

# **Laporan Tugas Pemrograman Fuzzy Logic**



Dosen Pembimbing:  
**IZZATUL UMMAH, S.T., M.T.**  
**CH2M3-IF-44-11**

Disusun oleh:  
**1301200240 Ramadhan Aditya Ibrahim**  
**1301204125 Ryan Chandra Hadi**

**S1 Informatika**  
**Universitas Telkom**  
**Bandung, Jawa Barat 2022**

## **Kata Pengantar**

Dengan mengucapkan puji dan rasa syukur kepada Allah SWT yang telah memberikan rahmat sehingga kita dapat menyelesaikan tugas dari mata kuliah Pengantar Kecerdasan Buatan dengan tema “Logika Fuzzy” dengan benar dan tepat waktu.

Untuk memenuhi nilai tugas pada mata kuliah Pengantar Kecerdasan Buatan, maka dibuatkan tugas yang dapat kita selesaikan. Tidak hanya itu, tujuan dari pembuatan laporan dan pengerjaan tugas ini adalah untuk menambah wawasan tentang pembahasan Logika Fuzzy bagi kita semua.

Kami mengucapkan terima kasih kepada semua pihak dari mulai ibu Izzatul Ummah selaku dosen pembimbing yang telah memberikan kita tugas besar untuk membuat Algoritma Genetika.

Kami sangat menyadari laporan yang kami susun masih jauh dari kata sempurna. Tetapi kita akan terus berusaha untuk selalu menjadi lebih baik untuk kedepannya.

Bandung, 14 Mei 2022

Kelompok 6

# **BAB I**

## **PENDAHULUAN**

### **1. PERSOALAN**

#### **1.1 DESKRIPSI TUGAS**

Diberikan file bengkel.xlsx berupa himpunan data 100 bengkel mobil yang ada di kota Bandung dengan dua atribut: Kualitas Servis (bilangan real 1-100; semakin tinggi semakin baik) dan Harga (bilangan real 1-10, semakin tinggi semakin mahal). Bangunlah sebuah sistem berbasis Fuzzy Logic untuk memilih 10 bengkel terbaik di kota Bandung. Sistem membaca masukan file bengkel.xlsx dan mengeluarkan output berupa sebuah file peringkat.xlsx yang berisi 10 nomor/ID bengkel terbaik beserta skor-nya (output Defuzzification).

Poin-poin yang harus Anda desain dan analisis:

- Jumlah dan Nama Linguistik setiap atribut input
- Bentuk dan Batas Fungsi Keanggotaan Input
- Aturan Inferensi
- Metode Defuzzification
- Bentuk dan Batas Fungsi Keanggotaan Output (sesuai metode Defuzzification)

Catatan: Poin-poin di atas harus ada di dalam Laporan Tugas!

Proses yang harus Anda implementasikan ke dalam program (bisa berbentuk fungsi/prosedur):

- Membaca data dari file
- Fuzzification
- Inferensi
- Defuzzification
- Menyimpan output ke file

Catatan: Proses-proses di atas harus dibangun tanpa menggunakan Library!

#### **1.2 OUTPUT PROGRAM**

Dengan masalah yang dideskripsikan di atas, output program Anda adalah sebuah file peringkat.xlsx yang berisi 10 nomor/ID bengkel terbaik beserta skor-nya (output Defuzzification).

## BAB II

### PEMBAHASAN

#### 2.1 Jumlah dan Nama Linguistik setiap atribut input

Pada pengerjaan soal ini kita menggunakan 2 variabel yaitu servis dan harga yang dimana masing-masing mempunyai 3 nama linguistik:

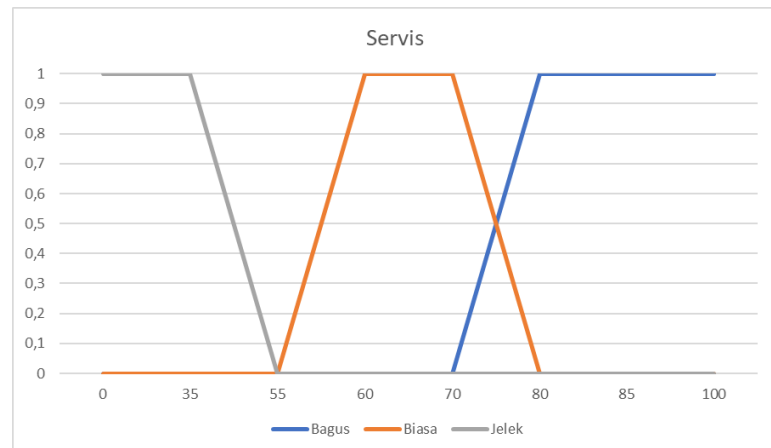
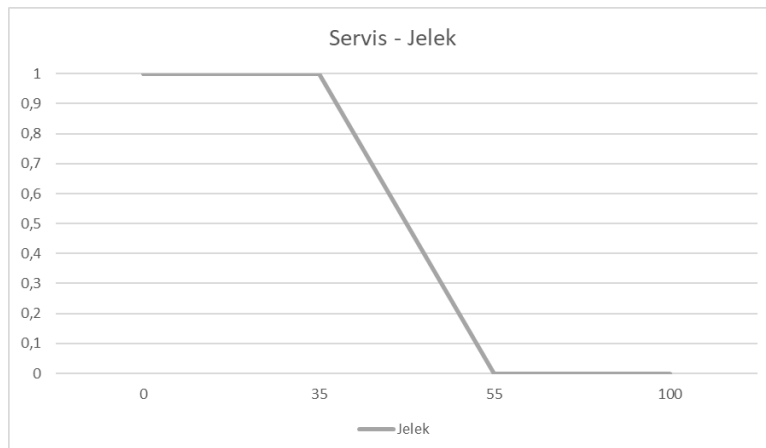
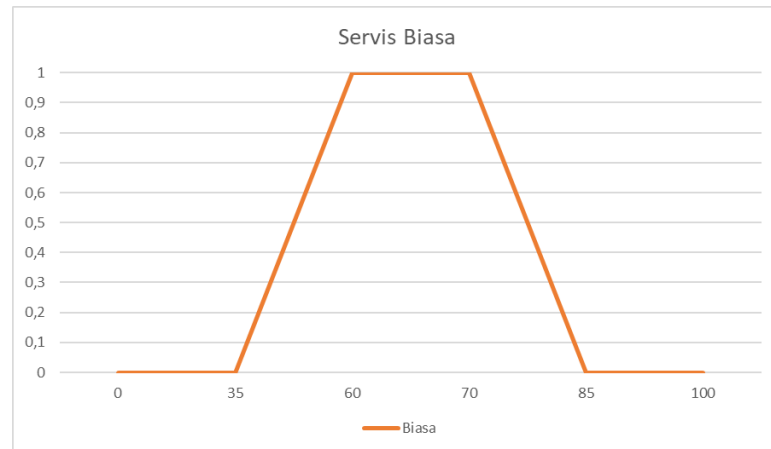
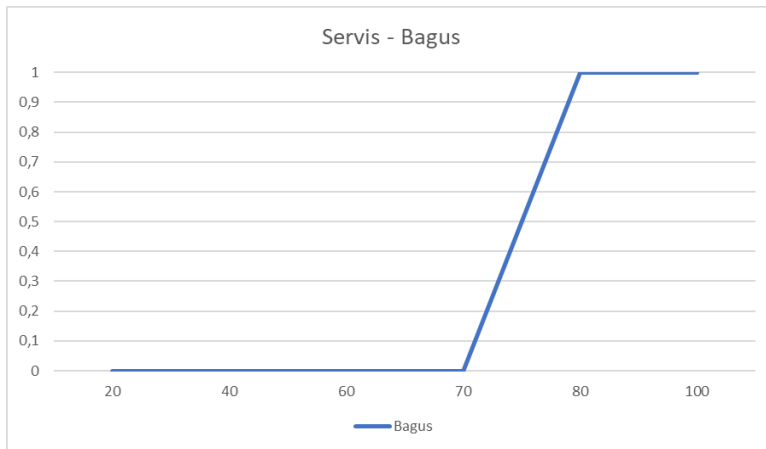
- Servis : bagus, biasa, dan jelek
- Harga : mahal, biasa, dan murah

Untuk Outputnya kita juga mendesain 3 linguistik yaitu:

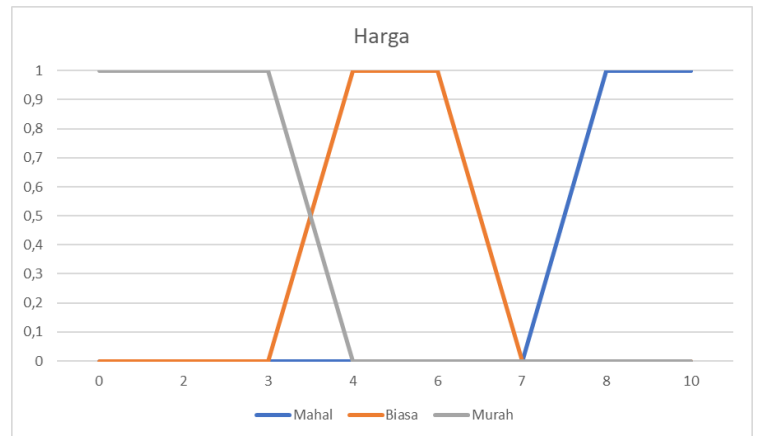
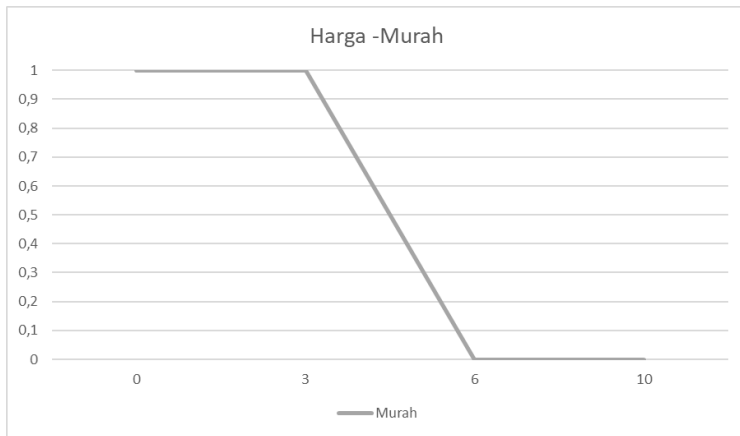
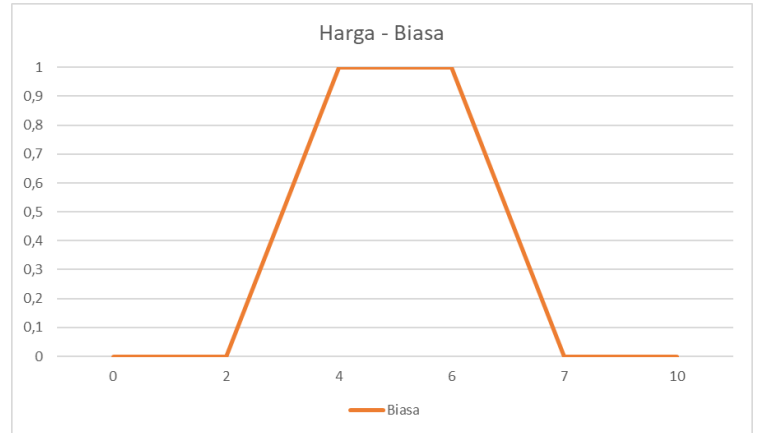
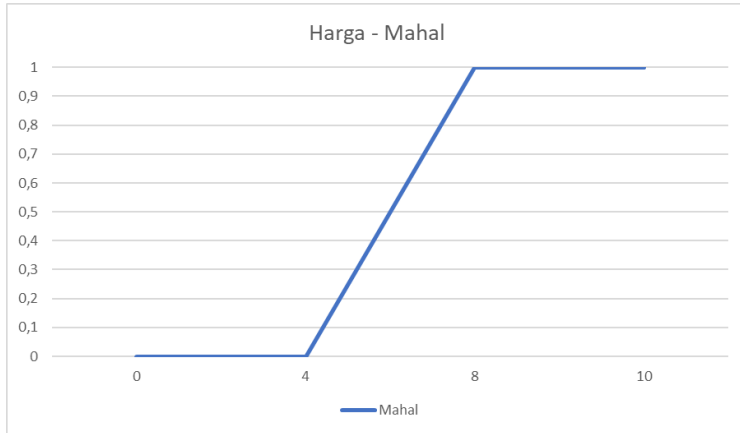
- Excellent
- Normal
- Bad

#### 2.2 Bentuk dan Batas Fungsi Keanggotaan Input

- Servis



• Harga



### 2.3 Aturan Inferensi

Service	Harga	Score
Bagus	Mahal	Normal
Bagus	Biasa	Excellent
Bagus	Murah	Excellent
Biasa	Mahal	Normal
Biasa	Biasa	Normal
Biasa	Murah	Excellent
Jelek	Mahal	Bad
Jelek	Biasa	Bad
Jelek	Murah	Normal

### 2.4 Metode Defuzzification

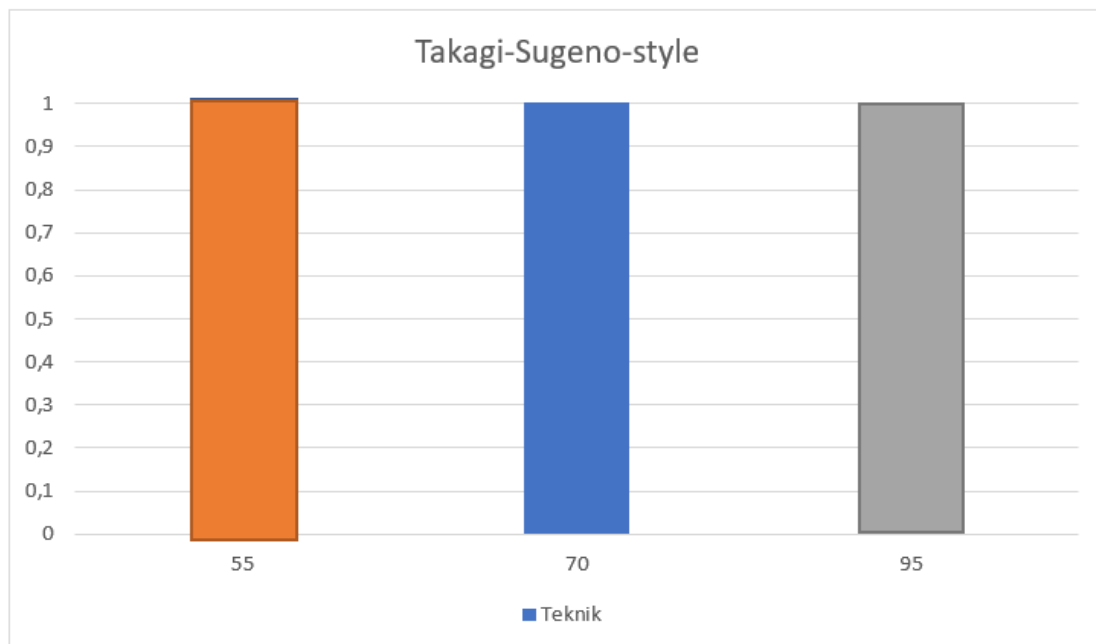
Pada metode defuzifikasi ini kita menggunakan metode Takagi Sugeno Style dengan menggunakan rumus sebagai berikut :

$$z^* = \frac{\sum_{i=1}^l \mu B_i \cdot c_i}{\sum_{i=1}^l \mu B_i}$$

$c_i = \text{constant for } i^{th} \text{ linguistic}$   
 $\mu B_i = \text{membership for } i^{th} \text{ linguistic}$

Dengan menggunakan teknik Takagi Sugeno Style kita harus memilih nilai yang mempresentasikan setiap keluaran dari linguistik.

## 2.5 Bentuk dan Batas Fungsi Keanggotaan Output (sesuai metode Defuzzification)



## BAB III

# IMPLEMENTASI PROGRAM

### 3.1 Library

```
import pandas as pd
```

**3.1.2 Pandas** pada program ini berfungsi merubah data array menjadi data berbentuk tabel.

### 3.2 Inisiasi

```
dataset = pd.read_excel('bengkel.xlsx')  
dataset
```

```
#define data  
ID = dataset["id"]  
servis = dataset["servis"]  
harga = dataset["harga"]
```

Pada bagian ini kita membuat sebuah variabel dataset yang berfungsi untuk membaca file bengkel dan memisahkan setiap kolomnya (data id, data servis, data harga)



### 3.3 fungsi fService

```
#design membership for servis

def fServis(x):
    #bagus
    if x > 80:
        bagus = 1
    elif x <= 70:
        bagus = 0
    else:
        bagus = (x-70)/(80-70)

    #jelek
    if x < 35:
        jelek = 1
    elif x >= 55:
        jelek = 0
    else:
        jelek = (55-x)/(55-35)

    #biasa
    if x <= 35 and x > 85:
        biasa = 0
    elif 35 < x <= 60:
        biasa = (x-35)/(60-35)
    elif 60 < x <= 70:
        biasa = 1
    elif 70 < x <= 85:
        biasa = (85-x)/(85-70)
    else:
        biasa = 0

    result = [bagus, biasa, jelek]
    return result
```

Function ini berfungsi untuk membuat desain membership untuk servis disini kita membuat 3 linguistik yaitu bagus, biasa, dan jelek.

### 3.4 fungsi fharga

```
#design membership for harga

def fHarga(x):
    #mahal
    if x >= 8:
        mahal = 1
    elif x < 4:
        mahal = 0
    elif 4 < x <= 8:
        mahal = (x-4)/(8-4)
    else:
        mahal = 0

    #murah
    if x > 3:
        murah = 0
    elif x <= 6:
        murah = 1
    else:
        murah = (6-x)/(6-3)

    #biasa
    if x <= 2 and x > 7:
        biasa = 0
    elif 2 < x <= 4:
        biasa = (x-3)/(4-3)
    elif 4 < x <= 6:
        biasa = 1
    elif 6 < x <= 7:
        biasa = (7-x)/(7-6)
    else:
        biasa = 0

    result = [mahal, biasa, murah]
    return result
```

Function ini berfungsi untuk membuat desain membership untuk harga disini kita membuat 3 linguistik yaitu mahal, biasa, dan murah.

### 3.5 fungsi rules

```
#inference
def rules(x, y): #x = servis, y = harga
    #menggunakan clipping technique
    excellent = [min(x[0], y[1]), min(x[0], y[2]), min(x[1], y[2])]
    normal = [min(x[1], y[0]), min(x[1], y[1]), min(x[0], y[0]), min(x[2], y[2])]
    bad = [min(x[2], y[0]), min(x[2], y[1])]

    #menggunakan disjunction rule
    excellent = max(excellent)
    normal = max(normal)
    bad = max(bad)

    result = [excellent, normal, bad]
    return result
```

Pada fungsi ini kita sudah berada pada tahap inference dimana kita akan membuat rules yang menentukan apakah termasuk ke dalam excellent, normal, atau bad berdasarkan input dari servis dan harga.

### 3.6 fungsi defuzzification

```
# defusifikasi dengan metode Takagi-Sugeno-style
def defuzzification(p):
    z = ((p[0]*95)+(p[1]*70)+(p[2]*55))/(p[0]+p[1]+p[2])

    return z
```

Setelah inference kita masuk ke dalam metode defuzzifikasi yang merupakan langkah terakhir dalam suatu sistem logika fuzzy dimana tujuannya adalah mengkonversi setiap hasil dari inference engine yang diekspresikan dalam bentuk fuzzy set ke suatu bilangan real. Pada kali ini kita menggunakan metode Takagi Sugeno Style.

### 3.7 show data

```
#untuk memunculkan data
for i in range(100):
    d = servis[i]
    e = harga[i]
    f = rules(fServis(d), fHarga(e))
    print("data ke:", i+1, "Value Servis:", fServis(d), "Value Harga:", fHarga(e), "Hasil inference:", rules(fServis(d), fHarga(e)), "Hasil defuzzification:", defuzzification(f))
```

Pada bagian ini akan diperlihatkan data mengenai nilai dari servis, nilai dari harga, hasil inference dan hasil defuzzifikasi.

### 3.8 final

```
: #mengurutkan data
data = []

for i in range(100):
    c = ID[i]
    d = servis[i]
    e = harga[i]
    f = rules(fServis(d), fHarga(e))
    data += ([defuzzification(f)])

#Merubah array menjadi data frame
data1 = pd.DataFrame(data)
data1.columns = ['Score']

#untuk menggabungkan data1 dengan noID
noID = ID
data1['ID Bengkel'] = noID

#sorting
data1 = data1.sort_values(by='Score', ascending = False)[:10]
data1

#membuat file excel
file = pd.ExcelWriter('peringkat.xlsx')

#memindahkan data1 ke excel
data1.to_excel(file)

#menyimpan hasil file yang sudah di export
file.save()
print("Data Peringkat Bengkel berhasil di export ke dalam file excel")
```

Pada bagian final ini kita melakukan sorting secara descending berdasarkan nilai dari hasil defuzzifikasi. Setelah itu kita mengambil 10 data pertama yang nantinya akan kita export ke dalam file peringkat.xlsx.

### 3.11 output

	id	servis	harga
0	1	58	7
1	2	54	1
2	3	98	2
3	4	52	4
4	5	11	4
...	...	...	...
95	96	30	1
96	97	25	3
97	98	27	10
98	99	8	6
99	100	11	8

```
data ke: 1 Value Servis: [0, 0.92, 0] Value Harga: [0.75, 0.0, 0] Hasil inference: [0, 0.75, 0] Hasil defuzzification: 70.0
data ke: 2 Value Servis: [0, 0.76, 0.05] Value Harga: [0, 0, 1] Hasil inference: [0.76, 0.05, 0] Hasil defuzzification: 93.45679012345678
data ke: 3 Value Servis: [1, 0, 0] Value Harga: [0, 0, 1] Hasil inference: [1, 0, 0] Hasil defuzzification: 95.0
data ke: 4 Value Servis: [0, 0.68, 0.15] Value Harga: [0, 1.0, 0] Hasil inference: [0, 0.68, 0.15] Hasil defuzzification: 67.28915662650601
data ke: 5 Value Servis: [0, 0, 1] Value Harga: [0, 1.0, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 6 Value Servis: [0, 0.96, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 0.96, 0] Hasil defuzzification: 70.0
data ke: 7 Value Servis: [0, 1, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 1, 0] Hasil defuzzification: 70.0
data ke: 8 Value Servis: [0, 0, 1] Value Harga: [1, 0, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 9 Value Servis: [0, 0.4, 0.5] Value Harga: [0, 0, 1] Hasil inference: [0.4, 0.5, 0] Hasil defuzzification: 81.11111111111111
data ke: 10 Value Servis: [0, 0.04, 0.95] Value Harga: [1, 0, 0] Hasil inference: [0, 0.04, 0.95] Hasil defuzzification: 55.6060606060606
data ke: 11 Value Servis: [0, 0, 1] Value Harga: [0.25, 1, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 12 Value Servis: [0, 0.12, 0.85] Value Harga: [0.75, 0.0, 0] Hasil inference: [0, 0.12, 0.75] Hasil defuzzification: 57.06896551724138
data ke: 13 Value Servis: [1.0, 0.3333333333333333, 0] Value Harga: [0, 0.0, 1] Hasil inference: [1.0, 0, 0] Hasil defuzzification: 95.0
data ke: 14 Value Servis: [0, 0, 1] Value Harga: [1, 0, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 15 Value Servis: [0.8, 0.4666666666666667, 0] Value Harga: [0.25, 1, 0] Hasil inference: [0.8, 0.466666666666667, 0] Hasil defuzzification: 85.78947368421052
data ke: 16 Value Servis: [1, 0.2, 0] Value Harga: [0.5, 1, 0] Hasil inference: [1, 0.5, 0] Hasil defuzzification: 86.66666666666667
data ke: 17 Value Servis: [0, 1, 0] Value Harga: [0, 0.0, 1] Hasil inference: [1, 0, 0] Hasil defuzzification: 95.0
data ke: 18 Value Servis: [0, 0, 1] Value Harga: [1, 0, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 19 Value Servis: [0, 0.28, 0.65] Value Harga: [0, 0.0, 1] Hasil inference: [0.28, 0.65, 0] Hasil defuzzification: 77.5268817204301
data ke: 20 Value Servis: [0, 0.56, 0.3] Value Harga: [1, 0, 0] Hasil inference: [0, 0.56, 0.3] Hasil defuzzification: 64.76744186046511
data ke: 21 Value Servis: [0, 0.52, 0.35] Value Harga: [0, 0, 1] Hasil inference: [0.52, 0.35, 0] Hasil defuzzification: 84.9425287356322
data ke: 22 Value Servis: [0.9, 0.4, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 0.9, 0] Hasil defuzzification: 70.0
data ke: 23 Value Servis: [0, 0, 1] Value Harga: [0, 1.0, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 24 Value Servis: [1, 0, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 1, 0] Hasil defuzzification: 70.0
data ke: 25 Value Servis: [0, 1, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 1, 0] Hasil defuzzification: 70.0
data ke: 26 Value Servis: [0, 0, 1] Value Harga: [0, 0, 1] Hasil inference: [0, 1, 0] Hasil defuzzification: 70.0
data ke: 27 Value Servis: [0, 0.96, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 0.96, 0] Hasil defuzzification: 70.0
data ke: 28 Value Servis: [0, 0.36, 0.55] Value Harga: [0, 0.0, 1] Hasil inference: [0.36, 0.55, 0] Hasil defuzzification: 79.89010989010987
data ke: 29 Value Servis: [0, 0, 1] Value Harga: [1, 0, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 30 Value Servis: [0, 0, 1] Value Harga: [0.5, 1, 0] Hasil inference: [0, 0, 1] Hasil defuzzification: 55.0
data ke: 31 Value Servis: [0.4, 0.7333333333333333, 0] Value Harga: [1, 0, 0] Hasil inference: [0, 0.7333333333333333, 0] Hasil defuzzification: 70.0
```

[illegible]

Links	3 months ago
Music	3 months ago
OneDrive	3 months ago
Pictures	an hour ago
Saved Games	3 months ago
Searches	3 months ago
Videos	15 hours ago
VirtualBox VMs	4 days ago
• AI.ipynb	a month ago
awdwadawd....	an hour ago
bengkel.xlsx	21 days ago
• fuzzy.ipynb	a minute ago
• peringkat.xlsx	2 minutes ago

```

[ ]:

```

	Score	ID Bengkel
2	95.00000	3
74	95.00000	75
33	95.00000	34
43	95.00000	44
16	95.00000	17
51	95.00000	52
91	95.00000	92
12	95.00000	13
90	95.00000	91
1	93.45679	2

Data Peringkat Bengkel berhasil di export ke dalam file excel

```

[ ]:

```

```

[ ]:

```

## **BAB IV**

### **KESIMPULAN**

Setelah kita melakukan implementasi pada fuzzy logic kita mendapatkan bahwasannya untuk mencari bengkel terbaik kita tidak bisa menentukan siapa bengkel terbaik secara asal. Pada fuzzy logic ini, kita dituntut untuk melihat ke dalam berbagai aspek, mulai dari aspek servis sampai harga harus kita perhatikan secara matang - matang. Bisa kita lihat pada dataset yang telah diberikan terdapat contoh data dimana ada bengkel yang memiliki harga 10 tetapi memiliki nilai servis sebesar 30 yang berarti bahwa aspek harga tidak menjadi acuan utama kita dalam menentukan bengkel terbaik tetapi aspek penilaian pada servis juga sangat berpengaruh.



## **Pembagian dalam pengerjaan tugas besar AI:**

Ramadhan Aditya Ibrahim:

- Pembuatan Code pada bagian inisiasi sampai pada fungsi fHarga
- Pembuatan Laporan dan hasil rangkuman.

Ryan Chandra Hadi:

- Pembuatan Code pada bagian tahap inference sampai menentukan metode defuzzification
- Pembuatan Laporan dan crosscheck nilai defuzzification.

## **Berikut link video presentasi kelompok kami:**

Via G-Drive:

[https://drive.google.com/file/d/1yWqiSTrrfOErP0HsC4-LV\\_rpTA8TOFdz/view?usp=sharing](https://drive.google.com/file/d/1yWqiSTrrfOErP0HsC4-LV_rpTA8TOFdz/view?usp=sharing)