LAPORAN PRAKTIKUM
KECERDASAN BUATAN
"LOGIKA FUZZY"

NAMA : **AFRIDHO IKHSAN**
NPM : **2210631170002**
KELAS : **4A**

PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SINGAPERBANGSA KARAWANG
2023

DAFTAR ISI

LATIHAN

Latihan: Uji Coba Menggunakan Berbagai Jenis Fungsi Keanggotaan (Membership Function)

1. Gunakan fungsi keanggotaan Trapesium dan Gaussian untuk data yang sama dan tunjukkan hasilnya!
2. Apakah terjadi perbedaan nilai derajat keanggotaan pada saat menggunakan jenis fungsi keanggotaan yang berbeda? Tunjukkan dan jelaskan!
3. Tampilkan hasil penentuan karyawan tetap atau karyawan kontrak beserta nama karyawannya!

JAWABAN

1. Gunakan fungsi keanggotaan Trapesium dan Gaussian untuk data yang sama dan tunjukkan hasilnya!

   Jawab:

   Pada kode di bawah, variable seperti: xage, xyos, xsalary; dan fungsi seperti : MembershipDeg1(), MembershipDeg1(), Status1(), Status2(), & RuleBasedGaussian() sudah didefinisikan pada kode pada pertemuan 4 (Saat menggunakan Fungsi Keanggotaan Triangular)

```python
# Mendefinisikan range data 'Umur' untuk Fungsi Keanggotaan
Trapesium
lo_age_trap_data = [25, 25, 35, 40]
mi_age_trap_data = [30, 30, 40, 50]
hi_age_trap_data = [40, 40, 55, 55]

# Mendefinisikan range data 'Tahun Kedinasan' untuk Fungsi
Keanggotaan Trapesium
lo_yos_trap_data = [3, 3, 9, 12]
hi_yos_trap_data = [9, 12, 18, 18]

# Mendefinisikan range data 'Gaji' untuk Fungsi Keanggotaan
Trapesium
lo_salary_trap_data = [1000, 1000, 3000, 3500]
mi_salary_trap_data = [2000, 3500, 4500, 6000]
hi_salary_trap_data = [4500, 6000, 7000, 7000]
```

```python
# Mendefinisikan fungsi keanggotaan untuk Umur menggunakan
fungsi keanggotaan Trapesium
lo_age_trap = fuz.trapmf(xage, lo_age_trap_data)
mi_age_trap = fuz.trapmf(xage, mi_age_trap_data)
hi_age_trap = fuz.trapmf(xage, hi_age_trap_data)
```

```python
# Mendefinisikan fungsi keanggotaan untuk Tahun Kedinasan
menggunakan fungsi keanggotaan Trapesium
lo_yos_trap = fuz.trapmf(xyos, lo_yos_trap_data)
hi_yos_trap = fuz.trapmf(xyos, hi_yos_trap_data)
```

```python
# Mendefinisikan fungsi keanggotaan untuk Gaji menggunakan
fungsi keanggotaan Trapesium
lo_salary_trap = fuz.trapmf(xsalary, lo_salary_trap_data)
mi_salary_trap = fuz.trapmf(xsalary, mi_salary_trap_data)
hi_salary_trap = fuz.trapmf(xsalary, hi_salary_trap_data)
```

```python
# Menampilkan fungsi keanggotaan untuk Umur (Trapesium)
plt.figure(figsize=(8, 3))
plt.plot(xage, lo_age_trap, 'b', linewidth=1.5, label='Low')
```

```python
plt.plot(xage, mi_age_trap, 'g', linewidth=1.5,
label='Medium')
plt.plot(xage, hi_age_trap, 'r', linewidth=1.5, label='High')
plt.title('Membership Functions for Age (Trapezoidal)')
plt.legend()
plt.show()
# Menampilkan fungsi keanggotaan untuk Tahun Kedinasan
(Trapesium)
plt.figure(figsize=(8, 3))
plt.plot(xyos, lo_yos_trap, 'b', linewidth=1.5, label='Low')
plt.plot(xyos, hi_yos_trap, 'r', linewidth=1.5, label='High')
plt.title('Membership Functions for Year of Service
(Trapezoidal)')
plt.legend()
plt.show()

# Menampilkan fungsi keanggotaan untuk Gaji (Trapesium)
plt.figure(figsize=(8, 3))
plt.plot(xsalary, lo_salary_trap, 'b', linewidth=1.5,
label='Low')
plt.plot(xsalary, mi_salary_trap, 'g', linewidth=1.5,
label='Medium')
plt.plot(xsalary, hi_salary_trap, 'r', linewidth=1.5,
label='High')
plt.title('Membership Functions for Salary (Trapezoidal)')
plt.legend()
plt.show()
```


Membership Functions for Age (Trapezoidal)

Membership Functions for Year of Service (Trapezoidal)

Membership Functions for Salary (Trapezoidal)
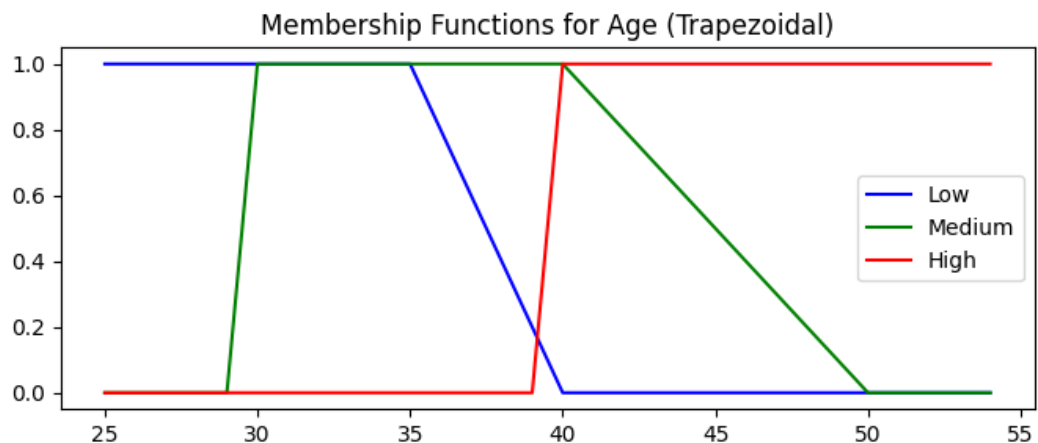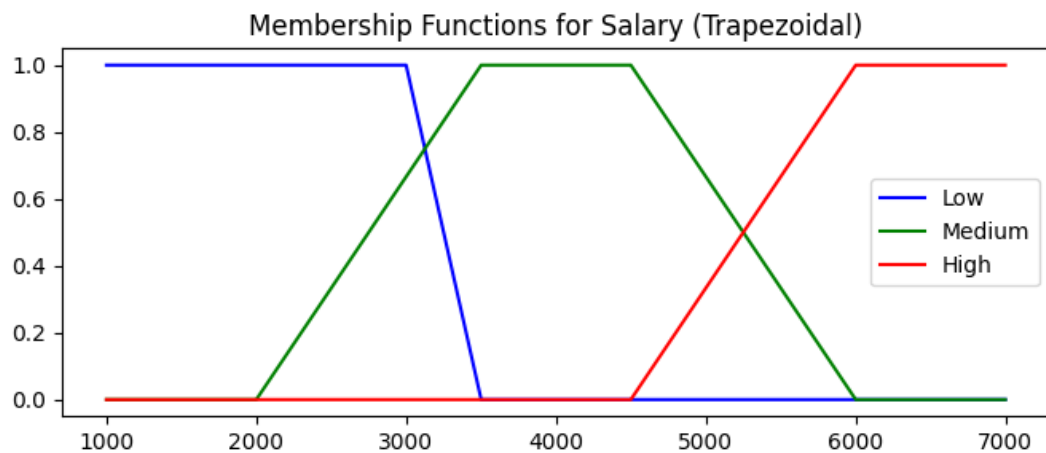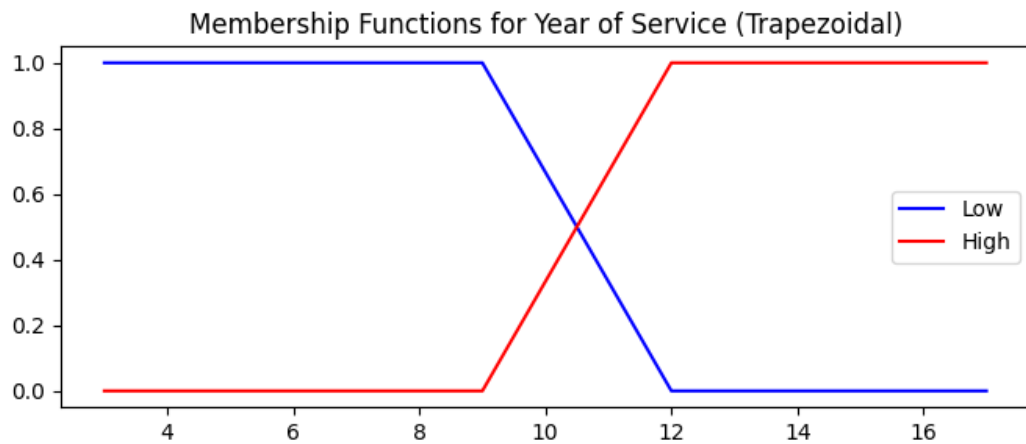

```
# Pendefinisian fungsi perhitungan nilai tengah & stdev
(Standar Deviasi) sebagai nilai yang akan dibutuhkan dalam
penggunaan Fungsi Keanggotaan Gaussian

# Hal tersebut perlu dilakukan, karena method fuz.gaussmf()
membutuhkan nilai tengah & nilai standar deviasi dari interval
beberapa nilai.

from math import sqrt

def mean (numbers):
  return sum(numbers)/float (len (numbers))

def stdev(numbers):
  avg = mean (numbers)
  variance=sum([(x-avg)**2 for x in numbers])/ float
(len(numbers)-1)
  return sqrt(variance)
```

```python
def nilaiTengahDari(numbers):
  return ((numbers[0] + numbers[-1]) / 2)
```

```python
# Mendefinisikan fungsi keanggotaan untuk Usia menggunakan
fungsi keanggotaan Gaussian
lo_age_gaussian = fuz.gaussmf(xage,
nilaiTengahDari(lo_age_trap_data), stdev(lo_age_trap_data))
mi_age_gaussian = fuz.gaussmf(xage,
nilaiTengahDari(mi_age_trap_data), stdev(mi_age_trap_data))
hi_age_gaussian = fuz.gaussmf(xage,
nilaiTengahDari(hi_age_trap_data), stdev(hi_age_trap_data))
```

```python
# Mendefinisikan fungsi keanggotaan untuk Tahun Kedinasan
menggunakan fungsi keanggotaan Gaussian lo_yos_gaussian =
fuz.gaussmf(xyos, nilaiTengahDari(lo_yos_trap_data),
stdev(lo_yos_trap_data))
hi_yos_gaussian = fuz.gaussmf(xyos,
nilaiTengahDari(hi_yos_trap_data), stdev(hi_yos_trap_data))

# Mendefinisikan fungsi keanggotaan untuk Gaji menggunakan
fungsi keanggotaan Gaussian
lo_salary_gaussian = fuz.gaussmf(xsalary,
nilaiTengahDari(lo_salary_trap_data),
stdev(lo_salary_trap_data))
mi_salary_gaussian = fuz.gaussmf(xsalary,
nilaiTengahDari(mi_salary_trap_data),
stdev(mi_salary_trap_data))
hi_salary_gaussian = fuz.gaussmf(xsalary,
nilaiTengahDari(hi_salary_trap_data),
stdev(hi_salary_trap_data))
```

```python
# Menampilkan fungsi keanggotaan untuk Umur (Gaussian)
plt.figure(figsize=(8, 3))
plt.plot(xage, lo_age_gaussian, 'b', linewidth=1.5,
label='Low')
plt.plot(xage, mi_age_gaussian, 'g', linewidth=1.5,
label='Medium')
plt.plot(xage, hi_age_gaussian, 'r', linewidth=1.5,
label='High')
plt.title('Membership Functions for Age (Gaussian)')
plt.legend()
plt.show()

# Menampilkan fungsi keanggotaan untuk Tahun Kedinasan
(Gaussian) plt.figure(figsize=(8, 3))
```
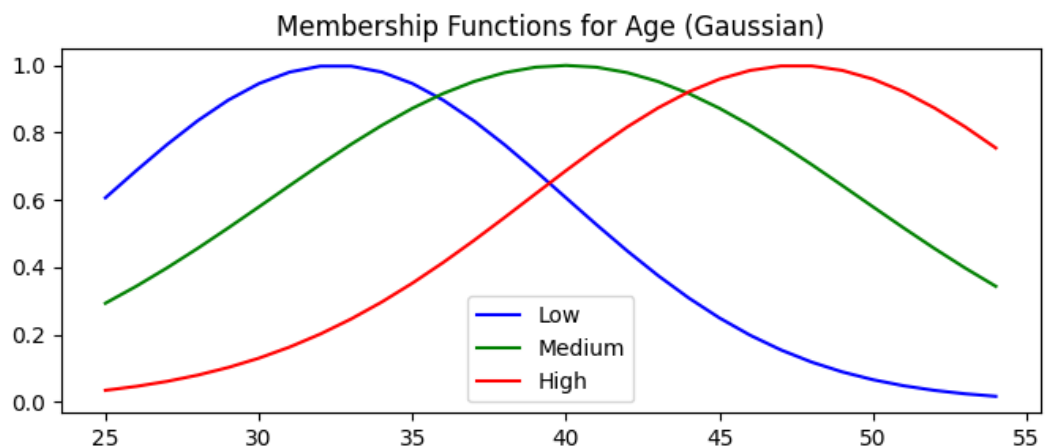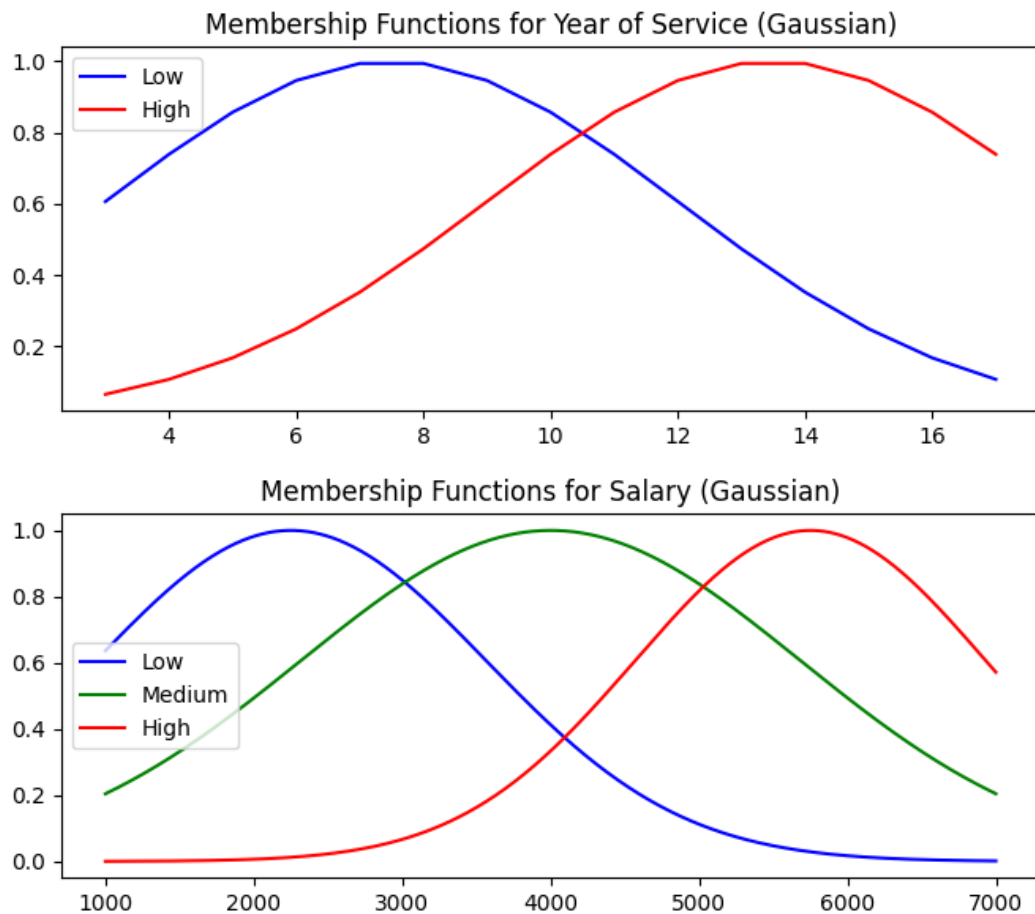
```python
plt.plot(xyos, lo_yos_gaussian, 'b', linewidth=1.5,
label='Low')
plt.plot(xyos, hi_yos_gaussian, 'r', linewidth=1.5,
label='High')
plt.title('Membership Functions for Year of Service
(Gaussian)')
plt.legend()
plt.show()

# Menampilkan fungsi keanggotaan untuk Gaji (Gaussian)
plt.figure(figsize=(8, 3))
plt.plot(xsalary, lo_salary_gaussian, 'b', linewidth=1.5,
label='Low')
plt.plot(xsalary, mi_salary_gaussian, 'g', linewidth=1.5,
label='Medium')
plt.plot(xsalary, hi_salary_gaussian, 'r', linewidth=1.5,
label='High')
plt.title('Membership Functions for Salary (Gaussian)')
plt.legend()
plt.show()
```


Membership Functions for Age (Gaussian)

Membership Functions for Year of Service (Gaussian)

Membership Functions for Salary (Gaussian)

```
# Hitung derajat keanggotaan berdasarkan Usia menggunakan
fungsi keanggotaan trapesium
age_member_01_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age01)
age_member_02_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age02)
age_member_03_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age03)
age_member_04_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age04)
age_member_05_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age05)
age_member_06_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age06)
age_member_07_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age07)
age_member_08_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age08)
age_member_09_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age09)
```

```python
age_member_10_trap = MembershipDeg1(xage, lo_age_trap,
mi_age_trap, hi_age_trap, age10)

# Menampilkan derajat keanggotaan untuk Usia menggunakan
fungsi keanggotaan trap
print("Membership Degrees for Age (trap):")
print("Age 01:", age_member_01_trap)
print("Age 02:", age_member_02_trap)
print("Age 03:", age_member_03_trap)
print("Age 04:", age_member_04_trap)
print("Age 05:", age_member_05_trap)
print("Age 06:", age_member_06_trap)
print("Age 07:", age_member_07_trap)
print("Age 08:", age_member_08_trap)
print("Age 09:", age_member_09_trap)
print("Age 10:", age_member_10_trap)

# Penentuan status umur (Trapesium)
trap_age_status_01=Status1(age_member_01_trap, 'young',
'middle-aged', 'old')
trap_age_status_02=Status1(age_member_02_trap, 'young',
'middle-aged', 'old')
trap_age_status_03=Status1(age_member_03_trap, 'young',
'middle-aged', 'old')
trap_age_status_04=Status1(age_member_04_trap, 'young',
'middle-aged', 'old')
trap_age_status_05=Status1 (age_member_05_trap, 'young',
'middle-aged', 'old')
trap_age_status_06=Status1(age_member_06_trap, 'young',
'middle-aged', 'old')
trap_age_status_07=Status1 (age_member_07_trap, 'young',
'middle-aged', 'old')
trap_age_status_08=Status1(age_member_08_trap, 'young',
'middle-aged', 'old')
trap_age_status_09=Status1 (age_member_09_trap, 'young',
'middle-aged', 'old')
trap_age_status_10=Status1 (age_member_10_trap, 'young',
'middle-aged', 'old')

# Menampilkan status umur (Trapsium)
print("age_status_01: ",age_status_01)
print("age_status_02: ",age_status_02)
print("age_status_03: ", age_status_03)
print("age_status_04: ",age_status_04)
print("age_status_05: ", age_status_05)
print("age_status_06: ", age_status_06)
print("age_status_07: ", age_status_07)
```

```python
print("age_status_07: ", age_status_08)
print("age_status_09: ",age_status_09)
print("age_status_10: ", age_status_10)

# Hitung derajat keanggotaan berdasarkan Tahun Kedinasan
menggunakan fungsi keanggotaan trapesium
yos_member_01_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos01)
yos_member_02_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos02)
yos_member_03_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos03)
yos_member_04_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos04)
yos_member_05_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos05)
yos_member_06_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos06)
yos_member_07_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos07)
yos_member_08_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos08)
yos_member_09_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos09)
yos_member_10_trap = MembershipDeg2(xyos, lo_yos_trap,
hi_yos_trap, yos10)

# Menampilkan derajat keanggotaan untuk Tahun Kedinasan
menggunakan fungsi keanggotaan trap
print("\nMembership Degrees for Year of Service (trap):")
print("YOS 01:", yos_member_01_trap)
print("YOS 02:", yos_member_02_trap)
print("YOS 03:", yos_member_03_trap)
print("YOS 04:", yos_member_04_trap)
print("YOS 05:", yos_member_05_trap)
print("YOS 06:", yos_member_06_trap)
print("YOS 07:", yos_member_07_trap)
print("YOS 08:", yos_member_08_trap)
print("YOS 09:", yos_member_09_trap)
print("YOS 10:", yos_member_10_trap)

# Penentuan status tahun kedinasan (Trapesium)
trap_yos_status_01=Status2 (yos_member_01_trap, 'new',
'loyal')
trap_yos_status_02=Status2 (yos_member_02_trap, 'new',
'loyal')
```

```python
trap_yos_status_03=Status2 (yos_member_03_trap, 'new',
'loyal')
trap_yos_status_04=Status2 (yos_member_04_trap, 'new',
'loyal')
trap_yos_status_05=Status2 (yos_member_05_trap, 'new',
'loyal')
trap_yos_status_06=Status2 (yos_member_06_trap, 'new',
'loyal')
trap_yos_status_07=Status2 (yos_member_07_trap, 'new',
'loyal')
trap_yos_status_08=Status2 (yos_member_08_trap, 'new',
'loyal')
trap_yos_status_09=Status2 (yos_member_09_trap, 'new',
'loyal')
trap_yos_status_10=Status2 (yos_member_10_trap, 'new',
'loyal')

# Menampilkan status Tahun Kedinasan (Trapesium)
print("yos_status_01: ",yos_status_01)
print("yos_status_02: ",yos_status_02)
print("yos_status_03: ",yos_status_03)
print("yos_status_04: ",yos_status_04)
print("yos_status_05: ",yos_status_05)
print("yos_status_06: ",yos_status_06)
print("yos_status_07: ",yos_status_07)
print("yos_status_08: ",yos_status_08)
print("yos_status_09: ",yos_status_09)
print("yos_status_10: ",yos_status_10)

# Menghitung derajat keanggotaan berdasarkan Gaji menggunakan
fungsi keanggotaan trapesium
salary_member_01_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary01)
salary_member_02_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary02)
salary_member_03_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary03)
salary_member_04_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary04)
salary_member_05_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary05)
salary_member_06_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary06)
salary_member_07_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary07)
salary_member_08_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary08)
```

```python
salary_member_09_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary09)
salary_member_10_trap = MembershipDeg1(xsalary,
lo_salary_trap, mi_salary_trap, hi_salary_trap, salary10)

# Menampilkan derajat keanggotaan untuk Gaji menggunakan
fungsi keanggotaan trapesium
print("\nMembership Degrees for Salary (trap):")
print("Salary 01:", salary_member_01_trap)
print("Salary 02:", salary_member_02_trap)
print("Salary 03:", salary_member_03_trap)
print("Salary 04:", salary_member_04_trap)
print("Salary 05:", salary_member_05_trap)
print("Salary 06:", salary_member_06_trap)
print("Salary 07:", salary_member_07_trap)
print("Salary 08:", salary_member_08_trap)
print("Salary 09:", salary_member_09_trap)
print("Salary 10:", salary_member_10_trap)

# Penentuan status gaji (Trapesium)
trap_salary_status_01=Status1 (salary_member_01_trap, 'low',
'medium', 'high')
trap_salary_status_02=Status1(salary_member_02_trap, 'low',
'medium', 'high')
trap_salary_status_03=Status1 (salary_member_03_trap, 'low',
'medium', 'high')
trap_salary_status_04=Status1 (salary_member_04_trap, 'low',
'medium', 'high')
trap_salary_status_05=Status1 (salary_member_05_trap, 'low',
'medium', 'high')
trap_salary_status_06=Status1 (salary_member_06_trap, 'low',
'medium', 'high')
trap_salary_status_07=Status1 (salary_member_07_trap, 'low',
'medium', 'high')
trap_salary_status_08=Status1 (salary_member_08_trap, 'low',
'medium', 'high')
trap_salary_status_09=Status1(salary_member_09_trap, 'low',
'medium', 'high')
trap_salary_status_10=Status1 (salary_member_10_trap, 'low',
'medium', 'high')

# Menampilkan status Gaji(Trapesium)
print("salary_status_01: ",trap_salary_status_01)
print("salary_status_02: ",trap_salary_status_02)
print("salary_status_03: ",trap_salary_status_03)
print("salary_status_04: ",trap_salary_status_04)
print("salary_status_05: ",trap_salary_status_05)
```

```python
print("salary_status_06: ",trap_salary_status_06)
print("salary_status_07: ",trap_salary_status_07)
print("salary_status_08: ",trap_salary_status_08)
print("salary_status_09: ",trap_salary_status_09)
print("salary_status_10: ",trap_salary_status_10)

# Menampilkan hasil Fuzzy Rule-Based menggunakan fungsi
keanggotaan trapesium
employee_status01_trap = RuleBasedtrap(age_status_01,
yos_status_01, salary_status_01)
employee_status02_trap = RuleBasedtrap(age_status_02,
yos_status_02, salary_status_02)
employee_status03_trap = RuleBasedtrap(age_status_03,
yos_status_03, salary_status_03)
employee_status04_trap = RuleBasedtrap(age_status_04,
yos_status_04, salary_status_04)
employee_status05_trap = RuleBasedtrap(age_status_05,
yos_status_05, salary_status_05)
employee_status06_trap = RuleBasedtrap(age_status_06,
yos_status_06, salary_status_06)
employee_status07_trap = RuleBasedtrap(age_status_07,
yos_status_07, salary_status_07)
employee_status08_trap = RuleBasedtrap(age_status_08,
yos_status_08, salary_status_08)
employee_status09_trap = RuleBasedtrap(age_status_09,
yos_status_09, salary_status_09)
employee_status10_trap = RuleBasedtrap(age_status_10,
yos_status_10, salary_status_10)

# Menampilkan hasil Fuzzy Rule-Based menggunakan fungsi
keanggotaan trap
print("\nOutcome of the Fuzzy Rule-Based (trap):")
print("Employee Status 01:", employee_status01_trap)
print("Employee Status 02:", employee_status02_trap)
print("Employee Status 03:", employee_status03_trap)
print("Employee Status 04:", employee_status04_trap)
print("Employee Status 05:", employee_status05_trap)
print("Employee Status 06:", employee_status06_trap)
print("Employee Status 07:", employee_status07_trap)
print("Employee Status 08:", employee_status08_trap)
print("Employee Status 09:", employee_status09_trap)
print("Employee Status 10:", employee_status10_trap)
```

Membership Degrees for Age (trap):
Age 01: (1.0, 1.0, 0.0)
Age 02: (1.0, 0.0, 0.0)
Age 03: (1.0, 1.0, 0.0)

Age 04: (0.0, 0.9, 1.0)
Age 05: (0.2, 1.0, 0.0)
Age 06: (1.0, 1.0, 0.0)
Age 07: (0.6, 1.0, 0.0)
Age 08: (0.0, 0.1, 1.0)
Age 09: (0.6, 1.0, 0.0)
Age 10: (0.8, 1.0, 0.0)

age_status_01: middle-aged
age_status_02: young
age_status_03: young
age_status_04: middle-aged
age_status_05: middle-aged
age_status_06:
age_status_07: middle-aged
age_status_07: old
age_status_09: middle-aged
age_status_10: middle-aged

Membership Degrees for Year of Service (trap):
YOS 01: (1.0, 0.0)
YOS 02: (0.0, 0.0)
YOS 03: (1.0, 0.0)
YOS 04: (0.0, 1.0)
YOS 05: (0.0, 1.0)
YOS 06: (0.0, 0.0)
YOS 07: (1.0, 0.0)
YOS 08: (0.0, 1.0)
YOS 09: (1.0, 0.0)
YOS 10: (0.0, 1.0)

yos_status_01: new
yos_status_02:
yos_status_03: new
yos_status_04: loyal
yos_status_05: loyal
yos_status_06:
yos_status_07: new
yos_status_08: loyal
yos_status_09: new
yos_status_10: loyal

Membership Degrees for Salary (trap):
Salary 01: (1.0, 0.6266666666666667, 0.0)
Salary 02: (0.12, 0.96, 0.0)
Salary 03: (1.0, 0.6666666666666666, 0.0)
Salary 04: (0.0, 1.0, 0.0)
Salary 05: (0.0, 0.0, 1.0)

Salary 06: (1.0, 0.13333333333333333, 0.0)
Salary 07: (0.0, 1.0, 0.0)
Salary 08: (0.0, 0.0, 1.0)
Salary 09: (0.0, 0.6666666666666666, 0.3333333333333333)
Salary 10: (0.0, 0.6533333333333333, 0.3466666666666667)

salary_status_01: low
salary_status_02: medium
salary_status_03: low
salary_status_04: medium
salary_status_05: high
salary_status_06: low
salary_status_07: medium
salary_status_08: high
salary_status_09: medium
salary_status_10: medium

Outcome of the Fuzzy Rule-Based (trap):
Employee Status 01: a contract employee
Employee Status 02: unknown
Employee Status 03: a contract employee
Employee Status 04: a permanent employee
Employee Status 05: a permanent employee
Employee Status 06: unknown
Employee Status 07: a contract employee
Employee Status 08: a permanent employee
Employee Status 09: a contract employee
Employee Status 10: a permanent employee

```
# Menghitung derajat keanggotaan berdasarkan Usia menggunakan
fungsi keanggotaan Gaussian
age_member_01_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age01)
age_member_02_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age02)
age_member_03_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age03)
age_member_04_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age04)
age_member_05_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age05)
age_member_06_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age06)
age_member_07_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age07)
age_member_08_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age08)
```

```python
age_member_09_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age09)
age_member_10_gaussian = MembershipDeg1(xage, lo_age_gaussian,
mi_age_gaussian, hi_age_gaussian, age10)

# Menampilkan derajat keanggotaan Usia menggunakan fungsi
keanggotaan Gaussian
print("Membership Degrees for Age (Gaussian):")
print("Age 01:", age_member_01_gaussian)
print("Age 02:", age_member_02_gaussian)
print("Age 03:", age_member_03_gaussian)
print("Age 04:", age_member_04_gaussian)
print("Age 05:", age_member_05_gaussian)
print("Age 06:", age_member_06_gaussian)
print("Age 07:", age_member_07_gaussian)
print("Age 08:", age_member_08_gaussian)
print("Age 09:", age_member_09_gaussian)
print("Age 10:", age_member_10_gaussian)


# Menghitung status umur (Gaussian)
gaussian_age_status_01=Status1(age_member_01_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_02=Status1(age_member_02_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_03=Status1(age_member_03_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_04=Status1(age_member_04_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_05=Status1 (age_member_05_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_06=Status1(age_member_06_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_07=Status1 (age_member_07_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_08=Status1(age_member_08_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_09=Status1 (age_member_09_gaussian,
'young', 'middle-aged', 'old')
gaussian_age_status_10=Status1 (age_member_10_gaussian,
'young', 'middle-aged', 'old')

# Menampilkan status Umur(Gaussian)
print("age_status_01: ",age_status_01)
print("age_status_02: ",age_status_02)
print("age_status_03: ", age_status_03)
print("age_status_04: ",age_status_04)
```

```python
print("age_status_05: ", age_status_05)
print("age_status_06: ", age_status_06)
print("age_status_07: ", age_status_07)
print("age_status_07: ", age_status_08)
print("age_status_09: ",age_status_09)
print("age_status_10: ", age_status_10)

# Menghitung derajat keanggotaan berdasarkan Tahun Kedinasan
menggunakan fungsi keanggotaan Gaussian
yos_member_01_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos01)
yos_member_02_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos02)
yos_member_03_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos03)
yos_member_04_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos04)
yos_member_05_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos05)
yos_member_06_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos06)
yos_member_07_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos07)
yos_member_08_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos08)
yos_member_09_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos09)
yos_member_10_gaussian = MembershipDeg2(xyos, lo_yos_gaussian,
hi_yos_gaussian, yos10)

# Menampilkan derajat keanggotaan Tahun Kedinasan menggunakan
fungsi keanggotaan Gaussian
print("\nMembership Degrees for Year of Service (Gaussian):")
print("YOS 01:", yos_member_01_gaussian)
print("YOS 02:", yos_member_02_gaussian)
print("YOS 03:", yos_member_03_gaussian)
print("YOS 04:", yos_member_04_gaussian)
print("YOS 05:", yos_member_05_gaussian)
print("YOS 06:", yos_member_06_gaussian)
print("YOS 07:", yos_member_07_gaussian)
print("YOS 08:", yos_member_08_gaussian)
print("YOS 09:", yos_member_09_gaussian)
print("YOS 10:", yos_member_10_gaussian)


# Menghitung status Tahun Kedinasan (Gaussian)
```

```python
gaussian_yos_status_01=Status2 (yos_member_01_gaussian, 'new',
'loyal')
gaussian_yos_status_02=Status2 (yos_member_02_gaussian, 'new',
'loyal')
gaussian_yos_status_03=Status2 (yos_member_03_gaussian, 'new',
'loyal')
gaussian_yos_status_04=Status2 (yos_member_04_gaussian, 'new',
'loyal')
gaussian_yos_status_05=Status2 (yos_member_05_gaussian, 'new',
'loyal')
gaussian_yos_status_06=Status2 (yos_member_06_gaussian, 'new',
'loyal')
gaussian_yos_status_07=Status2 (yos_member_07_gaussian, 'new',
'loyal')
gaussian_yos_status_08=Status2 (yos_member_08_gaussian, 'new',
'loyal')
gaussian_yos_status_09=Status2 (yos_member_09_gaussian, 'new',
'loyal')
gaussian_yos_status_10=Status2 (yos_member_10_gaussian, 'new',
'loyal')

# Menampilkan status Tahun Kedinasan (Gaussian)
print("yos_status_01: ",yos_status_01)
print("yos_status_02: ",yos_status_02)
print("yos_status_03: ",yos_status_03)
print("yos_status_04: ",yos_status_04)
print("yos_status_05: ",yos_status_05)
print("yos_status_06: ",yos_status_06)
print("yos_status_07: ",yos_status_07)
print("yos_status_08: ",yos_status_08)
print("yos_status_09: ",yos_status_09)
print("yos_status_10: ",yos_status_10)


# Menghitung derajat keanggotaan berdasarkan Gaji menggunakan
fungsi keanggotaan Gaussian
salary_member_01_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary01)
salary_member_02_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary02)
salary_member_03_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary03)
```

```python
salary_member_04_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary04)
salary_member_05_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary05)
salary_member_06_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary06)
salary_member_07_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary07)
salary_member_08_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary08)
salary_member_09_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary09)
salary_member_10_gaussian = MembershipDeg1(xsalary,
lo_salary_gaussian, mi_salary_gaussian, hi_salary_gaussian,
salary10)


# Menampilkan derajat keanggotaan Gaji menggunakan fungsi
keanggotaan Gaussian
print("\nMembership Degrees for Salary (Gaussian):")
print("Salary 01:", salary_member_01_gaussian)
print("Salary 02:", salary_member_02_gaussian)
print("Salary 03:", salary_member_03_gaussian)
print("Salary 04:", salary_member_04_gaussian)
print("Salary 05:", salary_member_05_gaussian)
print("Salary 06:", salary_member_06_gaussian)
print("Salary 07:", salary_member_07_gaussian)
print("Salary 08:", salary_member_08_gaussian)
print("Salary 09:", salary_member_09_gaussian)
print("Salary 10:", salary_member_10_gaussian)


# Menghitung status Gaji (Gaussian)
gaussian_salary_status_01=Status1(salary_member_01_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_02=Status1(salary_member_02_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_03=Status1(salary_member_03_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_04=Status1(salary_member_04_gaussian,
'low', 'medium', 'high')
```

```
gaussian_salary_status_05=Status1 (salary_member_05_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_06=Status1 (salary_member_06_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_07=Status1 (salary_member_07_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_08=Status1 (salary_member_08_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_09=Status1(salary_member_09_gaussian,
'low', 'medium', 'high')
gaussian_salary_status_10=Status1 (salary_member_10_gaussian,
'low', 'medium', 'high')

# Menampilkan status Gaji (Gaussian)
print("salary_status_01: ",gaussian_salary_status_01)
print("salary_status_02: ",gaussian_salary_status_02)
print("salary_status_03: ",gaussian_salary_status_03)
print("salary_status_04: ",gaussian_salary_status_04)
print("salary_status_05: ",gaussian_salary_status_05)
print("salary_status_06: ",gaussian_salary_status_06)
print("salary_status_07: ",gaussian_salary_status_07)
print("salary_status_08: ",gaussian_salary_status_08)
print("salary_status_09: ",gaussian_salary_status_09)
print("salary_status_10: ",gaussian_salary_status_10)

# Menampilkan hasil Fuzzy Rule-Based menggunakan fungsi
keanggotaan Gaussian
employee_status01_gaussian =
RuleBasedGaussian(gaussian_age_status_01,
gaussian_yos_status_01, gaussian_salary_status_01)
employee_status02_gaussian =
RuleBasedGaussian(gaussian_age_status_02,
gaussian_yos_status_02, gaussian_salary_status_02)
employee_status03_gaussian =
RuleBasedGaussian(gaussian_age_status_03,
gaussian_yos_status_03, gaussian_salary_status_03)
employee_status04_gaussian =
RuleBasedGaussian(gaussian_age_status_04,
gaussian_yos_status_04, gaussian_salary_status_04)
employee_status05_gaussian =
RuleBasedGaussian(gaussian_age_status_05,
gaussian_yos_status_05, gaussian_salary_status_05)
employee_status06_gaussian =
RuleBasedGaussian(gaussian_age_status_06,
gaussian_yos_status_06, gaussian_salary_status_06)
```

```
employee_status07_gaussian =
RuleBasedGaussian(gaussian_age_status_07,
gaussian_yos_status_07, gaussian_salary_status_07)
employee_status08_gaussian =
RuleBasedGaussian(gaussian_age_status_08,
gaussian_yos_status_08, gaussian_salary_status_08)
employee_status09_gaussian =
RuleBasedGaussian(gaussian_age_status_09,
gaussian_yos_status_09, gaussian_salary_status_09)
employee_status10_gaussian =
RuleBasedGaussian(gaussian_age_status_10,
gaussian_yos_status_10, gaussian_salary_status_10)

# Menampilkan hasil Fuzzy Rule-Based menggunakan fungsi
keanggotaan Gaussian
print("\nOutcome of the Fuzzy Rule-Based (Gaussian):")
print("Employee Status 01:", employee_status01_gaussian)
print("Employee Status 02:", employee_status02_gaussian)
print("Employee Status 03:", employee_status03_gaussian)
print("Employee Status 04:", employee_status04_gaussian)
print("Employee Status 05:", employee_status05_gaussian)
print("Employee Status 06:", employee_status06_gaussian)
print("Employee Status 07:", employee_status07_gaussian)
print("Employee Status 08:", employee_status08_gaussian)
print("Employee Status 09:", employee_status09_gaussian)
print("Employee Status 10:", employee_status10_gaussian)
```

Membership Degrees for Age (Gaussian):
Age 01: (0.9459594689067654, 0.8725252928694238, 0.35286608145884896)
Age 02: (0.6869075574573879, 0.34332048512565055, 0.045882721684815775)
Age 03: (0.9459594689067654, 0.5795782787848095, 0.12981217685543806)
Age 04: (0.5261219640934565, 0.994560303568022, 0.7545251510033499)
Age 05: (0.6869075574573879, 0.994560303568022, 0.6177529450972508)
Age 06: (0.9801986733067553, 0.8217133754796329, 0.2967100142940453)
Age 07: (0.835270211411272, 0.9520945717121367, 0.47950545897489416)
Age 08: (0.08892161745938634, 0.6428665096502189, 0.9851119396030626)
Age 09: (0.835270211411272, 0.9520945717121367, 0.47950545897489416)
Age 10: (0.8968300597468688, 0.9164271268770867, 0.4140921825962409)

age_status_01: middle-aged
age_status_02: young
age_status_03: young
age_status_04: middle-aged
age_status_05: middle-aged
age_status_06:
age_status_07: middle-aged
age_status_07: old

age_status_09: middle-aged
age_status_10: middle-aged

Membership Degrees for Year of Service (Gaussian):
YOS 01: (0.6065306597126334, 0.06572852861653047)
YOS 02: (0.0, 0.0)
YOS 03: (0.9459594689067654, 0.24935220877729622)
YOS 04: (0.6065306597126334, 0.9459594689067654)
YOS 05: (0.47382672639536405, 0.9938461733264412)
YOS 06: (0.0, 0.0)
YOS 07: (0.7389912962803088, 0.10770114519003211)
YOS 08: (0.10770114519003211, 0.7389912962803088)
YOS 09: (0.8569968914352789, 0.16797323675753362)
YOS 10: (0.35232195499549696, 0.9938461733264412)

yos_status_01: new
yos_status_02:
yos_status_03: new
yos_status_04: loyal
yos_status_05: loyal
yos_status_06:
yos_status_07: new
yos_status_08: loyal
yos_status_09: new
yos_status_10: loyal

Membership Degrees for Salary (Gaussian):
Salary 01: (0.8713884064654819, 0.8201382519951156, 0.059104173980092156)
Salary 02: (0.6639989579926046, 0.9461622845710673, 0.14786748085292273)
Salary 03: (0.8498880853931386, 0.8382234324229999, 0.06660635910874733)
Salary 04: (0.4992247942554211, 0.9929660314487113, 0.2561250307821973)
Salary 05: (0.006874062557496255, 0.3618698767618531, 0.8595525777191432)
Salary 06: (0.9992773696568941, 0.564528332689429, 0.010950957778277046)
Salary 07: (0.34823689809438557, 0.9954925421591123, 0.4043040602572814)
Salary 08: (0.01714129078609727, 0.49367278838913037, 0.9778606930848421)
Salary 09: (0.11228140794800233, 0.8382234324229999, 0.8175096783558297)
Salary 10: (0.10875366814447272, 0.8322686447119416, 0.8262238376850651)

salary_status_01: low
salary_status_02: medium
salary_status_03: low
salary_status_04: medium
salary_status_05: high
salary_status_06: low
salary_status_07: medium

salary_status_08: high
salary_status_09: medium
salary_status_10: medium

Outcome of the Fuzzy Rule-Based (Gaussian):
Employee Status 01: a contract employee
Employee Status 02: unknown
Employee Status 03: a contract employee
Employee Status 04: a permanent employee
Employee Status 05: a permanent employee
Employee Status 06: unknown
Employee Status 07: a contract employee
Employee Status 08: a permanent employee
Employee Status 09: a contract employee
Employee Status 10: a permanent employee

2. Apakah terjadi perbedaan nilai derajat keanggotaan pada saat menggunakan jenis keanggotaan yang berbeda? Tunjukkan dan jelaskan!

Iya, terjadi perbedaan dalam nilai derajat keanggotaan saat menggunakan jenis fungsi keanggotaan yang berbeda. Perbedaan ini disebabkan oleh perbedaan bentuk kurva yang digunakan oleh masing-masing fungsi keanggotaan (Trapesium dan Gaussian), serta parameter yang digunakan untuk mendefinisikan kurva tersebut.

Mari kita bandingkan derajat keanggotaan untuk contoh tertentu dalam latihan sebelumnya menggunakan fungsi keanggotaan triangular, trapesium, dan Gaussian untuk atribut "Age".

Dalam kasus ini, mari kita lihat contoh untuk "Age 01":
Fungsi Keanggotaan Trapesium:
Age 01: (1.0, 1.0, 0.0)
Fungsi Keanggotaan Gaussian:
Age 01: (0.9459594689067654, 0.8725252928694238, 0.35286608145884896)

Dapat dilihat bahwa nilai derajat keanggotaan untuk "Age 01" berbeda untuk setiap fungsi keanggotaan.

Perbedaan ini disebabkan oleh bentuk kurva yang berbeda dan parameter yang digunakan untuk mendefinisikannya. Fungsi trapesium memiliki kemiringan yang mendadak pada titik-titik tertentu, fungsi Gaussian memiliki penyebaran yang lebih luas di sekitar nilai tengah atau bahkan jika dibandingkan juga dengan Fungsi Triangular seperti pada pertemuan 4 lalu menghasilkan kemiringan yang tetap.

Hal ini menunjukkan pentingnya memilih fungsi keanggotaan yang sesuai dengan karakteristik data dan kebutuhan aplikasi tertentu.

3. Tampilkan hasil penentuan karyawan tetap atau karyawan kontrak beserta nama karyawannya!
    - Karyawan dengan nama: employee_status_01, memiliki status: a contract employee
    - Karyawan dengan nama: employee_status_02, memiliki status: unknown
    - Karyawan dengan nama: employee_status_03, memiliki status: a contract employee
    - Karyawan dengan nama: employee_status_04, memiliki status: a permanent employee
    - Karyawan dengan nama: employee_status_05, memiliki status: a permanent employee
    - Karyawan dengan nama: employee_status_06, memiliki status: unknown
    - Karyawan dengan nama: employee_status_07, memiliki status: a contract employee
    - Karyawan dengan nama: employee_status_08, memiliki status: a permanent employee
    - Karyawan dengan nama: employee_status_09, memiliki status: a contract employee
    - Karyawan dengan nama: employee_status_10, memiliki status: a permanent employee

    Dalam output di atas, karyawan yang memiliki status "a permanent employee" dianggap sebagai karyawan tetap, sementara karyawan yang memiliki status "a contract employee" dianggap sebagai karyawan kontrak. Karyawan dengan status "unknown" tidak memiliki status yang jelas dan perlu ditindaklanjuti lebih lanjut.