadi 0). Kebisingan di luar rentang 65 dB hingga 85 dB tidak dianggap masuk dalam kategori sedang. Nilai 1 terjadi ketika nilai kebisingan terletak antara 65 hingga 80.

3) Fungsi keanggotaan tinggi (garis hijau)

$$\mu_{\text{tinggi}}(x) = \begin{cases} 0, & x < 85 \\ \frac{x-85}{90-85}, & 85 \leq x \leq 90 \\ 1, & x > 90 \end{cases}$$

Fungsi keanggotaan tinggi terjadi pada kebisingan antara 85 dB hingga 90 dB. Sebelum 85 dB, nilai keanggotaan adalah 0. Kebisingan di atas 90 dB sepenuhnya dianggap tinggi dengan nilai 1. sedangkan di antara 85 hingga 90 nilainya berubah dari 0 perlahan meningkat ke 1.



Gambar 2.4 Fungsi keanggotaan pencahayaan

Penjelasan :

1) Fungsi keanggotaan redup (garis biru)

$$\mu_{\text{redup}}(x) = \begin{cases} 1, & 150 \leq x \leq 200 \\ \frac{250-x}{250-200}, & 200 < x \leq 250 \\ 0, & x > 250 \end{cases}$$

Fungsi keanggotaan pencahayaan dianggap sepenuhnya redup dengan nilai keanggotaan untuk rentang antara 150 hingga 200 lux. Kemudian antara 200 hingga 250 lux, nilai keanggotaan redup menurun secara linear dari 1 hingga 0. Setelah 250 lux, pencahayaan tidak lagi dianggap redup.

2) Fungsi keanggotaan normal (garis orange)

$$\mu_{\text{normal}}(x) = \begin{cases} 0, & x < 200 \text{ atau } x > 500 \\ \frac{x-200}{300-200}, & 200 \leq x \leq 300 \\ 1, & 300 < x < 400 \\ \frac{500-x}{500-400}, & 400 \leq x \leq 500 \end{cases}$$

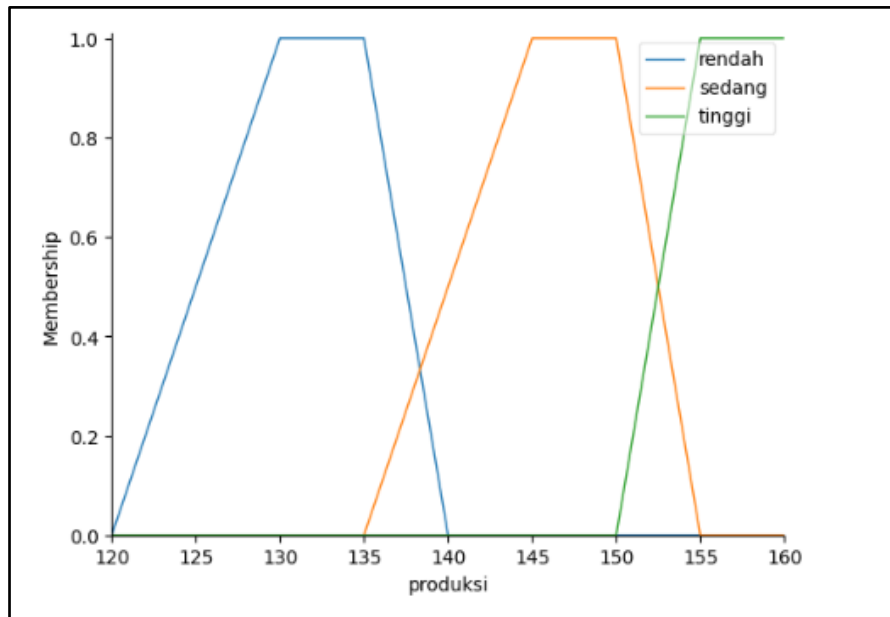
Fungsi keanggotaan Pencahayaan dianggap sepenuhnya normal dengan nilai keanggotaan = 1 untuk rentang antara 300 hingga 400 lux. Dari 200 hingga 300 lux, nilai keanggotaan normal meningkat secara linear dari 0 hingga 1. Dari 400 hingga 500 lux, nilai keanggotaan normal menurun secara linear dari 1 hingga 0. Dan setelah fi luar rentang 200 hingga 500 lux, pencahayaan tidak dianggap normal.

3) Fungsi keanggotaan terang (garis hijau)

$$\mu_{\text{terang}}(x) = \begin{cases} 0, & x < 450 \\ \frac{x-450}{500-450}, & 450 \leq x \leq 500 \\ 1, & x > 500 \end{cases}$$

Fungsi keanggotaan Pencahayaan dianggap sepenuhnya terang dengan nilai keanggotaan 1 untuk pencahayaan lebih dari 500 lux. Dari 450 hingga 500 lux, nilai keanggotaan normal meningkat secara linear dari 0 hingga 1. Dan setelah nilai pencahayaan kurang dari 450 maka, pencahayaan tidak dianggap terang.

$$\mu_{\text{rendah}}(x) = \begin{cases} 0, & x \leq 120 \\ \frac{x-120}{130-120}, & 120 < x \leq 130 \\ 1, & 130 < x \leq 135 \\ \frac{140-x}{140-135}, & 135 < x \leq 140 \\ 0, & x > 140 \end{cases}$$



Gambar 2.5 Fungsi keanggotaan produksi

Penjelasan :

1) Fungsi keanggotaan rendah (garis biru)

$$\mu_{\text{rendah}}(x) = \begin{cases} 0, & x \leq 120 \\ \frac{x-120}{130-120}, & 120 < x \leq 130 \\ 1, & 130 < x \leq 135 \\ \frac{140-x}{140-135}, & 135 < x \leq 140 \\ 0, & x > 140 \end{cases}$$

Fungsi keanggotaan produksi dimulai dari rentang 120 hingga 140. Untuk kurang dari atau sama dengan 120 nilai keanggotaannya adalah 0. Kemudian antara 120 hingga 130 lux, nilai keanggotaan rendah meningkat secara linear dari 0 hingga 1. Setelah 130 sampai 135 lux maka dianggap derajat keanggotaannya adalah 1. Kemudian dimulai dari 135 hingga 140 nilai keanggotaan menurun dari 1 hingga 0 dan puncaknya setelah lebih dari 140, maka nilai keanggotaan dianggap 0.

2) Fungsi keanggotaan sedang (garis orange)

$$\mu_{\text{sedang}}(x) = \begin{cases} 0, & x \leq 135 \\ \frac{x-135}{145-135}, & 135 < x \leq 145 \\ 1, & 145 < x \leq 150 \\ \frac{155-x}{155-150}, & 150 < x \leq 155 \\ 0, & x > 155 \end{cases}$$

Fungsi keanggotaan produksi sedang dimulai dari rentang 135 hingga 155. Untuk kurang dari atau sama dengan 135 nilai keanggotaannya adalah 0. Kemudian antara 135 hingga 145, nilai keanggotaan rendah meningkat secara linear dari 0 hingga 1. Setelah 145 hingga 150 maka dianggap derajat keanggotaannya adalah 1. Kemudian dimulai dari 150 hingga 155 nilai keanggotaan menurun dari 1 hingga 0 dan puncaknya setelah lebih dari 155, maka nilai keanggotaan dianggap 0.

3) Fungsi keanggotaan tinggi (garis hijau)

$$\mu_{\text{tinggi}}(x) = \begin{cases} 0, & x \leq 150 \\ \frac{x-150}{155-150}, & 150 < x \leq 155 \\ 1, & x \geq 155 \end{cases}$$

Fungsi keanggotaan produksi sedang dimulai dari rentang 150 hingga 160. Untuk kurang dari atau sama dengan 150 nilai keanggotaannya adalah 0. Kemudian antara 150 hingga 155, nilai keanggotaan rendah meningkat secara linear dari 0 hingga 1. Setelah itu puncaknya setelah lebih dari 155, maka nilai keanggotaan dianggap 1.

b. 27 Aturan Fuzzy

```
# Kombinasi untuk suhu 'dingin'
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['normal'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['terang'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['normal'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['terang'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['terang'], produksi['sedang']))

# Kombinasi untuk suhu 'normal'
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['redup'], produksi['rendah']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['normal'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['terang'], produksi['tinggi']))

# Kombinasi untuk suhu 'panas'
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['normal'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['normal'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['terang'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['redup'], produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['normal'], produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['terang'], produksi['tinggi']))
```

Gambar 2.6 Kode 27 Aturan Fuzzy

Source Code :

Definisikan aturan fuzzy

```
rules = []
```

Kombinasi untuk suhu 'dingin'

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['redup'],
produksi['rendah'])))
```

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['normal'],
produksi['rendah'])))
```

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['terang'],
produksi['sedang']))
```

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['redup'],
produksi['rendah'])))
```

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['normal'],
produksi['rendah'])))
```

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['terang'],
produksi['sedang']))
```

```
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['redup'],
```

```

produksi['rendah']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['normal'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['terang'],
produksi['sedang']))
# Kombinasi untuk suhu 'normal'
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['redup'],
produksi['rendah']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['normal'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['terang'],
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['redup'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['normal'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['terang'],
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['redup'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['normal'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['terang'],
produksi['tinggi']))
# Kombinasi untuk suhu 'panas'
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['redup'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['normal'],
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['terang'],
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['redup'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['normal'],

```

```
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['terang'],
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['redup'],
produksi['sedang']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['normal'],
produksi['tinggi']))
rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['terang'],
produksi['tinggi']))
```

Gambar 2.7 Output Aturan Fuzzy part 1

Gambar 2.8 Output Aturan Fuzzy part 2

- 1) Jika suhu dingin, kebisingan rendah, dan pencahayaan redup, maka produksi rendah
- 2) Jika suhu dingin, kebisingan rendah, dan pencahayaan normal, maka produksi rendah.
- 3) Jika suhu dingin, kebisingan rendah, dan pencahayaan terang, maka produksi sedang.
- 4) Jika suhu dingin, kebisingan sedang, dan pencahayaan redup, maka produksi rendah.
- 5) Jika suhu dingin, kebisingan sedang, dan pencahayaan normal, maka produksi rendah.
- 6) Jika suhu dingin, kebisingan sedang, dan pencahayaan terang, maka produksi sedang.
- 7) Jika suhu dingin, kebisingan tinggi, dan pencahayaan redup, maka produksi rendah.
- 8) Jika suhu dingin, kebisingan tinggi, dan pencahayaan normal, maka produksi sedang.
- 9) Jika suhu dingin, kebisingan tinggi, dan pencahayaan terang, maka produksi sedang.
- 10) Jika suhu normal, kebisingan rendah, dan pencahayaan redup, maka produksi rendah.
- 11) Jika suhu normal, kebisingan rendah, dan pencahayaan normal, maka produksi sedang.
- 12) Jika suhu normal, kebisingan rendah, dan pencahayaan terang, maka produksi tinggi.
- 13) Jika suhu normal, kebisingan sedang, dan pencahayaan redup, maka produksi sedang.
- 14) Jika suhu normal, kebisingan sedang, dan pencahayaan normal, maka produksi sedang.
- 15) Jika suhu normal, kebisingan sedang, dan pencahayaan terang, maka produksi tinggi.
- 16) Jika suhu normal, kebisingan tinggi, dan pencahayaan redup, maka produksi sedang.
- 17) Jika suhu normal, kebisingan tinggi, dan pencahayaan normal, maka produksi sedang.
- 18) Jika suhu normal, kebisingan tinggi, dan pencahayaan terang, maka produksi tinggi.
- 19) Jika suhu panas, kebisingan rendah, dan pencahayaan redup, maka produksi sedang.
- 20) Jika suhu panas, kebisingan rendah, dan pencahayaan normal, maka produksi tinggi.
- 21) Jika suhu panas, kebisingan rendah, dan pencahayaan terang, maka produksi tinggi.
- 22) Jika suhu panas, kebisingan sedang, dan pencahayaan redup, maka produksi sedang.
- 23) Jika suhu panas, kebisingan sedang, dan pencahayaan normal, maka produksi tinggi.
- 24) Jika suhu panas, kebisingan sedang, dan pencahayaan terang, maka produksi tinggi.
- 25) Jika suhu panas, kebisingan tinggi, dan pencahayaan redup, maka produksi sedang.
- 26) Jika suhu panas, kebisingan tinggi, dan pencahayaan normal, maka produksi tinggi.
- 27) Jika suhu panas, kebisingan tinggi, dan pencahayaan terang, maka produksi tinggi.

Kemudian ada juga yang namanya f_{min} dan f_{max} . f_{min} digunakan untuk operasi AND, yang berarti hasil dari operasi AND adalah nilai keanggotaan terendah dari aturan yang bersangkutan. Dengan kata lain, output keanggotaan ditentukan oleh nilai terkecil di antara semua kondisi input (suhu, kebisingan, pencahayaan).

Fmax digunakan untuk operasi OR, yang berarti hasil dari operasi OR adalah nilai keanggotaan tertinggi dari aturan yang bersangkutan. Artinya, jika ada dua atau lebih kondisi yang menghasilkan nilai keanggotaan, output akan mengikuti nilai terbesar di antara input yang memenuhi syarat.

c. Derajat keanggotaan nilai dalam setiap himpunan variabel

Printscreen code:

```
# Contoh nilai input
nilai_suhu = 24 # Contoh suhu
nilai_kebisingan = 90 # Contoh kebisingan
nilai_pencahayaannya = 250 # Contoh pencahayaan

# Hitung derajat keanggotaan untuk setiap himpunan
derajat_keanggotaan_suhu_dingin = fuzz.interp_membership(suhu.universe, suhu['dingin'].mf, nilai_suhu)
derajat_keanggotaan_suhu_normal = fuzz.interp_membership(suhu.universe, suhu['normal'].mf, nilai_suhu)
derajat_keanggotaan_suhu_panas = fuzz.interp_membership(suhu.universe, suhu['panas'].mf, nilai_suhu)

derajat_keanggotaan_kebisingan_rendah = fuzz.interp_membership(kebisingan.universe, kebisingan['rendah'].mf, nilai_kebisingan)
derajat_keanggotaan_kebisingan_sedang = fuzz.interp_membership(kebisingan.universe, kebisingan['sedang'].mf, nilai_kebisingan)
derajat_keanggotaan_kebisingan_tinggi = fuzz.interp_membership(kebisingan.universe, kebisingan['tinggi'].mf, nilai_kebisingan)

derajat_keanggotaan_pencahayaannya_redup = fuzz.interp_membership(pencahayaannya.universe, pencahayaannya['redup'].mf, nilai_pencahayaannya)
derajat_keanggotaan_pencahayaannya_normal = fuzz.interp_membership(pencahayaannya.universe, pencahayaannya['normal'].mf, nilai_pencahayaannya)
derajat_keanggotaan_pencahayaannya_terang = fuzz.interp_membership(pencahayaannya.universe, pencahayaannya['terang'].mf, nilai_pencahayaannya)

# Print hasil derajat keanggotaan
print("Derajat Keanggotaan:")
print(f"Suhu Dingin: {derajat_keanggotaan_suhu_dingin:.2f}")
print(f"Suhu Normal: {derajat_keanggotaan_suhu_normal:.2f}")
print(f"Suhu Panas: {derajat_keanggotaan_suhu_panas:.2f}")

print(f"Kebisingan Rendah: {derajat_keanggotaan_kebisingan_rendah:.2f}")
print(f"Kebisingan Sedang: {derajat_keanggotaan_kebisingan_sedang:.2f}")
print(f"Kebisingan Tinggi: {derajat_keanggotaan_kebisingan_tinggi:.2f}")

print(f"Pencahayaannya Redup: {derajat_keanggotaan_pencahayaannya_redup:.2f}")
print(f"Pencahayaannya Normal: {derajat_keanggotaan_pencahayaannya_normal:.2f}")
print(f"Pencahayaannya Terang: {derajat_keanggotaan_pencahayaannya_terang:.2f}")
```

Gambar 2.9 Kode untuk derajat keanggotaan

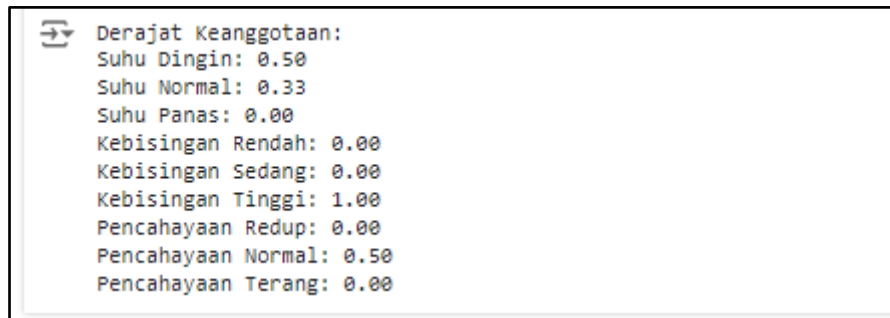
Penjelasan:

Pada bagian awal kode, saya memasukkan input yang akan dihitung, yaitu 24 untuk suhu, 90 untuk kebisingan dan 250 untuk pencahayaan. Kemudian Sebagai contoh, nilai suhu 24°C dapat memiliki derajat keanggotaan pada kategori dingin, normal, dan panas, dengan derajat keanggotaan yang berbeda-beda.

Untuk menghitung derajat keanggotaan ini, digunakan fungsi `interp_membership` dari library `skfuzzy`. Fungsi ini mengambil himpunan semesta (`universe`) dari variabel fuzzy, fungsi keanggotaan dari himpunan tertentu (misalnya, `dingin`, `normal`, atau `panas`), serta nilai input untuk menentukan derajat keanggotaannya. Hasilnya adalah nilai antara 0 dan 1 yang menunjukkan tingkat keanggotaan input dalam setiap kategori. Semakin mendekati 1, semakin besar keterkaitan nilai input dengan kategori tersebut. Proses ini diaplikasikan pada setiap

variabel fuzzy seperti suhu, kebisingan, dan pencahayaan untuk mendapatkan hasil yang akurat dalam pengambilan keputusan berbasis logika fuzzy.

Printscreen Hasil:



Gambar 2.10 Hasil derajat keanggotaan

Penjelasan Hasil:

Berdasarkan output hasil ternyata suhu 24 masuk ke suhu dingin 0,50 dan normal 0,33. Sedangkan kebisingan memiliki derajat keanggotaan 1 di kebisingan tinggi dan terakhir pencahayaan masuk ke pencahayaan normal dengan derajat 0,50.

d. α - predikat untuk setiap aturan

Printscreen code :



Gambar 2.11 Kode untuk α predikat


```
alpha_predikat_rule25 = np.min([derajat_keanggotaan_suhu_panas, derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaan_redup])
alpha_predikat_rule26 = np.min([derajat_keanggotaan_suhu_panas, derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaan_normal])
alpha_predikat_rule27 = np.min([derajat_keanggotaan_suhu_panas, derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaan_terang])

# Tampilkan alpha predikat
print(f"Alpha Predikat Rule 1: {alpha_predikat_rule1:.2f}")
print(f"Alpha Predikat Rule 2: {alpha_predikat_rule2:.2f}")
print(f"Alpha Predikat Rule 3: {alpha_predikat_rule3:.2f}")
print(f"Alpha Predikat Rule 4: {alpha_predikat_rule4:.2f}")
print(f"Alpha Predikat Rule 5: {alpha_predikat_rule5:.2f}")
print(f"Alpha Predikat Rule 6: {alpha_predikat_rule6:.2f}")
print(f"Alpha Predikat Rule 7: {alpha_predikat_rule7:.2f}")
print(f"Alpha Predikat Rule 8: {alpha_predikat_rule8:.2f}")
print(f"Alpha Predikat Rule 9: {alpha_predikat_rule9:.2f}")
print(f"Alpha Predikat Rule 10: {alpha_predikat_rule10:.2f}")
print(f"Alpha Predikat Rule 11: {alpha_predikat_rule11:.2f}")
print(f"Alpha Predikat Rule 12: {alpha_predikat_rule12:.2f}")
print(f"Alpha Predikat Rule 13: {alpha_predikat_rule13:.2f}")
print(f"Alpha Predikat Rule 14: {alpha_predikat_rule14:.2f}")
print(f"Alpha Predikat Rule 15: {alpha_predikat_rule15:.2f}")
print(f"Alpha Predikat Rule 16: {alpha_predikat_rule16:.2f}")
print(f"Alpha Predikat Rule 17: {alpha_predikat_rule17:.2f}")
print(f"Alpha Predikat Rule 18: {alpha_predikat_rule18:.2f}")
print(f"Alpha Predikat Rule 19: {alpha_predikat_rule19:.2f}")
print(f"Alpha Predikat Rule 20: {alpha_predikat_rule20:.2f}")
print(f"Alpha Predikat Rule 21: {alpha_predikat_rule21:.2f}")
print(f"Alpha Predikat Rule 22: {alpha_predikat_rule22:.2f}")
print(f"Alpha Predikat Rule 23: {alpha_predikat_rule23:.2f}")
print(f"Alpha Predikat Rule 24: {alpha_predikat_rule24:.2f}")
print(f"Alpha Predikat Rule 25: {alpha_predikat_rule25:.2f}")
print(f"Alpha Predikat Rule 26: {alpha_predikat_rule26:.2f}")
print(f"Alpha Predikat Rule 27: {alpha_predikat_rule27:.2f}")
```

Gambar 2.12 Kode untuk α predikat bagian 2

Source code:

```
# Alpha predikat adalah minimum dari derajat keanggotaan
alpha_predikat_rule1 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_rendah, derajat_keanggotaan_pencahayaan_redup])
alpha_predikat_rule2 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_rendah,
derajat_keanggotaan_pencahayaan_normal])
alpha_predikat_rule3 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_rendah, derajat_keanggotaan_pencahayaan_terang])

alpha_predikat_rule4 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_sedang, derajat_keanggotaan_pencahayaan_redup])
alpha_predikat_rule5 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_sedang,
derajat_keanggotaan_pencahayaan_normal])
alpha_predikat_rule6 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_sedang, derajat_keanggotaan_pencahayaan_terang])

alpha_predikat_rule7 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaan_redup])
```

```

alpha_predikat_rule8 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_normal])
alpha_predikat_rule9 = np.min([derajat_keanggotaan_suhu_dingin,
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_terang])

```

```

alpha_predikat_rule10 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_rendah, derajat_keanggotaan_pencahayaannya_redup])
alpha_predikat_rule11 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_rendah,
derajat_keanggotaan_pencahayaannya_normal])
alpha_predikat_rule12 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_rendah, derajat_keanggotaan_pencahayaannya_terang])

```

```

alpha_predikat_rule13 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_sedang, derajat_keanggotaan_pencahayaannya_redup])
alpha_predikat_rule14 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_sedang,
derajat_keanggotaan_pencahayaannya_normal])
alpha_predikat_rule15 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_sedang, derajat_keanggotaan_pencahayaannya_terang])

```

```

alpha_predikat_rule16 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_redup])
alpha_predikat_rule17 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_normal])
alpha_predikat_rule18 = np.min([derajat_keanggotaan_suhu_normal,
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_terang])

```

```

alpha_predikat_rule19 = np.min([derajat_keanggotaan_suhu_panas,
derajat_keanggotaan_kebisingan_rendah, derajat_keanggotaan_pencahayaannya_redup])
alpha_predikat_rule20 = np.min([derajat_keanggotaan_suhu_panas,
derajat_keanggotaan_kebisingan_rendah,
derajat_keanggotaan_pencahayaannya_normal])

```

```
alpha_predikat_rule21 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_rendah, derajat_keanggotaan_pencahayaannya_terang])
```

```
alpha_predikat_rule22 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_sedang, derajat_keanggotaan_pencahayaannya_redup])
```

```
alpha_predikat_rule23 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_sedang,  
derajat_keanggotaan_pencahayaannya_normal])
```

```
alpha_predikat_rule24 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_sedang, derajat_keanggotaan_pencahayaannya_terang])
```

```
alpha_predikat_rule25 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_redup])
```

```
alpha_predikat_rule26 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_normal])
```

```
alpha_predikat_rule27 = np.min([derajat_keanggotaan_suhu_panas,  
derajat_keanggotaan_kebisingan_tinggi, derajat_keanggotaan_pencahayaannya_terang])
```

```
# Tampilkan alpha predikat
```

```
print(f'Alpha Predikat Rule 1: {alpha_predikat_rule1:.2f}')
```

```
print(f'Alpha Predikat Rule 2: {alpha_predikat_rule2:.2f}')
```

```
print(f'Alpha Predikat Rule 3: {alpha_predikat_rule3:.2f}')
```

```
print(f'Alpha Predikat Rule 4: {alpha_predikat_rule4:.2f}')
```

```
print(f'Alpha Predikat Rule 5: {alpha_predikat_rule5:.2f}')
```

```
print(f'Alpha Predikat Rule 6: {alpha_predikat_rule6:.2f}')
```

```
print(f'Alpha Predikat Rule 7: {alpha_predikat_rule7:.2f}')
```

```
print(f'Alpha Predikat Rule 8: {alpha_predikat_rule8:.2f}')
```

```
print(f'Alpha Predikat Rule 9: {alpha_predikat_rule9:.2f}')
```

```
print(f'Alpha Predikat Rule 10: {alpha_predikat_rule10:.2f}')
```

```
print(f'Alpha Predikat Rule 11: {alpha_predikat_rule11:.2f}')
```

```
print(f'Alpha Predikat Rule 12: {alpha_predikat_rule12:.2f}')
```

```
print(f'Alpha Predikat Rule 13: {alpha_predikat_rule13:.2f}')
```

```
print(f'Alpha Predikat Rule 14: {alpha_predikat_rule14:.2f}')
```

```
print(f'Alpha Predikat Rule 15: {alpha_predikat_rule15:.2f}')
```

```

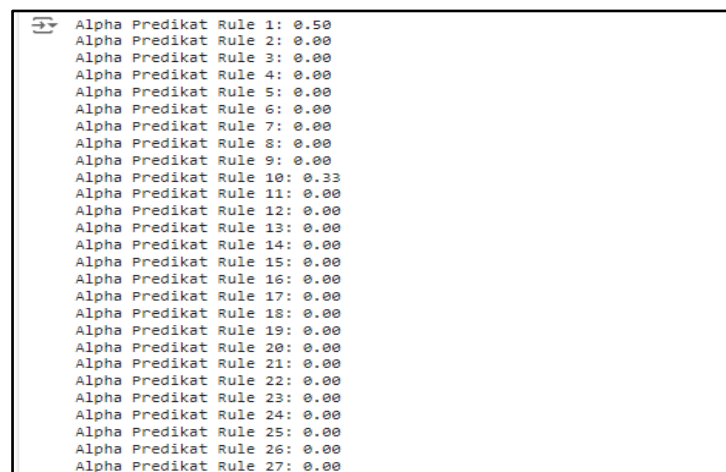
print(f'Alpha Predikat Rule 16: {alpha_predikat_rule16:.2f}')
print(f'Alpha Predikat Rule 17: {alpha_predikat_rule17:.2f}')
print(f'Alpha Predikat Rule 18: {alpha_predikat_rule18:.2f}')
print(f'Alpha Predikat Rule 19: {alpha_predikat_rule19:.2f}')
print(f'Alpha Predikat Rule 20: {alpha_predikat_rule20:.2f}')
print(f'Alpha Predikat Rule 21: {alpha_predikat_rule21:.2f}')
print(f'Alpha Predikat Rule 22: {alpha_predikat_rule22:.2f}')
print(f'Alpha Predikat Rule 23: {alpha_predikat_rule23:.2f}')
print(f'Alpha Predikat Rule 24: {alpha_predikat_rule24:.2f}')
print(f'Alpha Predikat Rule 25: {alpha_predikat_rule25:.2f}')
print(f'Alpha Predikat Rule 26: {alpha_predikat_rule26:.2f}')
print(f'Alpha Predikat Rule 27: {alpha_predikat_rule27:.2f}')

```

Penjelasan:

Kode ini saya buat untuk menghitung alpha predikat untuk 27 aturan dalam sistem logika fuzzy berdasarkan tiga variabel: suhu, kebisingan, dan pencahayaan. Alpha predikat dihitung sebagai nilai minimum dari derajat keanggotaan untuk setiap aturan yang mengkombinasikan kategori berbeda dari suhu (dingin, normal, panas), kebisingan (rendah, sedang, tinggi), dan pencahayaan (redup, normal, terang). Fungsi `np.min()` digunakan untuk mendapatkan nilai terendah dari derajat keanggotaan yang relevan pada setiap aturan, mencerminkan penerapan operator AND dalam logika fuzzy. Hasil alpha predikat dari masing-masing aturan kemudian ditampilkan menggunakan fungsi `print`. Alpha predikat ini digunakan dalam tahap inferensi untuk menentukan seberapa kuat suatu aturan berlaku berdasarkan nilai input.

Printscreen output:



```

Alpha Predikat Rule 1: 0.50
Alpha Predikat Rule 2: 0.00
Alpha Predikat Rule 3: 0.00
Alpha Predikat Rule 4: 0.00
Alpha Predikat Rule 5: 0.00
Alpha Predikat Rule 6: 0.00
Alpha Predikat Rule 7: 0.00
Alpha Predikat Rule 8: 0.00
Alpha Predikat Rule 9: 0.00
Alpha Predikat Rule 10: 0.33
Alpha Predikat Rule 11: 0.00
Alpha Predikat Rule 12: 0.00
Alpha Predikat Rule 13: 0.00
Alpha Predikat Rule 14: 0.00
Alpha Predikat Rule 15: 0.00
Alpha Predikat Rule 16: 0.00
Alpha Predikat Rule 17: 0.00
Alpha Predikat Rule 18: 0.00
Alpha Predikat Rule 19: 0.00
Alpha Predikat Rule 20: 0.00
Alpha Predikat Rule 21: 0.00
Alpha Predikat Rule 22: 0.00
Alpha Predikat Rule 23: 0.00
Alpha Predikat Rule 24: 0.00
Alpha Predikat Rule 25: 0.00
Alpha Predikat Rule 26: 0.00
Alpha Predikat Rule 27: 0.00

```

Gambar 2.13 Output Alpha Predikat

- e. Rata-rata jumlah produk (gunakan metode defuzzy weighted average)

Printscreen code :

```
# Sistem kontrol fuzzy
produksi_ctrl = ctrl.ControlSystem(rules)
produksi_simulasi = ctrl.ControlSystemSimulation(produksi_ctrl)

# Set input untuk simulasi
produksi_simulasi.input['suhu'] = 30 # Suhu
produksi_simulasi.input['kebisingan'] = 85 # Kebisingan
produksi_simulasi.input['pencahayaan'] = 300 # Pencahayaan

# Jalankan simulasi
produksi_simulasi.compute()

# Periksa apakah output tersedia
if 'produksi' in produksi_simulasi.output:
    prediksi_produk = produksi_simulasi.output['produksi']
    print(f"Prediksi produksi: {prediksi_produk:.2f}")
else:
    print("Hasil tidak ditemukan, periksa sistem kontrol fuzzy Anda.")

# Lakukan defuzzifikasi (metode centroid/weighted average)
defuzz = fuzz.defuzz(np.arange(120, 161, 1), produksi['tinggi'].mf, 'centroid')
print(f"Defuzzifikasi (Weighted Average): {defuzz:.2f}")
```

Gambar 2.14 Source code produksi

Penjelasan:

Kode di atas merupakan implementasi dari sistem kontrol fuzzy untuk memprediksi output produksi berdasarkan input suhu, kebisingan, dan pencahayaan. Pertama, sebuah sistem kontrol fuzzy (produksi_ctrl) dibuat dari sekumpulan aturan fuzzy (rules). Sistem simulasi kontrol fuzzy (produksi_simulasi) kemudian dibuat berdasarkan sistem kontrol ini.

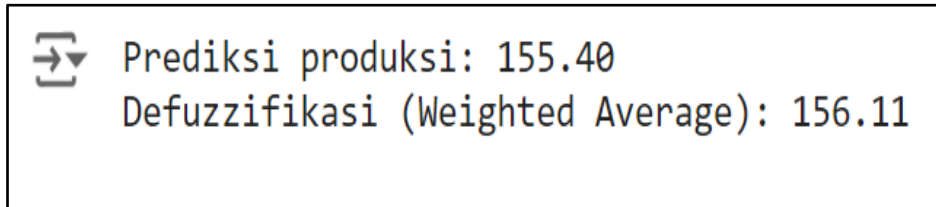
Setelah itu, input untuk simulasi diatur dengan memberikan nilai spesifik: suhu 30°C, kebisingan 85 dB, dan pencahayaan 300 lux. Simulasi kemudian dijalankan yang memproses input berdasarkan aturan fuzzy yang ada dan menghasilkan prediksi output berupa nilai produksi.

Jika sistem kontrol berhasil memprediksi output, nilai produksi akan ditampilkan. Jika tidak ada output yang terdeteksi, akan muncul pesan kesalahan yang meminta untuk memeriksa sistem kontrol fuzzy.

Selanjutnya, proses defuzzifikasi untuk rata rata produk dilakukan menggunakan metode centroid (atau rata-rata berbobot) untuk mengonversi hasil fuzzy menjadi output tunggal. Himpunan semesta untuk produksi ditentukan dalam rentang

120 hingga 160 (misalnya, mewakili produksi unit atau jumlah barang), dan fungsi keanggotaan dari kategori "tinggi" digunakan dalam proses ini. Hasil akhir dari defuzzifikasi juga ditampilkan untuk memberikan nilai output yang pasti.

Printscreen Hasil:



Gambar 2.15 Hasil Defuzzifikasi