

Bermanfaat
dan Mendunia



Universitas Padjadjaran

Reasoning (Penalaran)

Dr. Intan Nurma Yulita, M.T

unpad.ac.id



Tentang Saya



Dr. Intan Nurma Yulita, M.T

Dosen Departemen Ilmu Komputer Universitas Padjadjaran

Ketua Pusat Riset Kecerdasan Artifisial dan Big Data Universitas Padjadjaran (2021-2023)

Presiden Indonesian Association for Pattern Recognition (INAPR) (2023-sekarang)

Kepala Pusat Inovasi Pengajaran dan Pembelajaran Universitas Padjadjaran (2024-sekarang)

intan.nurma@unpad.ac.id



Pendahuluan

Reasoning atau penalaran adalah kemampuan sistem untuk **berpikir secara logis** dan **menarik kesimpulan** dari informasi yang sudah diketahui.

Tiga pendekatan dasar penalaran dalam AI:

1. logika proposisional
2. logika predikat
3. logika fuzzy



Logika Proposisional (Propositional Logic)

Logika proposisional adalah bentuk logika paling dasar yang digunakan untuk menangani pernyataan sederhana yang hanya bisa bernilai benar (true) atau salah (false).

Logika ini membantu sistem AI menyatakan fakta dan melakukan inferensi dasar dari pernyataan-pernyataan tersebut.



Tata Bahasa pada Propositional Logic

Dalam logika proposisional, kita menggunakan:

- **Proposisi:** Pernyataan seperti "Hari hujan" atau "Lampu menyala".
- **Simbol logika:**
 - AND (\wedge): dan
 - OR (\vee): atau
 - NOT (\neg): tidak
 - \rightarrow : jika-maka
- **Well-formed Formula (WFF):** Formula logika yang ditulis dengan struktur yang sah sesuai aturan logika.



Semantik pada Propositional Logic

- **Nilai kebenaran:** Proposisi hanya bisa bernilai TRUE atau FALSE.
- **Tabel kebenaran:** Alat bantu untuk menentukan hasil logika dari beberapa kombinasi nilai input.

p	q	$\neg p$	$\neg q$	$p \vee q$	$\neg(p \vee q)$	$\neg p \wedge \neg q$
B	B	S	S	B	S	S
B	S	S	B	B	S	S
S	B	B	S	B	S	S
S	S	B	B	S	B	B



Aturan Inferensi untuk Propositional Logic

- **Modus Ponens (MP)**

Jika $A \rightarrow B$ dan A benar, maka B juga benar.

Contoh: Jika hujan, maka tanah basah. Hujan benar \rightarrow tanah basah.

- **Modus Tollens (MT)**

Jika $A \rightarrow B$ dan B salah, maka A juga salah.

Contoh: Jika lampu menyala maka ruangan terang. Ruangan tidak terang \rightarrow lampu tidak menyala.

- **Hypothetical Syllogism (HS)**

Jika $A \rightarrow B$ dan $B \rightarrow C$, maka $A \rightarrow C$.

Contoh: Jika belajar \rightarrow paham, dan paham \rightarrow lulus, maka belajar \rightarrow lulus.

- **Disjunctive Syllogism (DS)**

Jika $A \vee B$ dan A salah, maka B benar.

Contoh: Saya tidur atau saya belajar. Saya tidak tidur \rightarrow saya belajar.

- **Addition**

Jika A benar, maka $A \vee B$ juga benar.

Contoh: Jika “matahari terbit” benar, maka “matahari terbit atau hujan turun” juga benar.

- **Simplification**

Jika $A \wedge B$, maka A juga benar (dan sebaliknya).

Contoh: Saya makan dan minum \rightarrow maka saya makan.

- **Resolution**

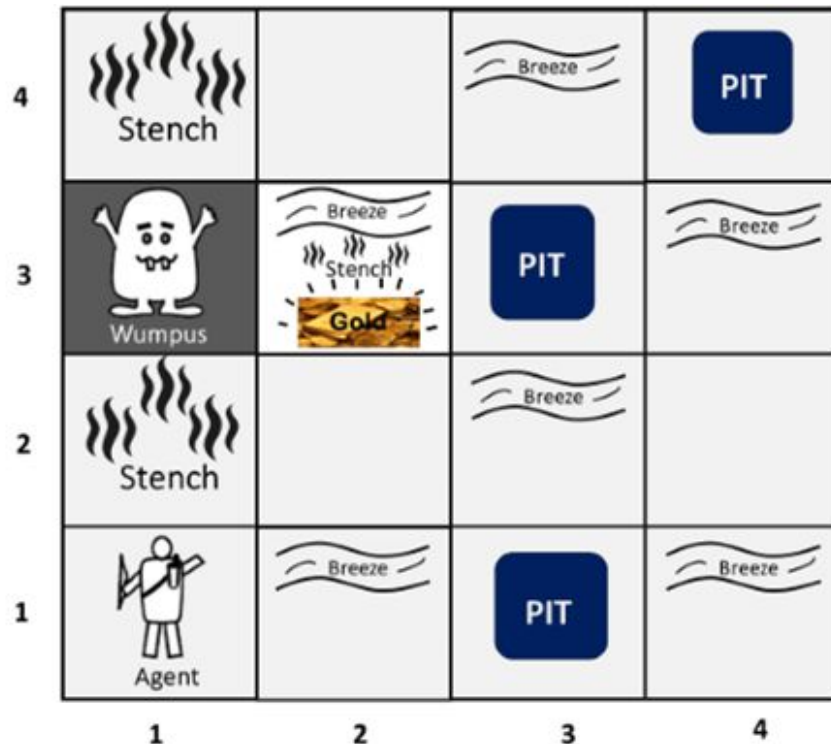
Dari dua kalimat: $A \vee B$ dan $\neg B \vee C$, kita bisa simpulkan $A \vee C$.

Contoh (disederhanakan): Jika saya belajar atau saya tidur, dan jika saya tidak tidur maka saya jalan-jalan \rightarrow maka saya belajar atau saya jalan-jalan.



Masalah Dunia Wumpus

Dunia Wumpus adalah simulasi sederhana yang sering digunakan dalam pembelajaran AI untuk melatih kemampuan penalaran logis (logical reasoning) dalam lingkungan yang tidak sepenuhnya diketahui (partially observable environment).



Lingkungan Wumpus berbentuk **grid 4x4**, seperti papan catur, yang terdiri atas beberapa elemen:

- **Wumpus:** Monster yang membunuh agen jika dimasuki.
- **Pit (lubang):** Agen jatuh dan mati jika masuk ke kotak ini.
- **Gold (emas):** Target yang harus ditemukan dan diambil.
- **Agent:** Tokoh utama yang bergerak dan mengumpulkan informasi.



Masalah Dunia Wumpus

Persepsi Agen: Agen tidak bisa melihat seluruh lingkungan, hanya menerima sinyal dari kotak tempat ia berada:

- **Stench:** Bau menandakan Wumpus ada di kotak tetangga.
- **Breeze:** Angin menandakan lubang di kotak tetangga.
- **Glitter:** Kilau menunjukkan ada emas.
- **Bump:** Menabrak tembok.
- **Scream:** Tanda bahwa Wumpus mati.

Penalaran dalam Dunia Wumpus: Agen harus membuat keputusan berdasarkan informasi terbatas dan menyusun proposisi logika seperti:

- $Breeze(1,1) \rightarrow Pit(1,2) \vee Pit(2,1)$
- $\neg Stench(2,2) \rightarrow \neg Wumpus(2,3) \wedge \neg Wumpus(3,2)$

Dengan logika ini, agen bisa:

- Menghindari kotak berbahaya
- Menentukan kotak yang aman untuk dimasuki
- Merancang jalur menuju emas dan keluar dengan selamat



Perbandingan dengan Algoritma Searching

- **Searching (pencarian):** Pendekatan eksplisit yang mencari jalur dari titik awal ke tujuan dengan menjelajahi kemungkinan jalur (contoh: DFS, BFS, A*).
- **Reasoning (penalaran):** Lebih bersifat deduktif, menggunakan fakta dan aturan untuk menyimpulkan kondisi dunia tanpa menjelajahi semua kemungkinan.

Aspek	Reasoning (Wumpus)	Searching (A*, DFS)
Input	Fakta/persepsi lokal	Peta lengkap atau asumsi lingkungan
Output	Kesimpulan logis	Jalur ke tujuan
Keunggulan	Bisa mengambil keputusan dalam dunia tidak lengkap	Efisien jika peta diketahui penuh
Kelemahan	Perlu logika rumit, tidak cocok untuk peta besar	Gagal jika informasi tidak lengkap



Mengapa Dunia Wumpus Penting dalam AI?

Representasi Pengetahuan: pengetahuan dalam bentuk proposisi logika seperti:

- $Breeze(1,1) \rightarrow Pit(1,2) \vee Pit(2,1)$
- $\neg Stench(2,2) \rightarrow \neg Wumpus(2,3) \wedge \neg Wumpus(3,2)$

Penarikan Kesimpulan (Inference): Agen harus menentukan apakah suatu kotak aman untuk dimasuki atau tidak, berdasarkan persepsi yang terbatas.

Strategi Navigasi: Membuat keputusan berdasarkan risiko. Misalnya, jika terdapat dua kemungkinan bahaya, agen memilih jalur yang belum pasti tetapi berisiko lebih kecil.

Model Dunia Nyata: Banyak aplikasi nyata di AI, seperti robotika atau sistem pakar medis, juga bekerja dalam dunia yang tidak sepenuhnya diketahui.



Knowledge-based System (KBS)

KBS adalah sistem komputer yang meniru cara manusia menggunakan pengetahuan untuk memecahkan masalah. KBS terdiri dari:

- **Knowledge Base:** Menyimpan fakta dan aturan.
- **Inference Engine:** Mesin penalaran untuk menarik kesimpulan dari knowledge base. Contoh nyata: sistem diagnosa medis yang membantu dokter dalam menentukan penyakit berdasarkan gejala.



First-Order Logic (Logika Predikat)

Logika predikat atau **First-Order Logic (FOL)** adalah perluasan dari logika proposisional. Logika ini memungkinkan representasi hubungan antar objek, atribut objek, serta penggunaan kuantor (misalnya "semua", "ada"). Dengan FOL, sistem AI dapat menyatakan fakta dan aturan yang lebih kompleks dan realistis



Tata Bahasa pada First-Order Logic

Dalam FOL, struktur kalimat logika terdiri atas:

- **Predikat:** Menyatakan sifat atau hubungan antar objek. Contoh: **Manusia(Socrates)**.
- **Fungsi dan konstanta:** Menjelaskan entitas tertentu.
- **Variabel:** Simbol yang dapat diganti objek apa saja.
- **Kuantor:**
 - \forall (Universal): berlaku untuk semua objek
 - \exists (Eksistensial): berlaku untuk minimal satu objek

Contoh:

- Semua manusia adalah fana: $\forall x (\text{Manusia}(x) \rightarrow \text{Fana}(x))$
- Socrates adalah manusia: $\text{Manusia}(\text{Socrates})$
- Maka: $\text{Fana}(\text{Socrates})$



Inferensi pada First-Order Logic

Inferensi dalam FOL mencakup proses logis yang lebih kompleks dibandingkan logika proposisional. Beberapa metode:

1. **Generalized Modus Ponens**
2. **Unification**
3. **Forward Chaining**
4. **Backward Chaining**
5. **Resolution**
6. **Refutation**
7. **Konversi ke Bentuk Normal (CNF/INF)**



Generalized Modus Ponens

Menyimpulkan fakta dari aturan umum.

1. **Aturan:** $\forall x (\text{Manusia}(x) \rightarrow \text{Fana}(x))$
2. **Fakta:** $\text{Manusia}(\text{Socrates})$
3. **Inferensi:** $\text{Fana}(\text{Socrates})$

→ Kita punya aturan umum: *Jika seseorang manusia, maka ia fana.*

→ Fakta khusus: *Socrates adalah manusia.*

→ Maka bisa disimpulkan: *Socrates adalah fana.*

Ini seperti menerapkan aturan umum pada kasus nyata.



Unification

Menyamakan dua ekspresi logika agar bisa digunakan bersama.

- Predikat 1: Teman(Ana, x)
- Predikat 2: Teman(Ana, Budi)
- Maka: $x = \text{Budi}$ (unifikasi berhasil)

→ Dua pernyataan logika memiliki bentuk yang hampir sama tapi ada variabelnya.

→ Teman(Ana, x) bisa disamakan (di-unify) dengan Teman(Ana, Budi) jika $x = \text{Budi}$.

✓ Ini penting agar dua fakta bisa digunakan dalam inferensi logis bersama.



Forward Chaining

Penalaran maju dari fakta menuju kesimpulan.

- Fakta: Manusia(Socrates), Manusia(Plato)
- Aturan: $\forall x \text{ (Manusia}(x) \rightarrow \text{Bernapas}(x))$
- Maka sistem menyimpulkan: Bernapas(Socrates), Bernapas(Plato)

→ Kita mulai dari fakta-fakta: *Socrates adalah manusia, Plato adalah manusia.*

→ Dan kita punya aturan: *Semua manusia bernapas.*

→ Maka sistem menyimpulkan: *Socrates dan Plato bernapas.*

✓ Logika ini bekerja maju dari fakta ke kesimpulan.



Backward Chaining

Penalaran mundur dari tujuan ke fakta pendukung.

- Tujuan: apakah Bernapas(Plato)?
 - Cek aturan: $\text{Bernapas}(x) \leftarrow \text{Manusia}(x)$
 - Cek fakta: $\text{Manusia}(\text{Plato}) \rightarrow \text{Maka benar}$
 - Kita ingin tahu apakah *Plato bernapas*.
 - Cari aturan: *Jika seseorang manusia, maka dia bernapas*.
 - Cek: *Plato adalah manusia* → Maka benar, jadi Plato bernapas.
- ✓ Strategi ini berguna jika kita ingin “membuktikan sesuatu” dari tujuan ke fakta.



Resolution

Menggabungkan dua klausa untuk menghasilkan informasi baru.

- Klausa 1: $\neg \text{Manusia}(x) \vee \text{Fana}(x)$
- Klausa 2: $\text{Manusia}(\text{Socrates})$
- Resolusi: $\text{Fana}(\text{Socrates})$

→ Menggunakan dua klausa untuk menyimpulkan sesuatu.

→ Klausa 1: *Jika bukan manusia, maka fana* ($\neg \text{Manusia} \vee \text{Fana}$)

→ Klausa 2: *Socrates adalah manusia* ($\text{Manusia}(\text{Socrates})$)

→ Maka hasilnya: *Socrates fana* ($\text{Fana}(\text{Socrates})$)

✓ Teknik ini penting dalam pembuktian otomatis.



Refutation

Membuktikan suatu klaim salah dengan menghasilkan kontradiksi

- Ingin membuktikan bahwa Fana(Socrates) benar.
 - Asumsikan \neg Fana(Socrates), dan tambahkan ke knowledge base.
 - Jika menghasilkan kontradiksi, maka Fana(Socrates) benar
 - Kita ingin membuktikan bahwa *Socrates adalah fana*.
 - Coba sebaliknya: *Socrates tidak fana*.
 - Gabungkan dengan pengetahuan lain → terjadi kontradiksi.
- ✓ Artinya, kesimpulan awal benar: *Socrates memang fana*.



Konversi ke Bentuk Normal (CNF/INF)

Mengubah ekspresi menjadi bentuk standar untuk keperluan inferensi.

- Kalimat: $\forall x (\neg \text{Manusia}(x) \vee \text{Fana}(x))$
 - Ini sudah dalam bentuk CNF.
 - CNF memudahkan proses resolution dalam sistem inferensi.
 - Untuk digunakan dalam sistem pembuktian otomatis, kalimat logika harus diubah ke bentuk standar (CNF).
 - Contoh: $\forall x (\neg \text{Manusia}(x) \vee \text{Fana}(x))$ sudah berbentuk CNF.
- ✓ CNF memudahkan penerapan teknik resolution.



Logical Programming

Logical programming merupakan teknik pemrograman berbasis logika. Bahasa yang umum digunakan adalah **Prolog**, dengan sintaks berbasis:

- Fakta: `manusia(socrates).`
- Aturan: `fana(X) :- manusia(X).`
- Pertanyaan: `?- fana(socrates).`



Knowledge Engineering (Rekayasa Pengetahuan)

Knowledge engineering adalah proses:

- Mengumpulkan pengetahuan dari pakar
- Merepresentasikan pengetahuan dalam bentuk logika
- Mengelola dan menguji pengetahuan agar bisa digunakan oleh sistem AI



Fuzzy Systems (Sistem Logika Fuzzy)

Logika fuzzy digunakan untuk menangani ketidakpastian dan data yang kabur (vague), seperti “cukup panas”, “sedikit tinggi”. Tidak seperti logika klasik yang kaku (benar atau salah), fuzzy logic memberikan fleksibilitas dalam pengambilan keputusan.



Logic [RUS95]

Jenis logic	Apa yang ada di dunia nyata	Apa yang dipercaya <i>Agent</i> tentang fakta
<i>Propositional logic</i>	fakta	benar/salah /tidak diketahui
<i>First-order logic</i>	fakta, objek, relasi	benar/salah /tidak diketahui
<i>Temporal logic</i>	fakta, objek, relasi, waktu	benar/salah /tidak diketahui
<i>Probability theory</i>	fakta	derajat kepercayaan [0,1]
<i>Fuzzy logic</i>	derajat kebenaran	derajat kepercayaan [0,1]



Peringatan:

1 dari 50 botol ini berisi cairan kimia mematikan yang warna dan rasanya sama dengan air mineral. Anda akan mati seketika jika meminumnya.

Probability

1

Peringatan:

Satu plastik cairan kimia mematikan dicampurkan ke dalam 50 botol ini secara tidak merata. Anda tidak akan mati jika cuma minum satu botol, tetapi anda akan menderita pusing ringan/berat.

Fuzziness

2



Fuzziness & Probability

- Banyak peneliti berbeda pendapat tentang *teori fuzzy* dan **teori probabilitas**
- Sebenarnya, kedua teori tersebut memang sama-sama untuk menangani masalah ketidakpastian
- Tetapi, perbedaannya adalah pada **jenis ketidakpastian** yang ditangani



Fuzzy Systems

- Ide dasar *fuzzy systems* adalah *fuzzy logic* dan *fuzzy sets*
- *Fuzzy logic* sudah lama dipikirkan oleh para filsuf Yunani kuno
- Plato: filsuf pertama yang meletakkan fondasi *fuzzy logic*
- Plato: “Terdapat area ketiga selain Benar dan Salah”
- *Fuzzy logic* menghilang selama 2 milenium
- Muncul kembali pada era 1960-an
- Konsep *fuzzy logic* yang sangat sistematis pertama kali diusulkan oleh Lotfi A. Zadeh, the University of California, Berkeley, Amerika Serikat



Classical Sets

- Teori himpunan klasik: suatu himpunan secara intuitif adalah setiap kumpulan elemen-elemen
- Himpunan klasik dikenal juga sebagai *crisp set*
- *Crisp* = *clear and distinct* [OXF95]
- *Crisp set* : himpunan yang membedakan anggota dan non anggotanya dengan batasan yang jelas



Contoh

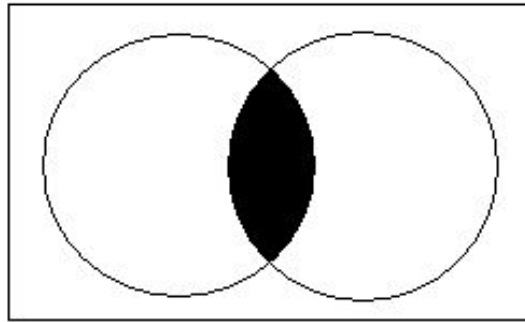
$$A = \{x \mid x \text{ bilangan bulat}, x > 6\}$$

Anggota himpunan A adalah 7, 8, 9, ...

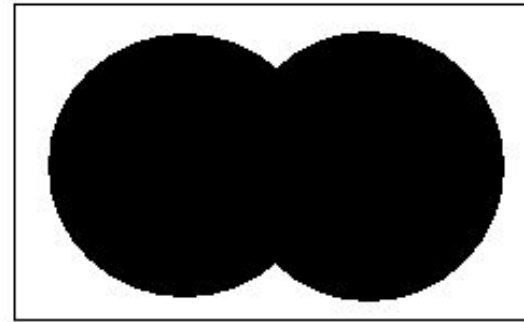
Bukan anggota A adalah 6, 5, 4, ...



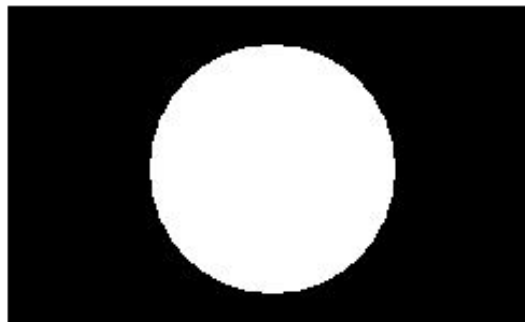
Intersection, union, complement, difference



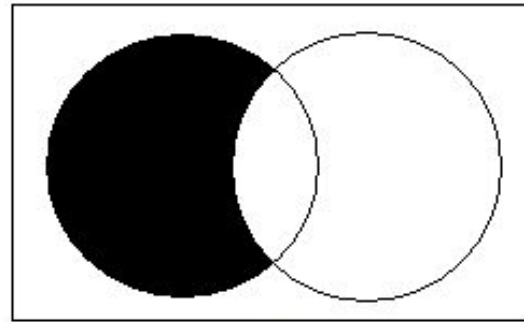
(a)



(b)



(c)



(d)



Fungsi Karakteristik

Fungsi karakteristik dari himpunan A adalah suatu pemetaan

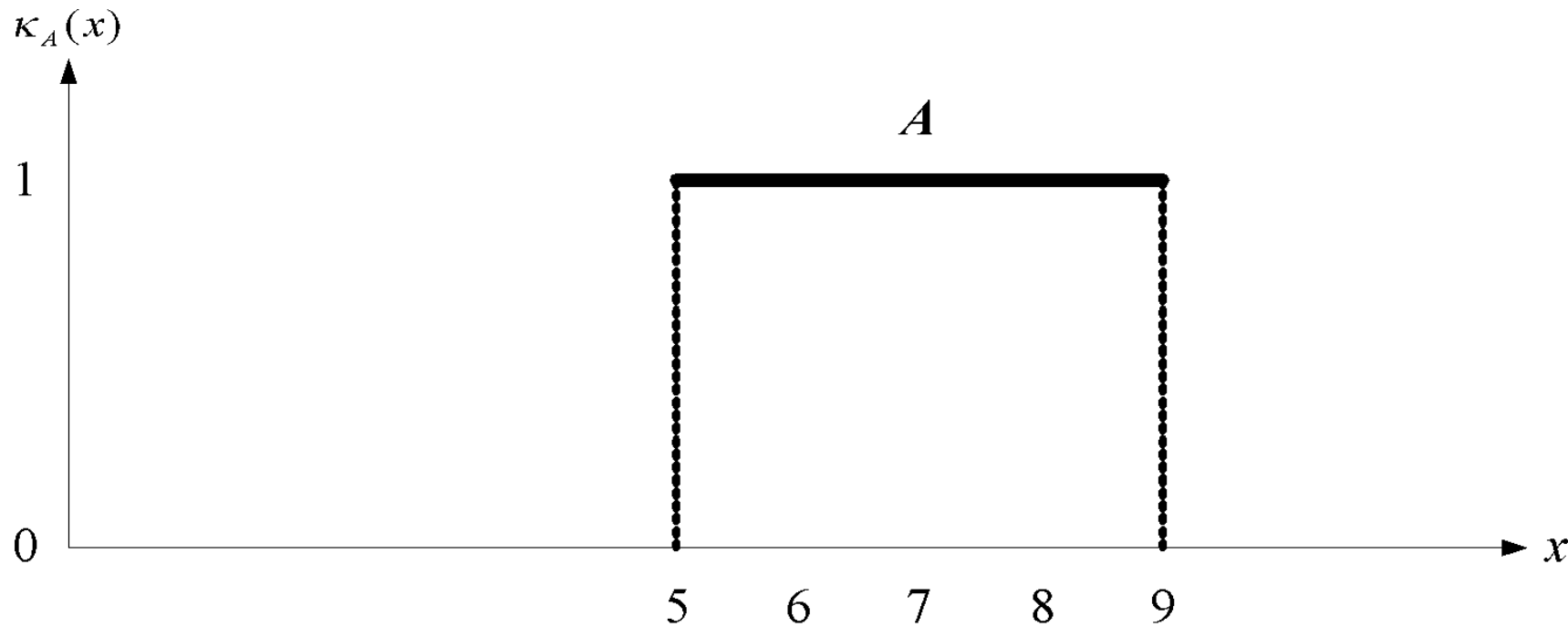
$$\kappa_A : U \rightarrow \{0, 1\}$$

sedemikian hingga, untuk semua x ,

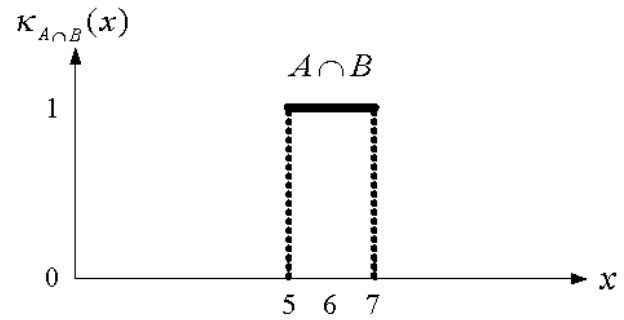
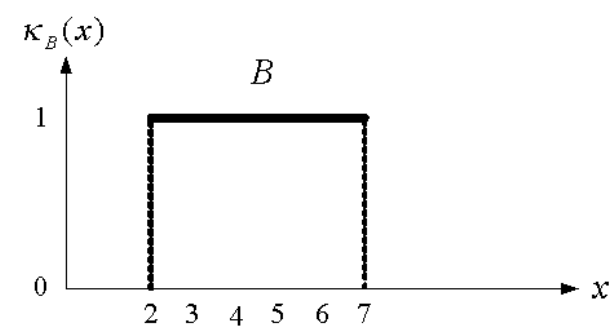
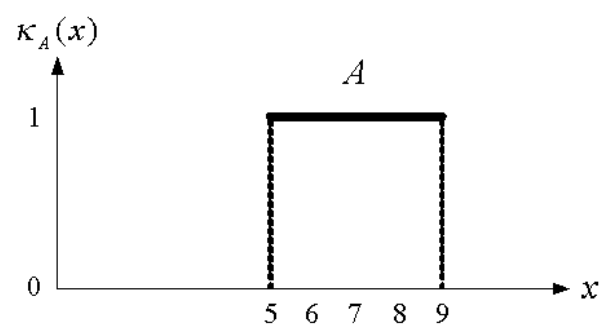
$$\kappa_A(x) = \begin{cases} 1, & \text{jika } x \in A; \\ 0 & \text{untuk kasus lainnya.} \end{cases}$$



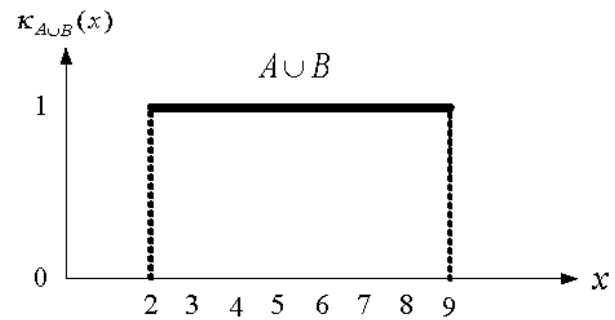
Fungsi Karakteristik *Classical Set*



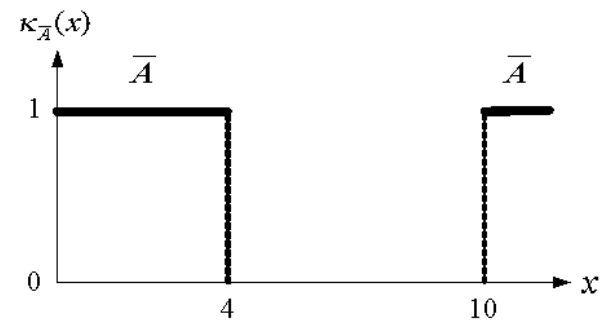
A = himpunan klasik semua bilangan bulat positif lebih dari 4 dan kurang dari 10 atau $\{5, 6, \dots, 9\}$.



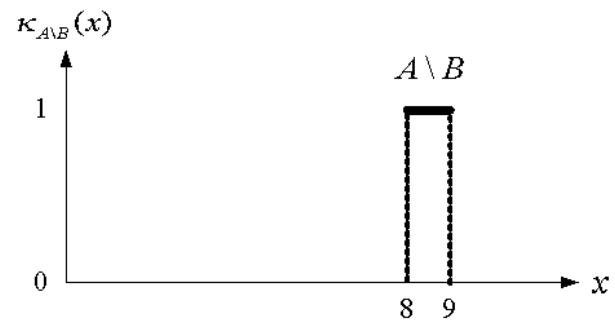
Intersection



Union



Complement



Difference



Kasus 1: Pemberian Beasiswa

Mahasiswa	IPK	Gaji Ortu (Rp/bulan)
A	3,00	10 juta
B	2,99	1 juta

if $IPK \geq 3,00$ and $Gaji \leq 10$ juta
then Dapat Beasiswa



Fuzzy Sets

- Digunakan untuk penalaran yang lebih manusiawi.
- Misalkan U adalah *universe of discourse* (semesta pembicaraan) dan x adalah anggota U .
- Suatu *fuzzy set* A di dalam U didefinisikan sebagai suatu *membership function* atau fungsi keanggotaan, yang memetakan setiap objek di U menjadi suatu nilai real dalam interval $[0, 1]$.
- Nilai-nilai menyatakan derajat keanggotaan x di dalam A .



Suhu ($^{\circ}$ C)	Dingin	Hangat	Panas
5	1	0,1	0
15	0,9	0,8	0
25	0,5	1	0,6
35	0,1	0,6	0,9
45	0	0,2	1



Fuzzy Set

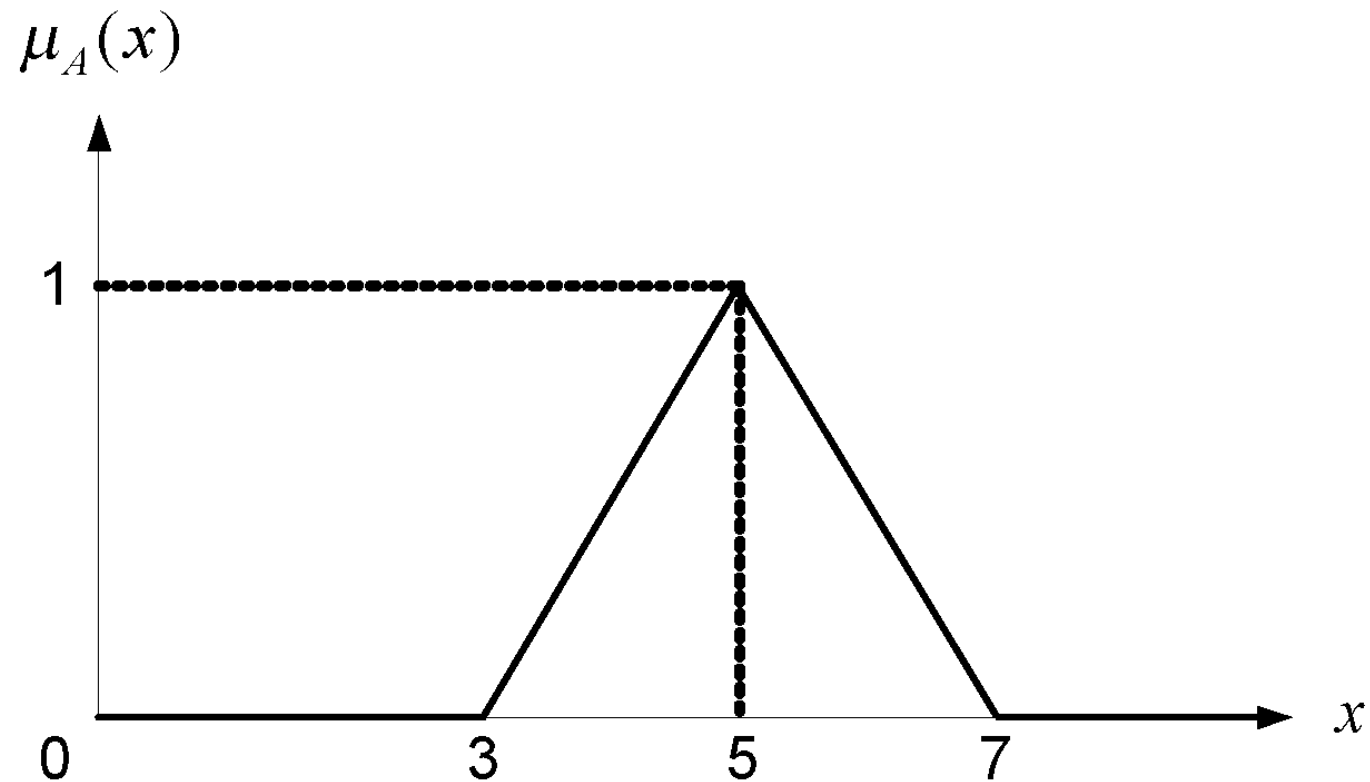
Dingin = {5, 15, 25, 35} dan derajat keanggotaannya dinyatakan oleh $\mu_{Dingin} = \{1; 0,9; 0,5; 0,1\}$

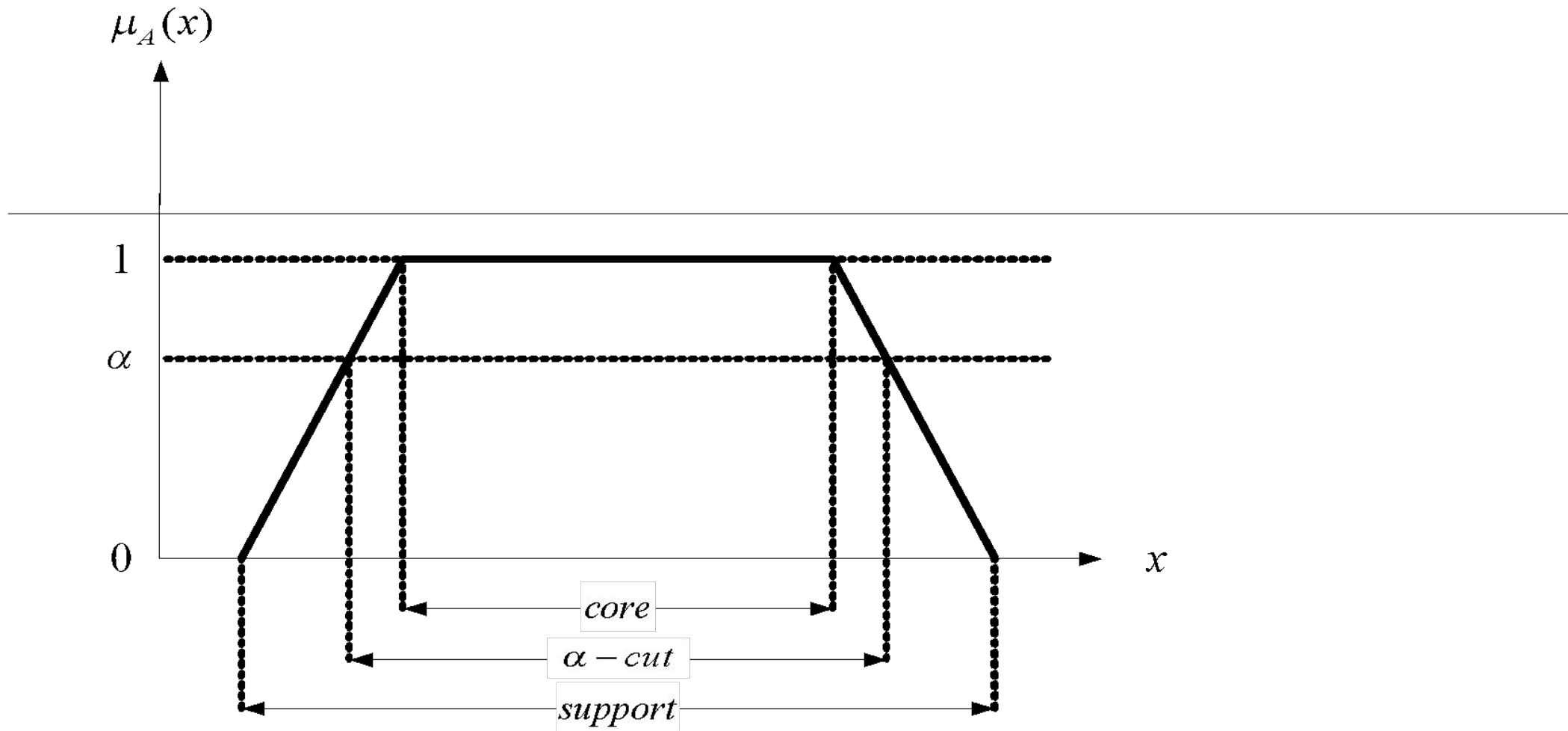
Hangat = {5, 15, 25, 35, 45} dan derajat keanggotaannya dinyatakan oleh $\mu_{Hangat} = \{0,1; 0,8; 1; 0,6; 0,2\}$

Panas = {25, 35, 45} dan derajat keanggotaannya dinyatakan oleh $\mu_{Panas} = \{0,6; 0,9; 1\}$



Graphical Representation



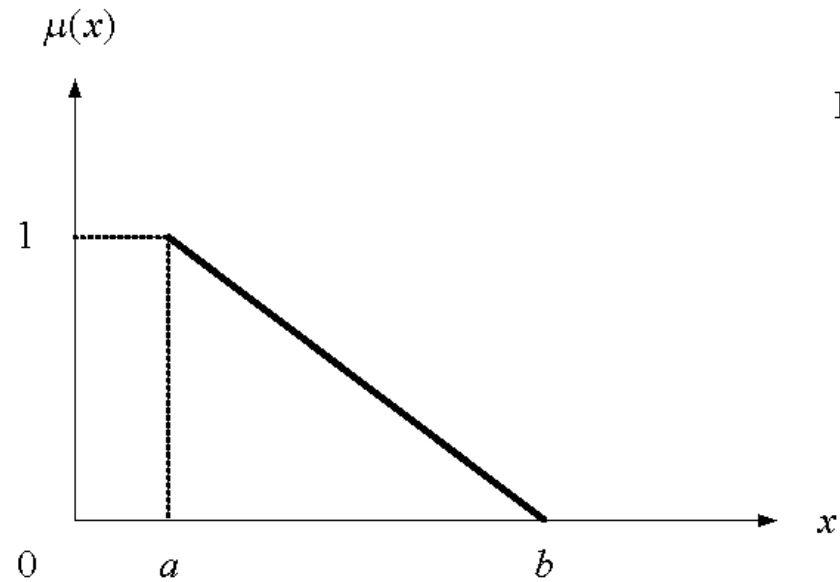
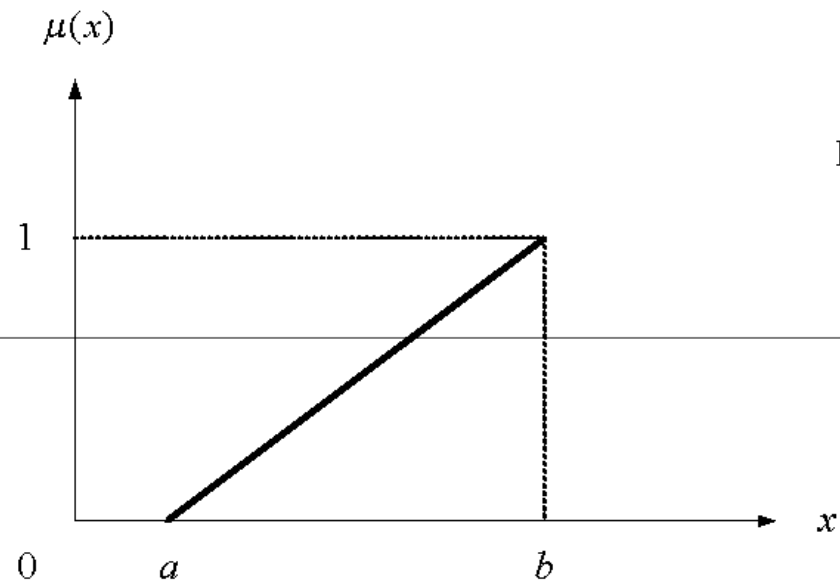


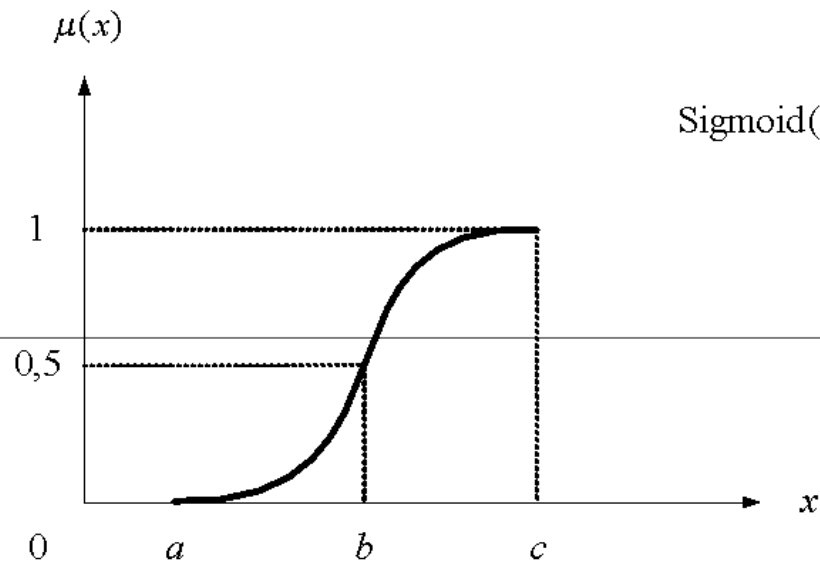
Fungsi keanggotaan μ_A dengan *core*, α - *cut* dan *support*



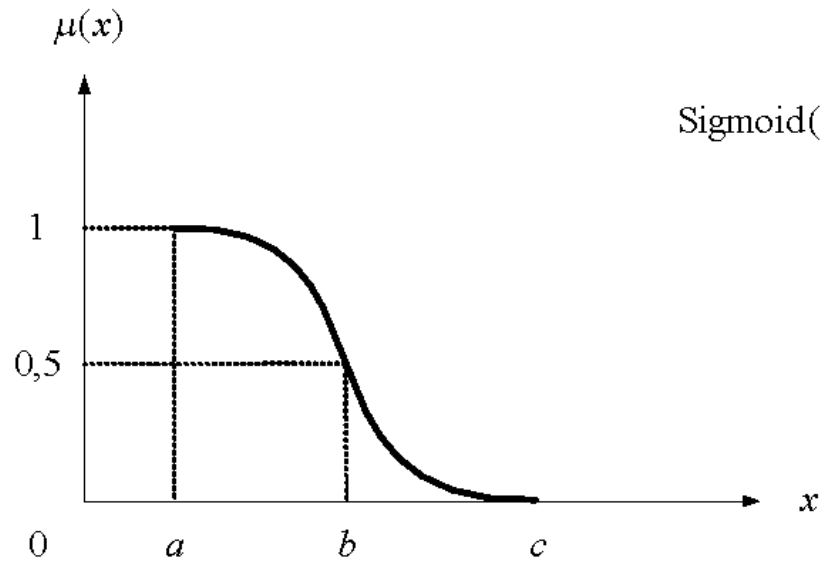
Bentuk Fungsi Keanggotaan

- Fungsi Linier
- Fungsi *Sigmoid*
- Fungsi Segitiga
- Fungsi Trapesium
- Fungsi Berbentuk *Bell*:
 - *Phi*
 - *Beta*
 - *Gauss*

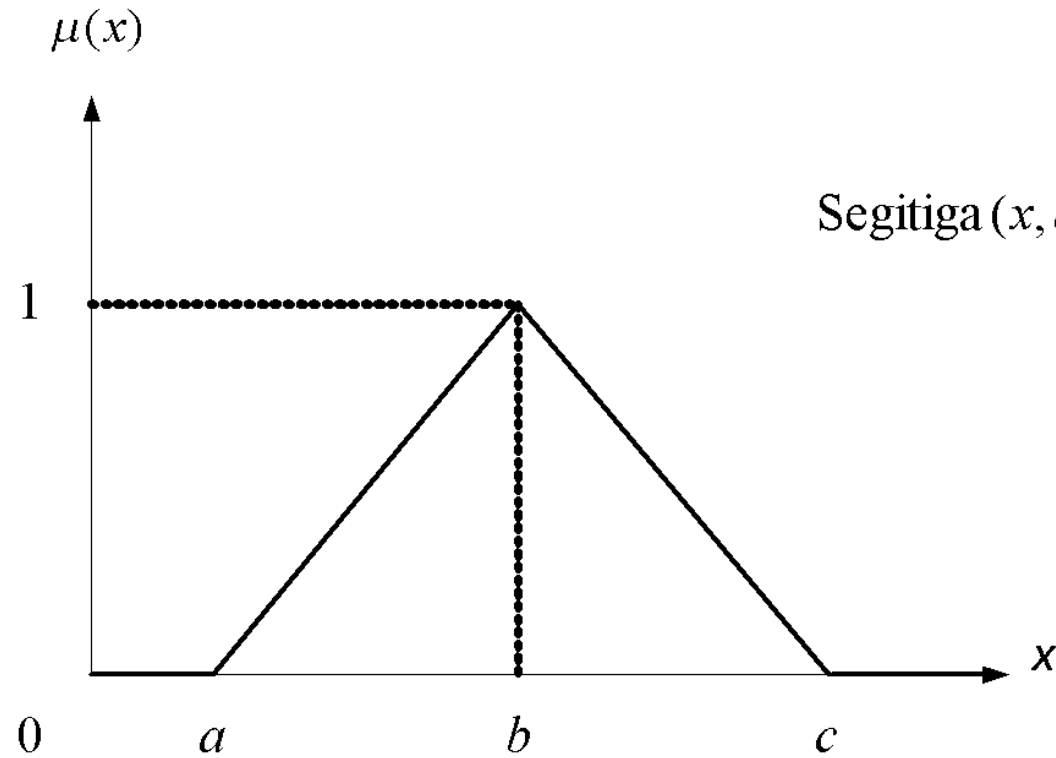




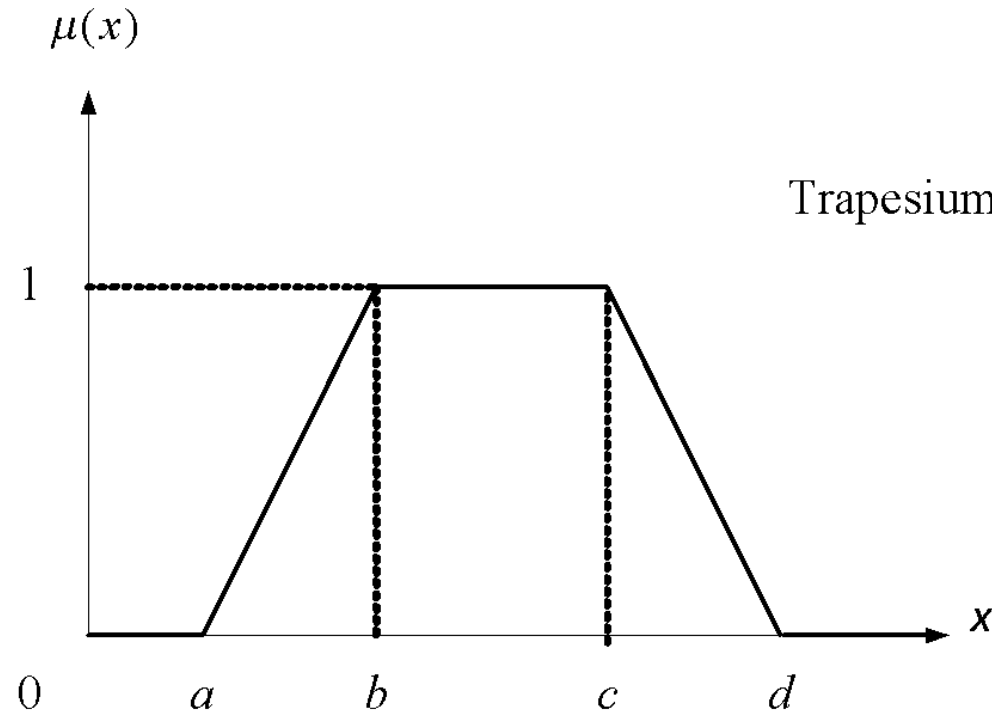
$$\text{Sigmoid}(x, a, b, c) = \begin{cases} 0, & x \leq a \\ 2((x-a)/(c-a))^2, & a < x \leq b \\ 1 - 2((c-x)/(c-a))^2, & b < x < c \\ 1, & c \leq x \end{cases}$$



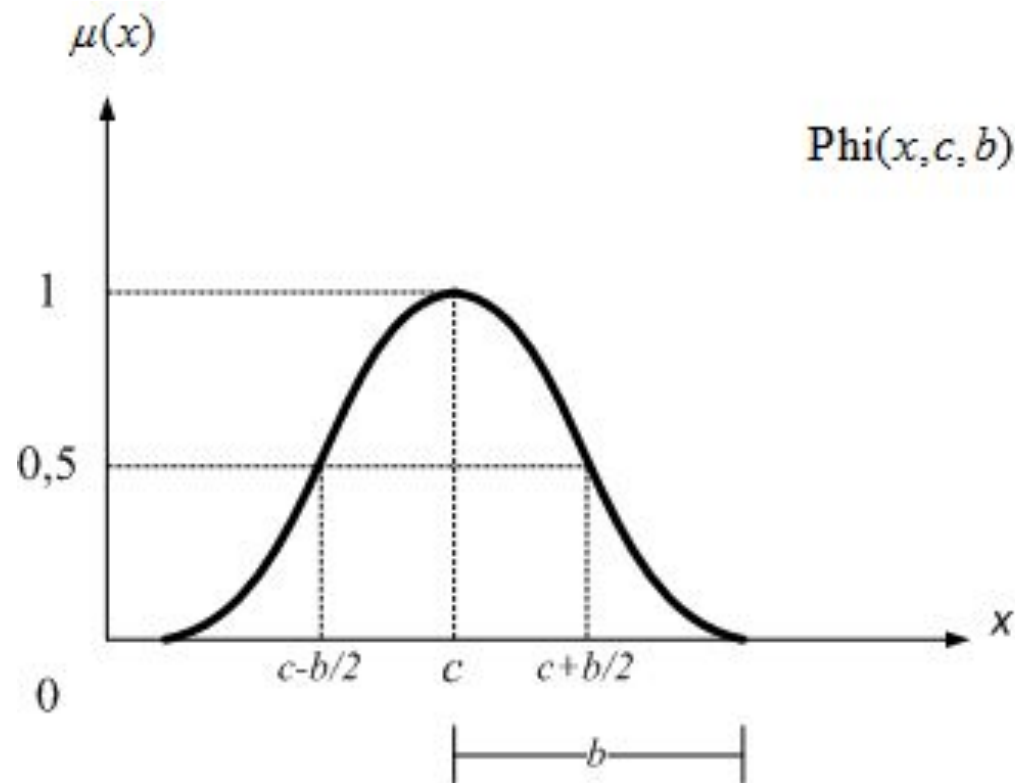
$$\text{Sigmoid}(x, a, b, c) = \begin{cases} 1, & x \leq a \\ 1 - 2((x-a)/(c-a))^2, & a < x \leq b \\ 2((c-x)/(c-a))^2, & b < x < c \\ 0, & x \geq c \end{cases}$$



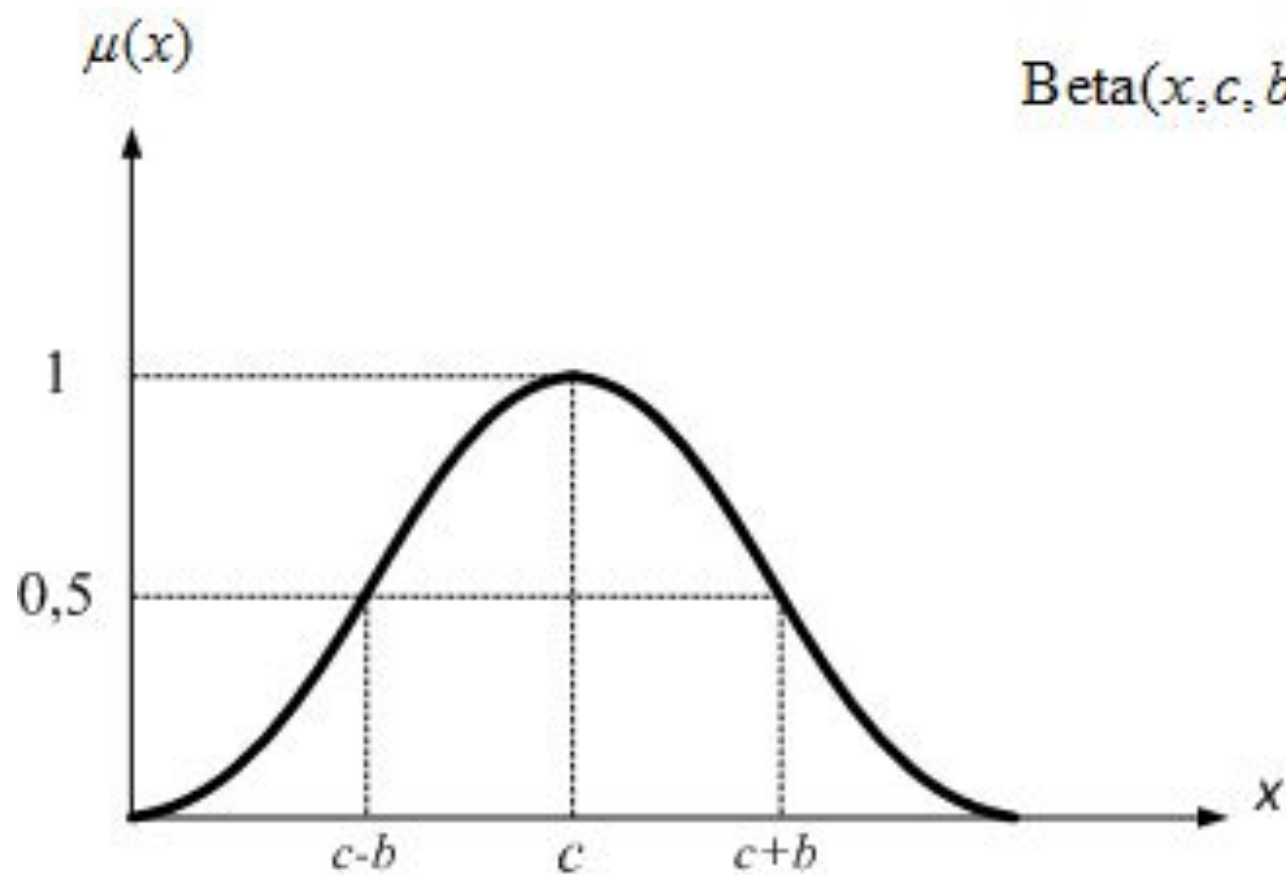
$$\text{Segitiga } (x, a, b, c) = \begin{cases} 0, & x \leq a, x \geq c \\ (x - a) / (b - a), & a < x \leq b \\ -(x - c) / (c - b), & b < x \leq c \end{cases}$$



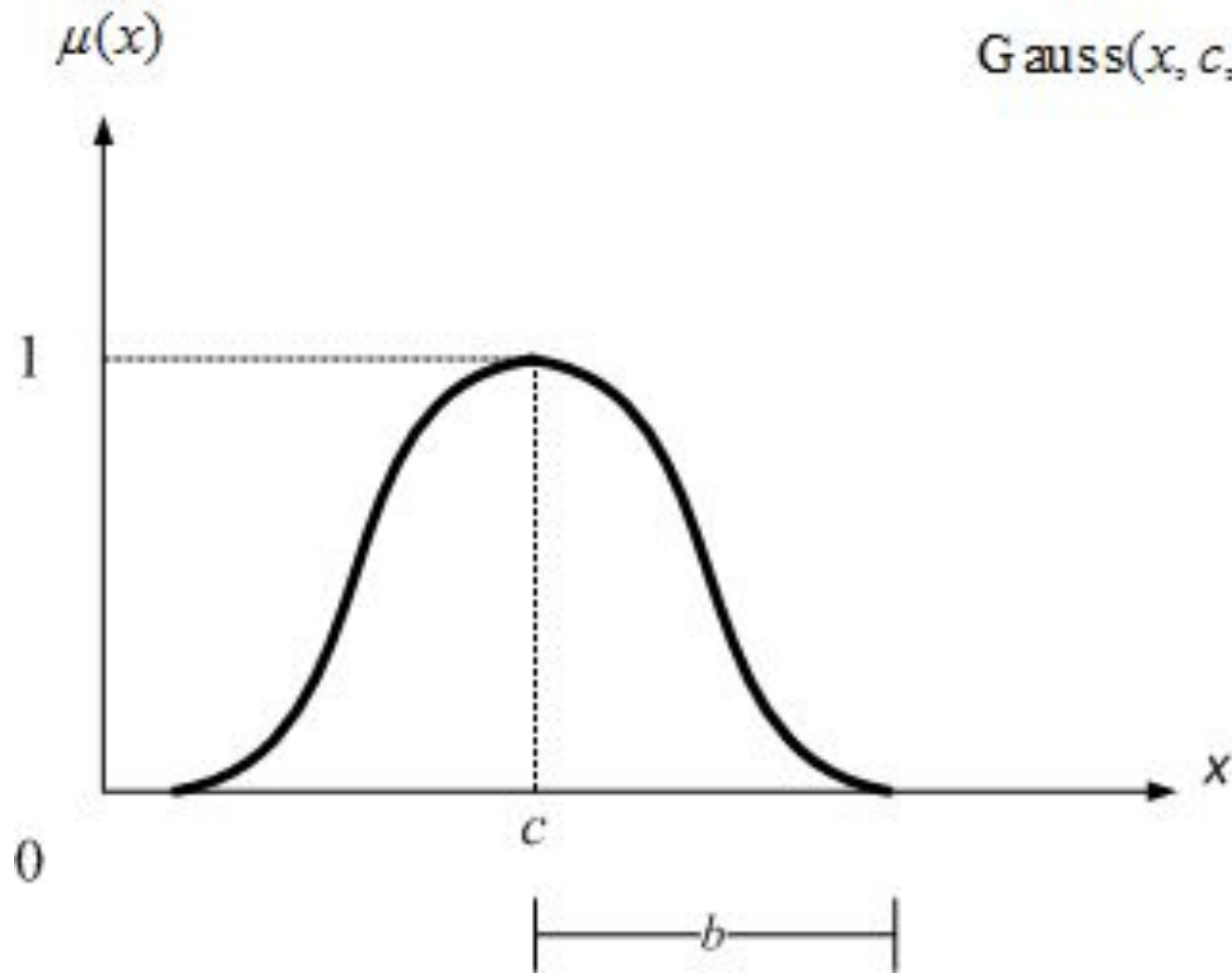
$$\text{Trapezium}(x, a, b, c, d) = \begin{cases} 0, & x \leq a, x \geq d \\ (x-a)/(b-a), & a < x < b \\ 1, & b \leq x \leq c \\ -(x-d)/(d-c), & c < x \leq d \end{cases}$$



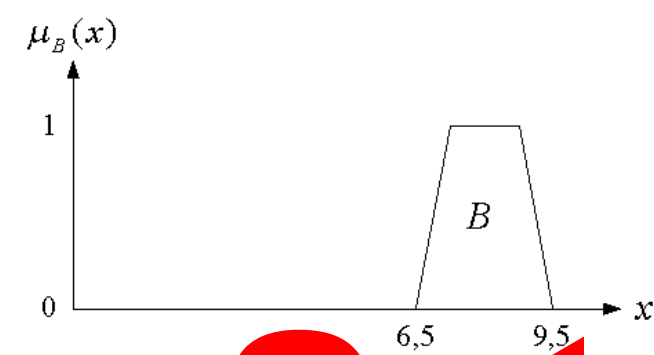
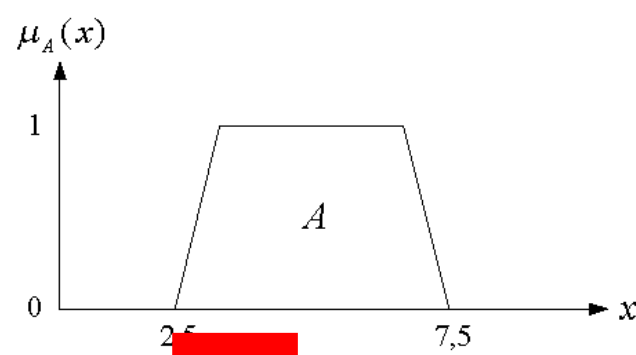
$$\text{Phi}(x, c, b) = \begin{cases} \text{Sigmoid}\left(x, c-b, c-\frac{b}{2}, c\right), & x \leq c \\ 1 - \text{Sigmoid}\left(x, c, c+\frac{b}{2}, c+b\right), & x > c \end{cases}$$



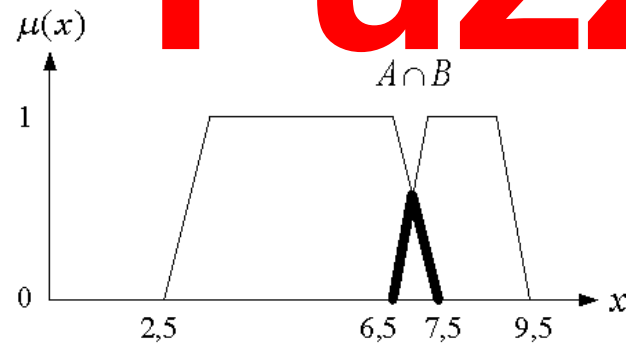
$$\text{Beta}(x, c, b) = \frac{1}{\left(1 + \left|\frac{x-c}{b}\right|^2\right)}$$



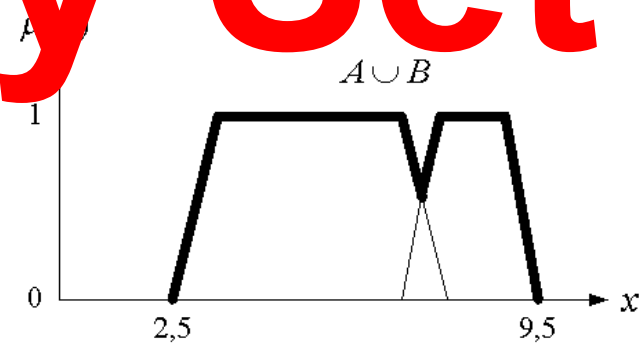
$$\text{Gauss}(x, c, b) = e^{-b(c-x)^2}$$



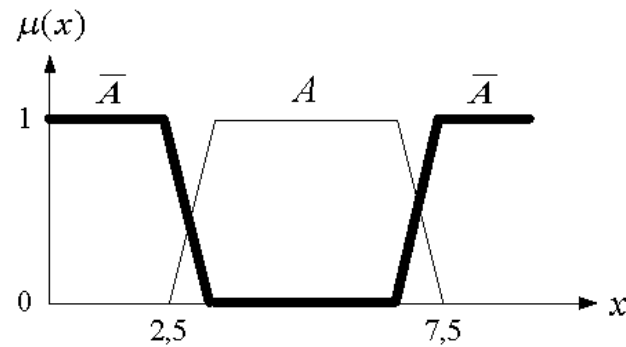
Fuzzy Set



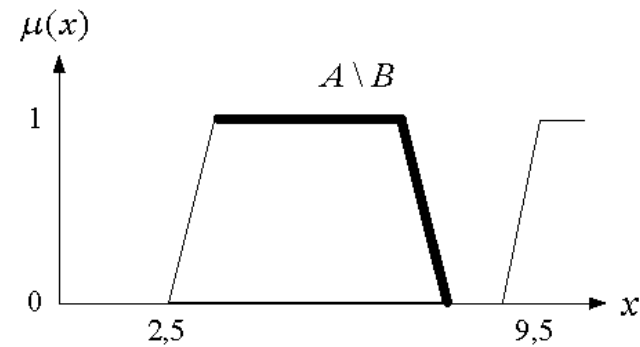
Intersection



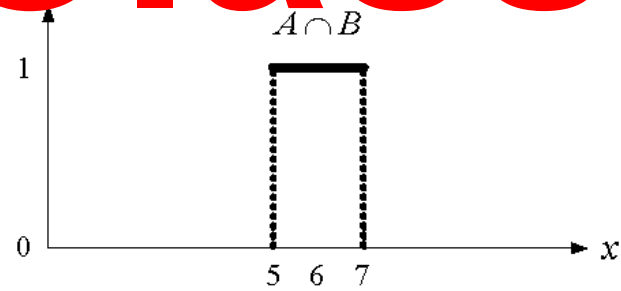
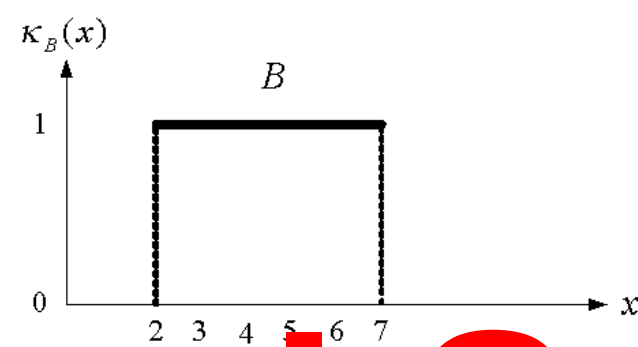
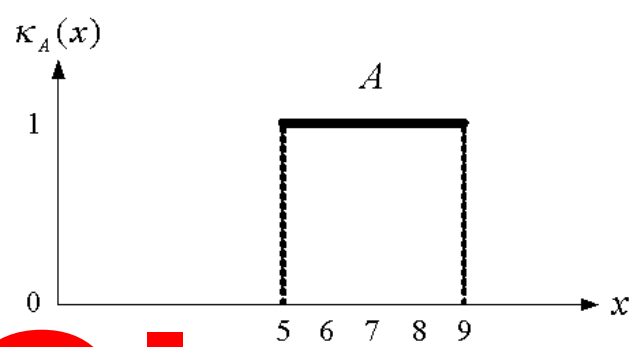
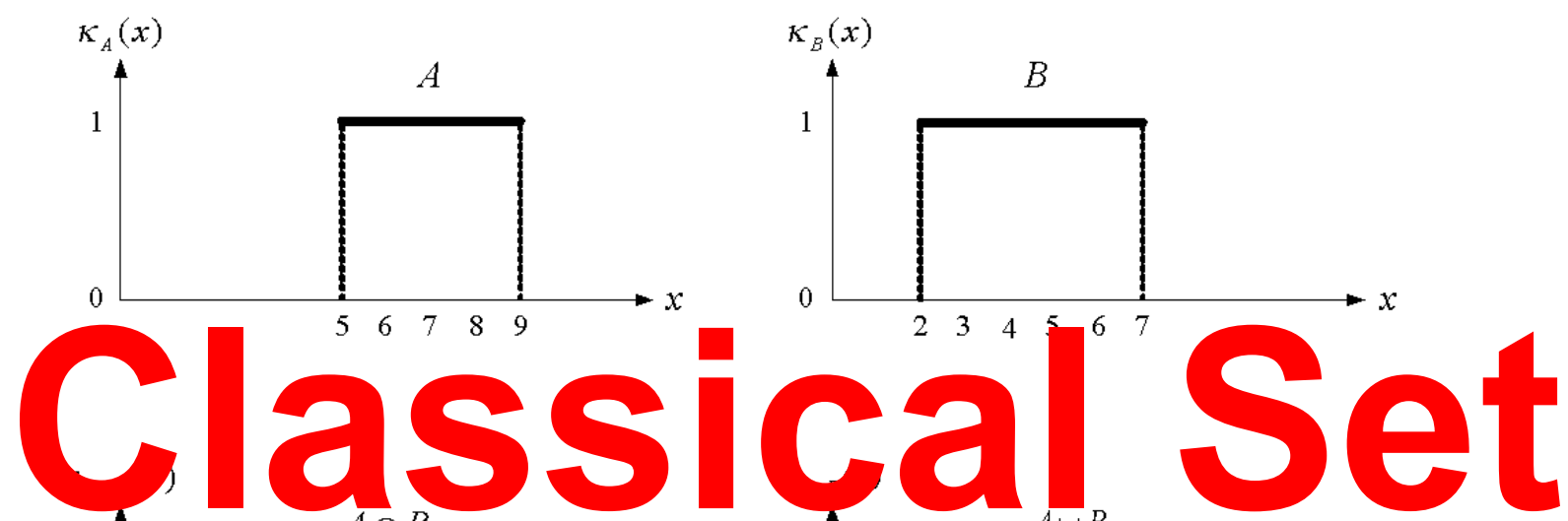
Union



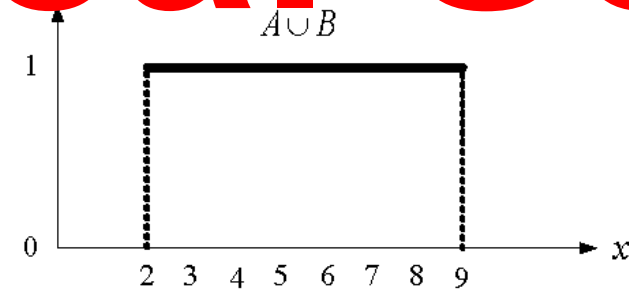
Complement



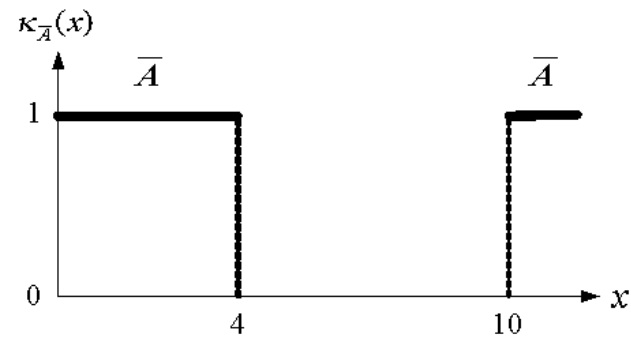
Difference



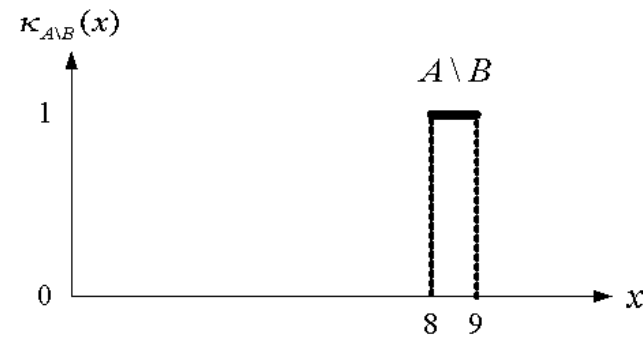
Intersection



Union



Complement



Difference



Logical connectives & Implication

Dalam bahasa manusia, banyak percakapan yang menggunakan kalimat yang tidak pasti kebenarannya

- 'Hampir semua orang suka permen'
- 'Sepertinya dia anak yang pintar'

Misalkan P adalah suatu *fuzzy logic proposition*

Nilai kebenaran P adalah $[0, 1]$.

- Nilai 0 menyatakan bahwa P adalah salah
- Nilai 1 menyatakan bahwa P adalah benar



Logical connectives & Implication

$$T : P \rightarrow [0, 1]$$

T adalah fungsi kebenaran yang memetakan P ke suatu nilai dalam interval $[0, 1]$.



Logical Connectives

Negation

$$T(\neg P) = 1 - T(P)$$

Disjunction

$$T(P \vee Q) = \max\{T(P), T(Q)\}$$

Conjunction

$$T(P \wedge Q) = \min\{T(P), T(Q)\}$$



Approximate Reasoning

A : 'Apakah dia anak yang **pintar**?'

B : '**Sepertinya begitu.**'

A : 'Apakah Indeks Prestasi dan hasil tes psikologinya **bagus**?'

B : 'Ya, keduanya **sangat bagus.**'

A : 'Apakah dia layak mendapatkan beasiswa?'

B : 'Ya, **sepertinya** itu adalah keputusan yang **baik.**'



Approximate Reasoning

P_1 : Sebagian besar mahasiswa suka membaca

P_2 : Dani adalah mahasiswa

P_3 : Sepertinya Dani suka membaca



Reasoning yang Pasti

P_1 : Semua manusia pasti akan mati

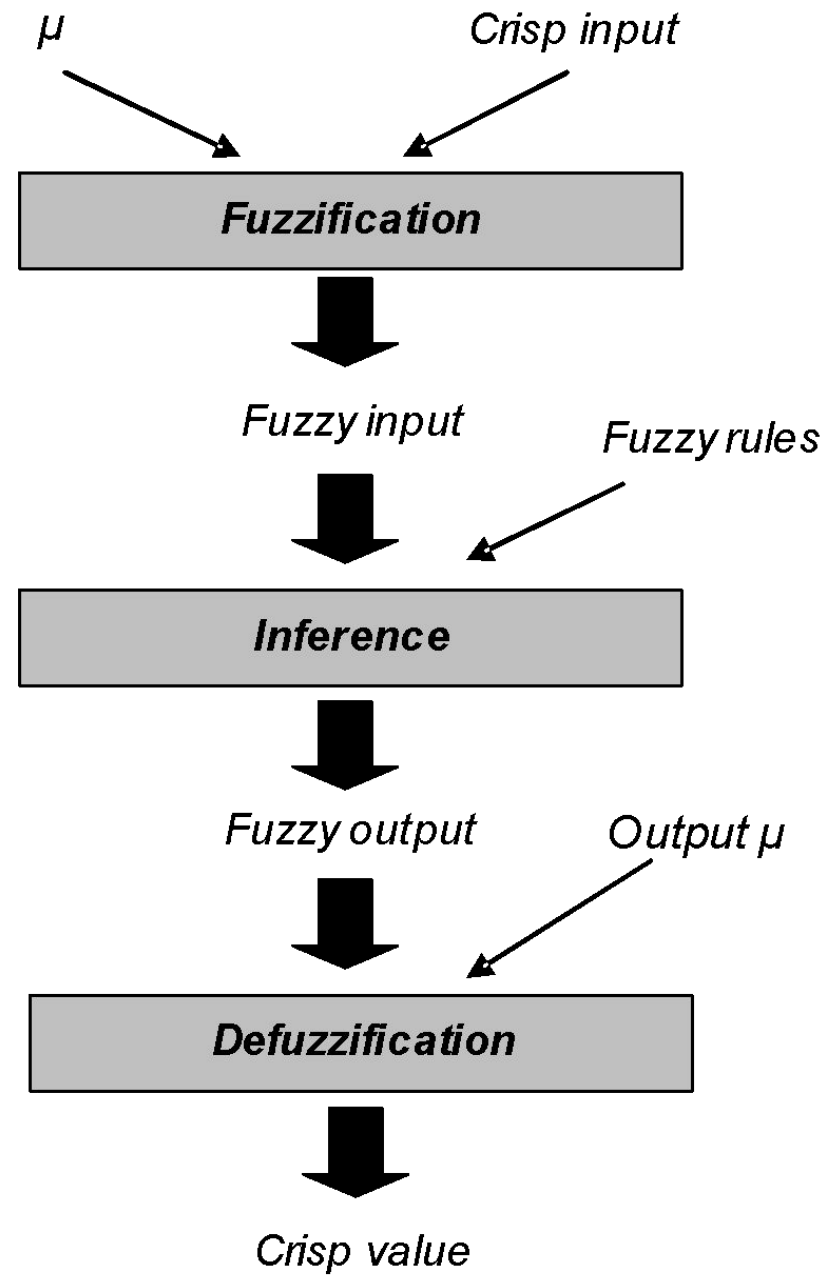
P_2 : Dani adalah manusia

P_3 : Dani pasti akan mati



Fuzzy Inference Systems

- Variabel linguistik adalah suatu interval numerik dan mempunyai nilai-nilai linguistik, yang semantiknya didefinisikan oleh fungsi keanggotaannya.
- Misalnya, *Suhu* adalah suatu variabel linguistik yang bisa didefinisikan pada interval $[-10^{\circ}\text{C}, 40^{\circ}\text{C}]$.
- Variabel tersebut bisa memiliki nilai-nilai linguistik seperti 'Dingin', 'Hangat', 'Panas' yang semantiknya didefinisikan oleh fungsi-fungsi keanggotaan tertentu.





Model Inferensi

- Mamdani ☐ *Intuitive*
- Sugeno ☐ *Control*



Kasus 1: Pemberian Beasiswa

Mahasiswa	IPK	Gaji Ortu (Rp/bulan)
A	3,00	10 juta
B	2,99	1 juta

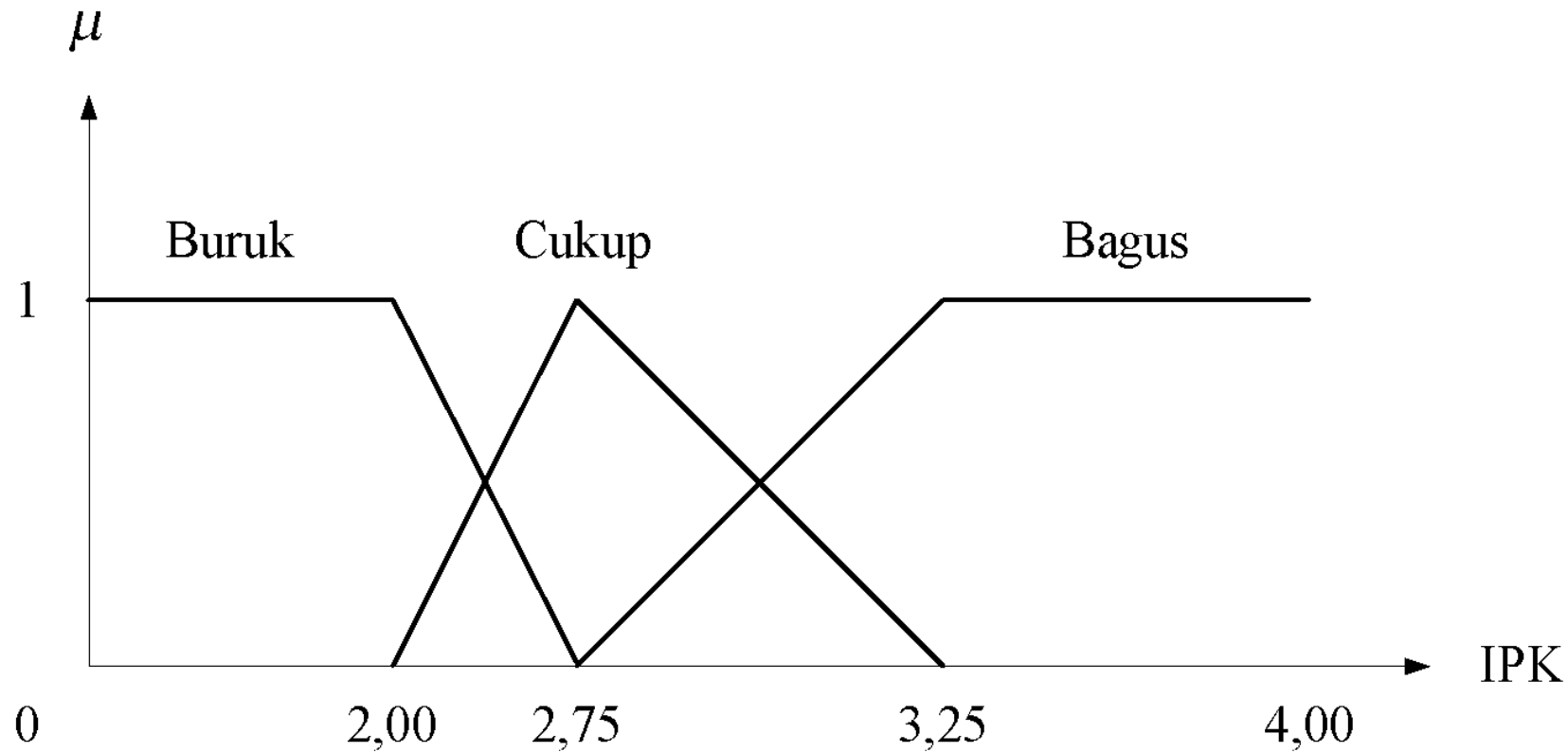


Model Mamdani

- Untuk membangun sistem yang penalarannya menyerupai perasaan manusia
- Perhitungannya kompleks sehingga membutuhkan waktu relatif lama
- Ketelitiannya tinggi



FK untuk IPK

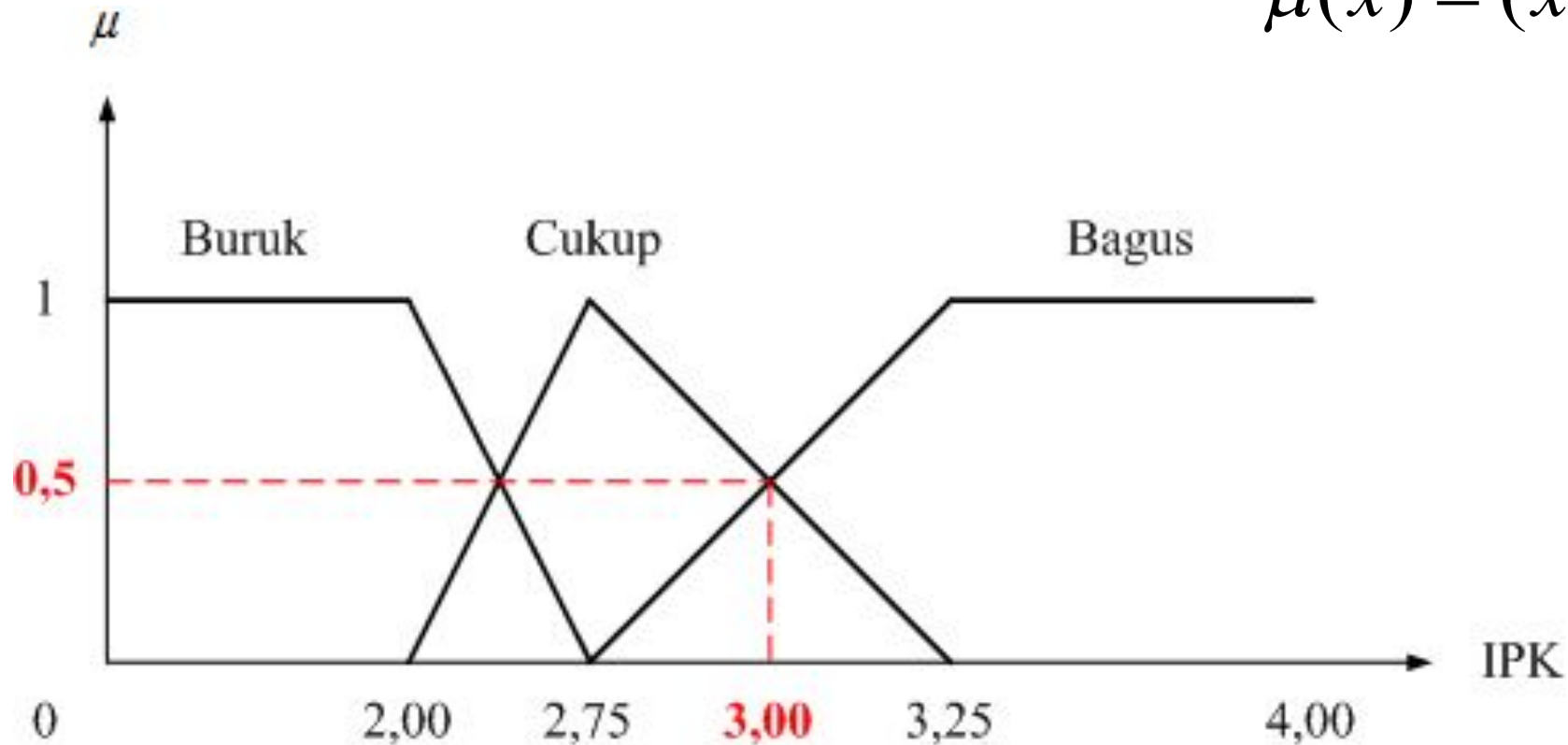




IPK mahasiswa A

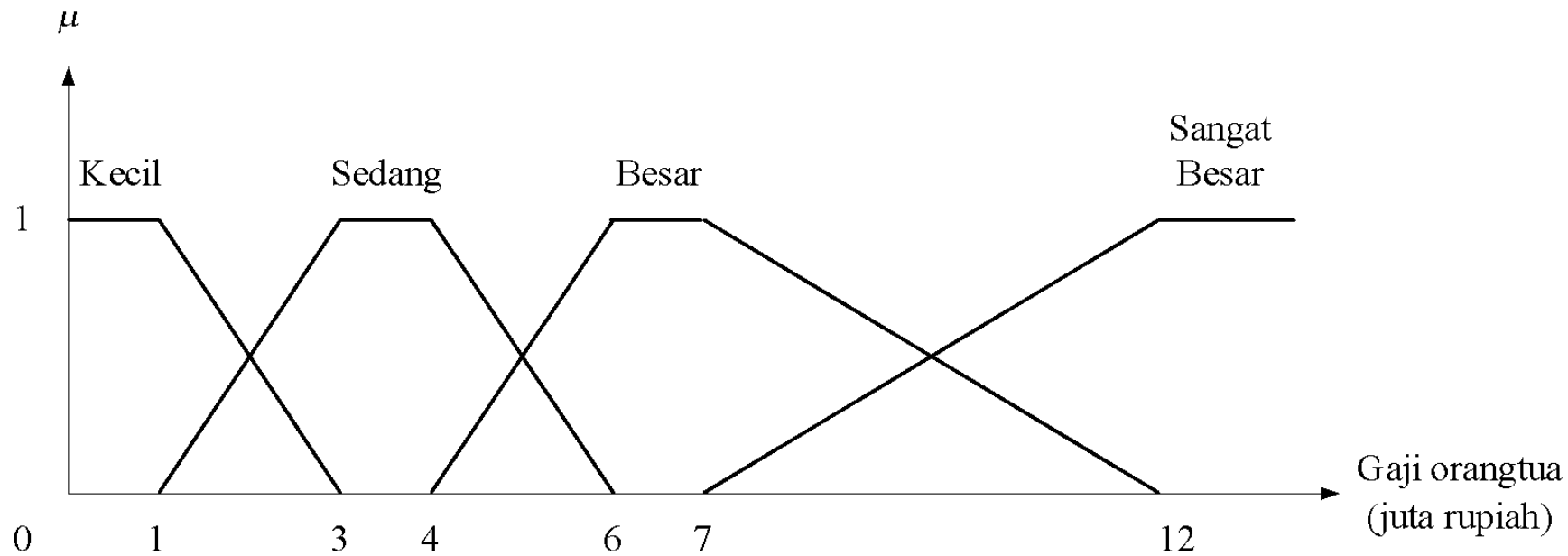
$$\mu(x) = -(x - c) / (c - b)$$

$$\mu(x) = (x - a) / (b - a)$$





FK Gaji Orangtua

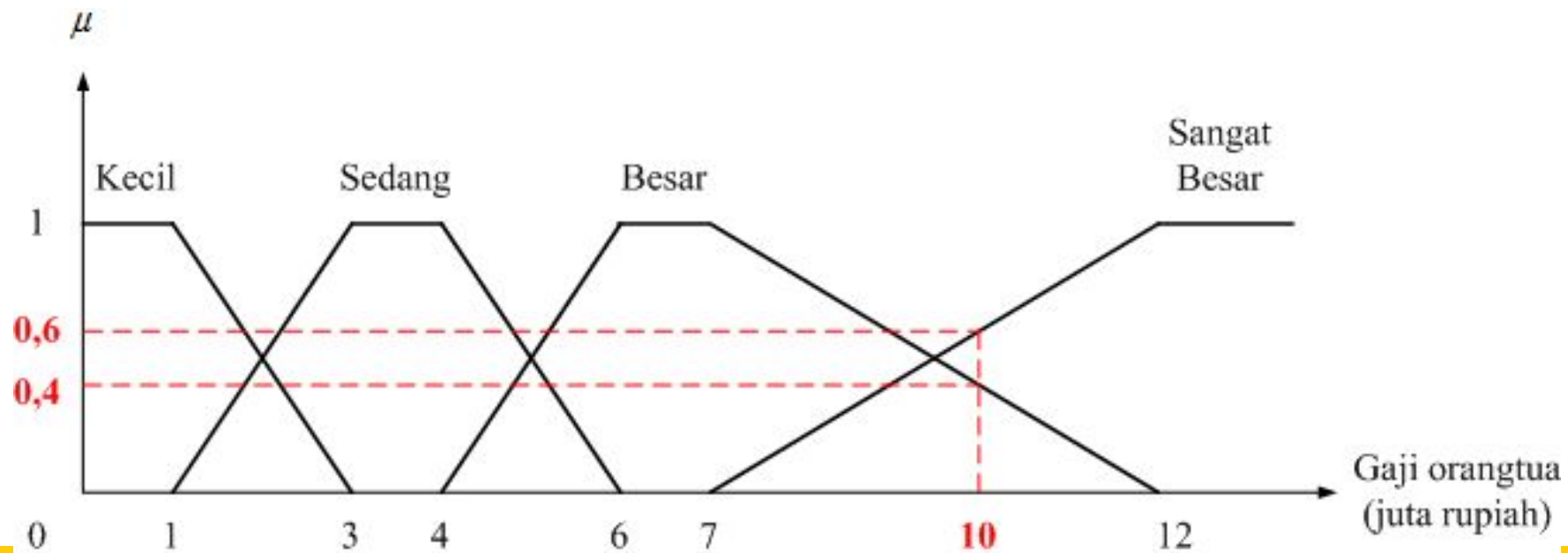




Gaji Ortu mhs A

$$\mu(x) = -(x - d) / (d - c), c < x \leq d$$

$$\mu(x) = (x - a) / (b - a), a < x < b$$





Fuzzification untuk mhs A

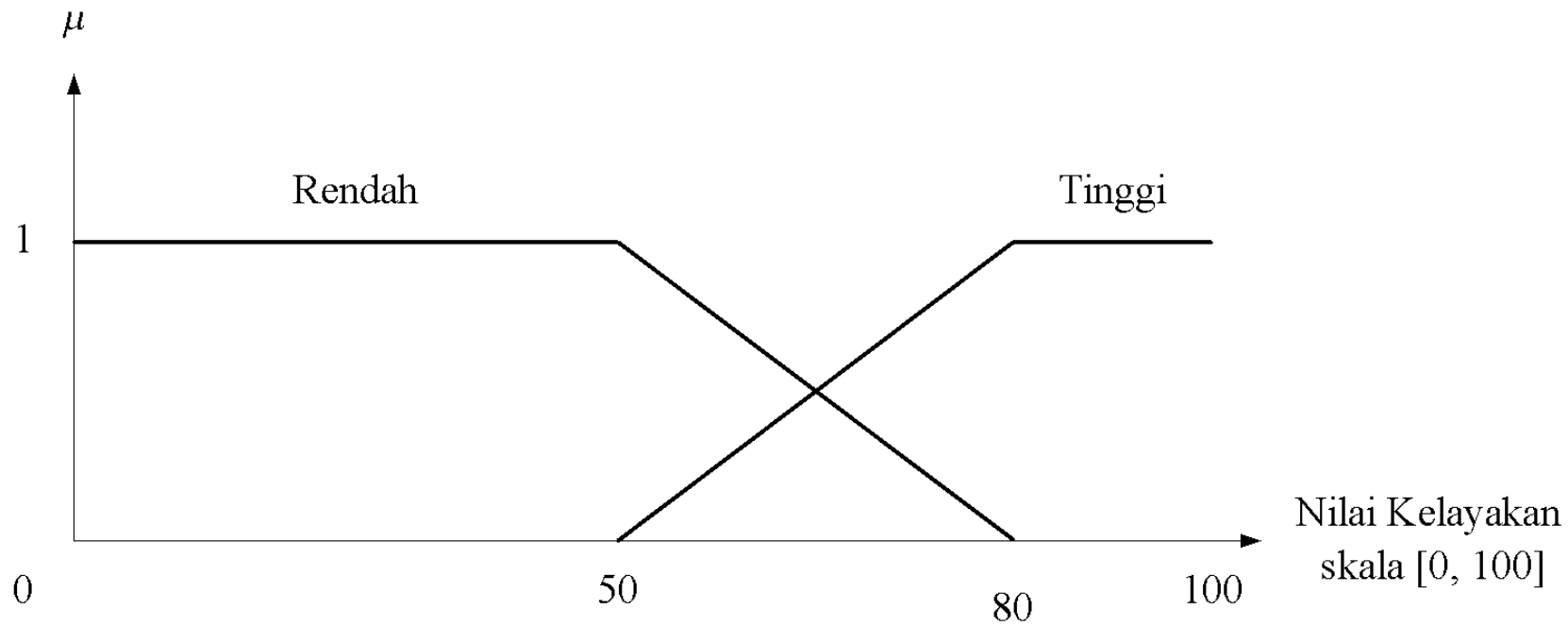
IPK = 3,00
Gaji Orangtua = 10 juta/bulan



IPK = **Cukup** (0,5)
IPK = **Bagus** (0,5)
Gaji Orangtua = **Besar** (0,4)
Gaji Orangtua = **Sangat Besar** (0,6)



Fungsi Keanggotaan Nilai Kelayakan





Aturan Fuzzy untuk Nilai Kelayakan

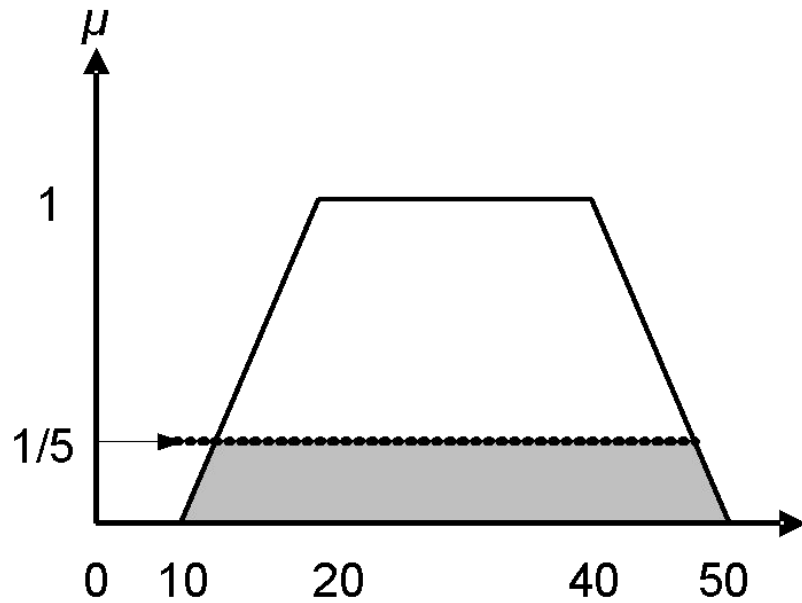
IPK \ Gaji	Kecil	Sedang	Besar	Sangat Besar
Buruk	Rendah	Rendah	Rendah	Rendah
Cukup	Tinggi	Rendah	Rendah	Rendah
Bagus	Tinggi	Tinggi	Tinggi	Rendah



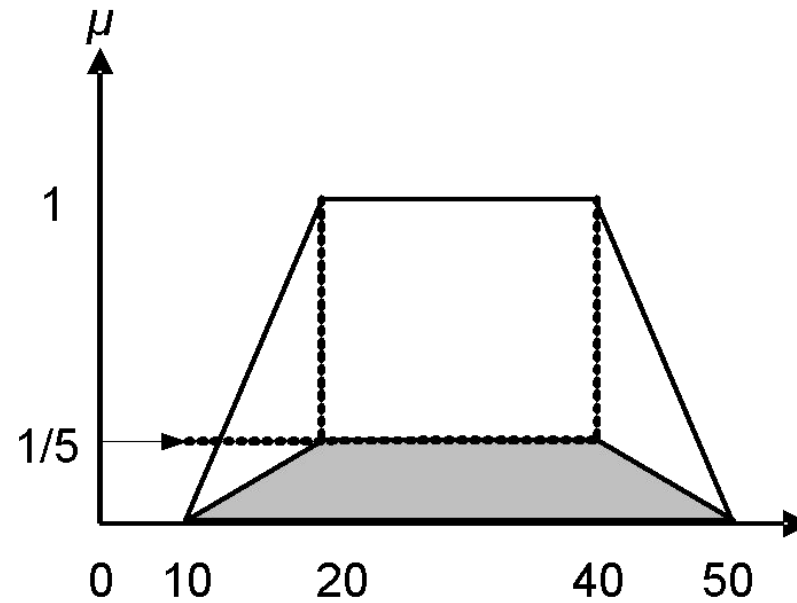
1. IF $IPK = \text{Buruk}$ AND $Gaji = \text{Kecil}$ THEN $NK = \text{Rendah}$
2. IF $IPK = \text{Buruk}$ AND $Gaji = \text{Sedang}$ THEN $NK = \text{Rendah}$
3. IF $IPK = \text{Buruk}$ AND $Gaji = \text{Besar}$ THEN $NK = \text{Rendah}$
4. IF $IPK = \text{Buruk}$ AND $Gaji = \text{Sangat Besar}$ THEN $NK = \text{Rendah}$
5. IF $IPK = \text{Cukup}$ AND $Gaji = \text{Kecil}$ THEN $NK = \text{Tinggi}$
6. IF $IPK = \text{Cukup}$ AND $Gaji = \text{Sedang}$ THEN $NK = \text{Rendah}$
7. IF $IPK = \text{Cukup}$ AND $Gaji = \text{Besar}$ THEN $NK = \text{Rendah}$
8. IF $IPK = \text{Cukup}$ AND $Gaji = \text{Sangat Besar}$ THEN $NK = \text{Rendah}$
9. IF $IPK = \text{Bagus}$ AND $Gaji = \text{Kecil}$ THEN $NK = \text{Tinggi}$
10. IF $IPK = \text{Bagus}$ AND $Gaji = \text{Sedang}$ THEN $NK = \text{Tinggi}$
11. IF $IPK = \text{Bagus}$ AND $Gaji = \text{Besar}$ THEN $NK = \text{Tinggi}$
12. IF $IPK = \text{Bagus}$ AND $Gaji = \text{Sangat Besar}$ THEN $NK = \text{Rendah}$



Inferensi pada model Mamdani: *Clipping* dan *Scaling*



(a) *Clipping*



(b) *Scaling*



Aturan *fuzzy* yang diaplikasikan

7. IF $IPK = \text{Cukup}$ AND $Gaji = \text{Besar}$ THEN $NK = \text{Rendah}$

8. IF $IPK = \text{Cukup}$ AND $Gaji = \text{Sangat Besar}$ THEN $NK = \text{Rendah}$

11. IF $IPK = \text{Bagus}$ AND $Gaji = \text{Besar}$ THEN $NK = \text{Tinggi}$

12. IF $IPK = \text{Bagus}$ AND $Gaji = \text{Sangat Besar}$ THEN $NK = \text{Rendah}$



Nilai *fuzzy* untuk mhs A

IPK = 3,00
Gaji Orangtua = 10 juta/bulan



IPK = **Cukup** (0,5)
IPK = **Bagus** (0,5)
Gaji Orangtua = **Besar** (0,4)
Gaji Orangtua = **Sangat Besar** (0,6)



Conjunction (\wedge) & Disjunction (\vee)

IF $IPK = \text{Cukup}(0,5)$ AND $Gaji = \text{Besar}(0,4)$ THEN $NK = \text{Rendah}(0,4)$

IF $IPK = \text{Cukup}(0,5)$ AND $Gaji = \text{Sangat Besar}(0,6)$ THEN $NK = \text{Rendah}(0,5)$

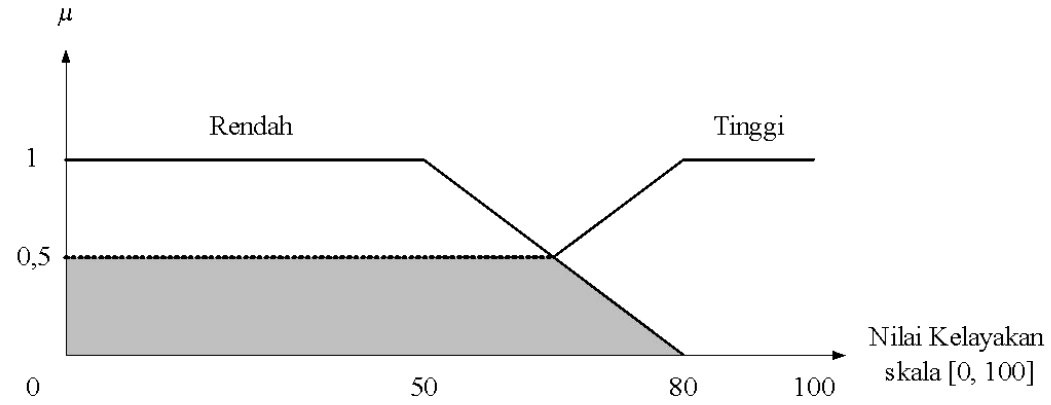
IF $IPK = \text{Bagus}(0,5)$ AND $Gaji = \text{Besar}(0,4)$ THEN $NK = \text{Tinggi}(0,4)$

IF $IPK = \text{Bagus}(0,5)$ AND $Gaji = \text{Sangat Besar}(0,6)$ THEN $NK = \text{Rendah}(0,5)$

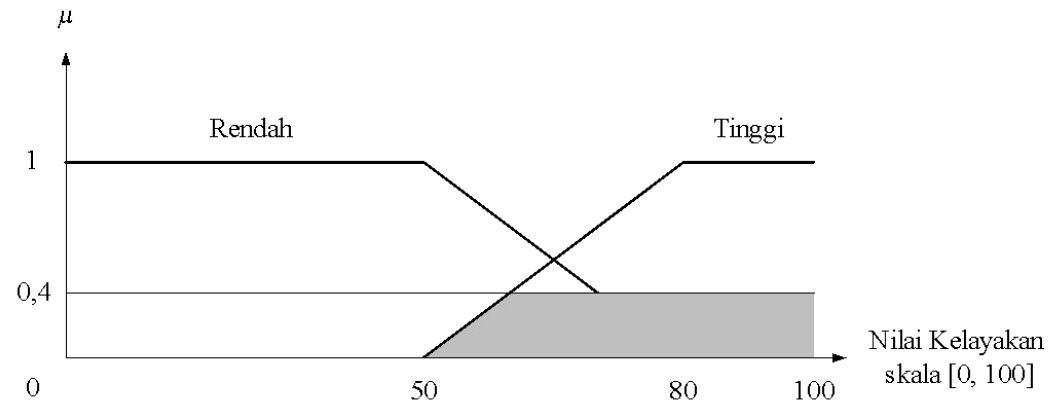


$NK = \text{Rendah}(0,5)$

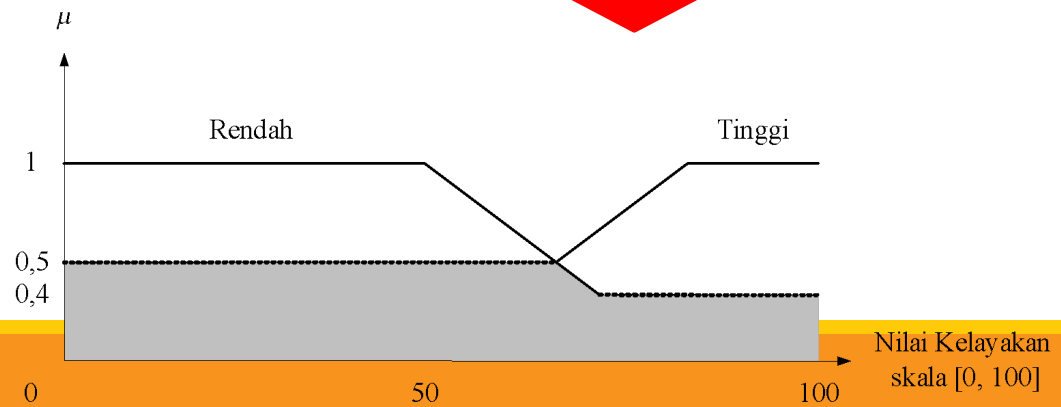
$NK = \text{Tinggi}(0,4)$

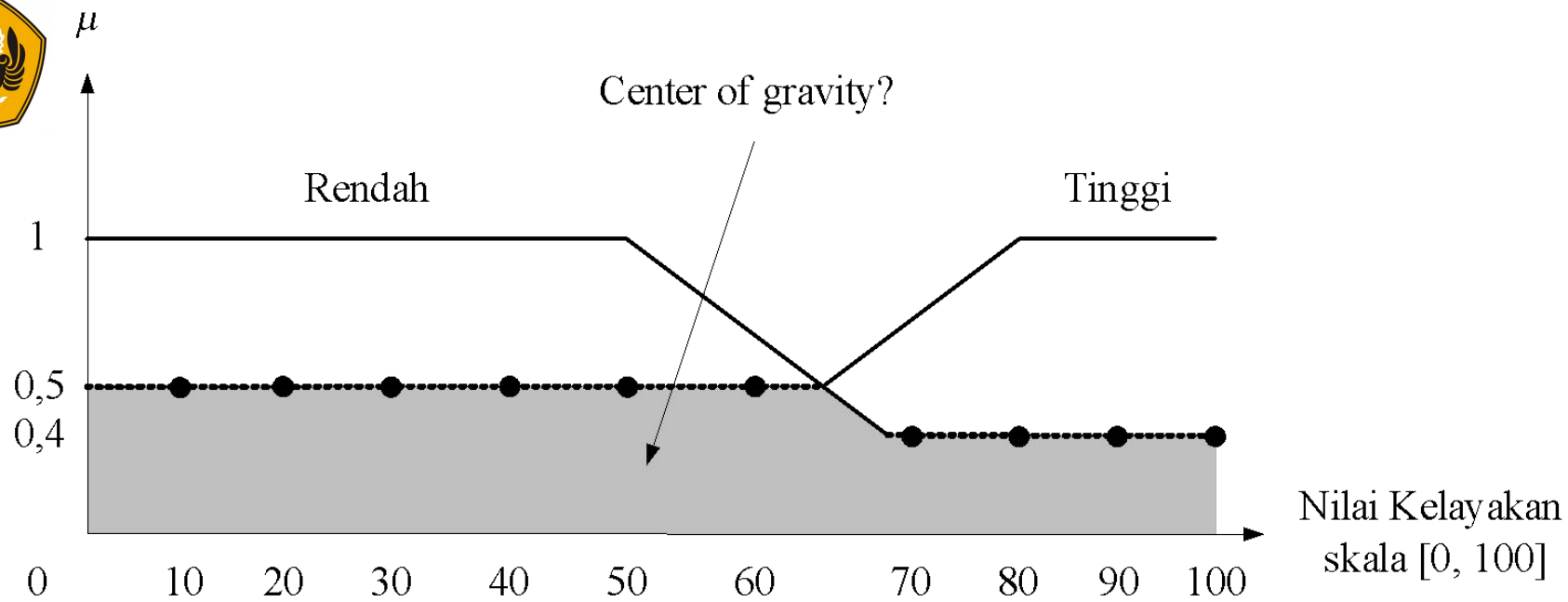


(a)



(b)



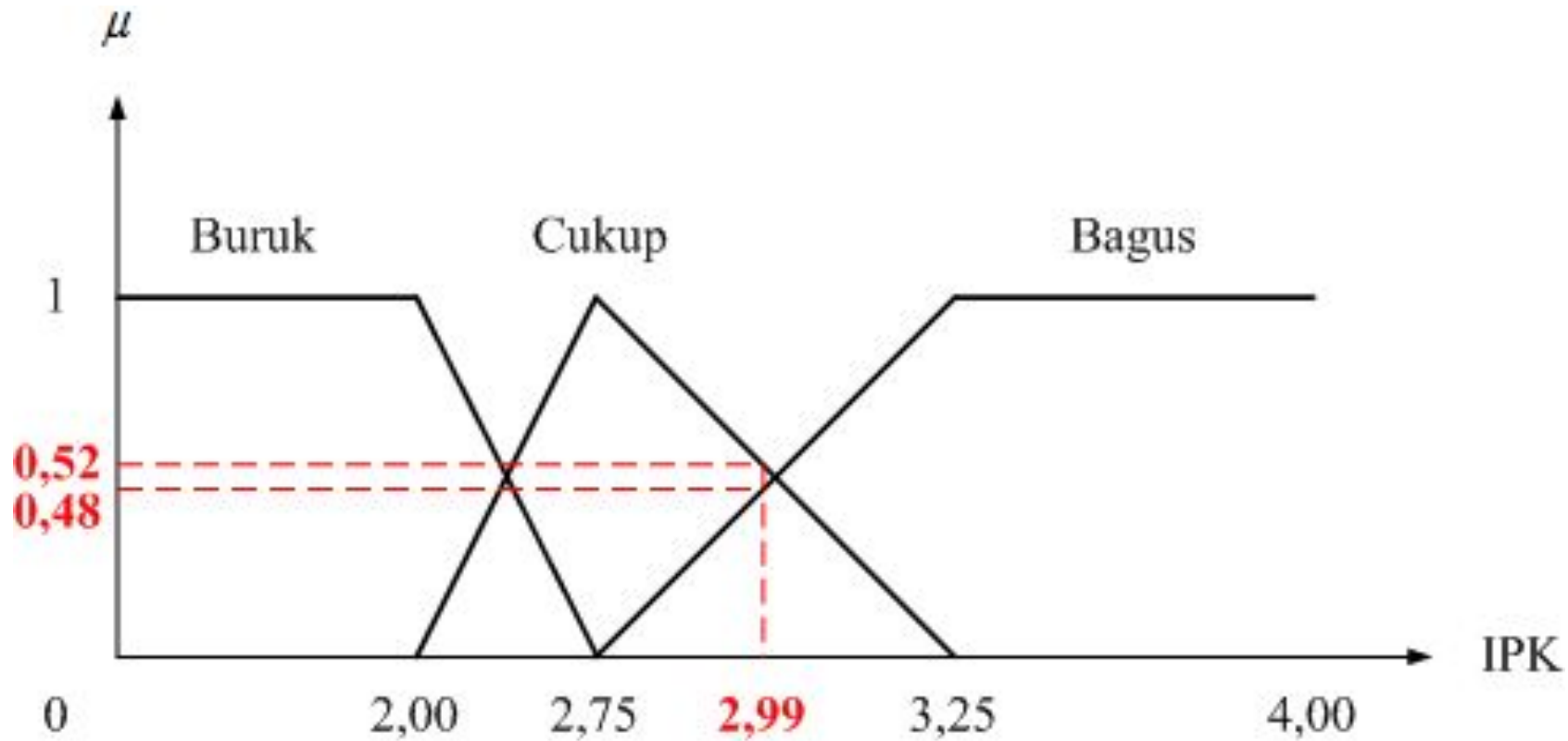


$$y^* = \frac{(10 + 20 + 30 + 40 + 50 + 60)0,5 + (70 + 80 + 90 + 100)0,4}{6(0,5) + 4(0,4)}$$

$$y^* = \frac{105 + 136}{4,6} = 52,39$$

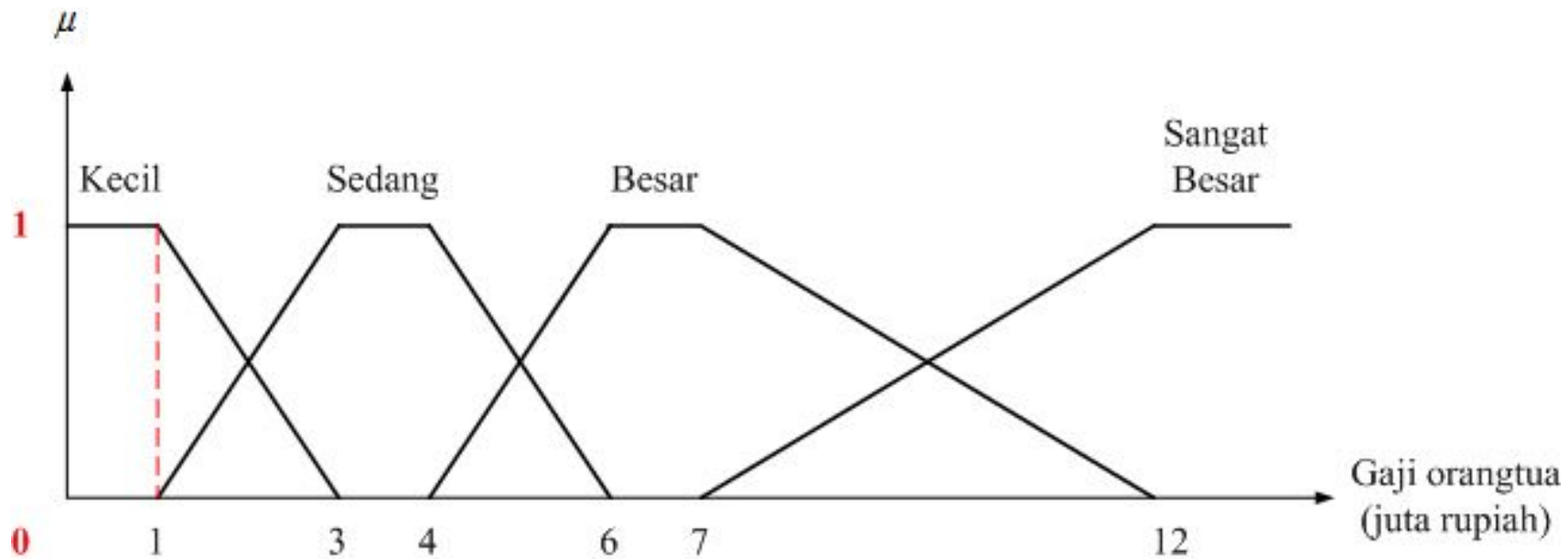


IPK mahasiswa B





Gaji Orangtua mhs B





Conjunction (\wedge) & Disjunction (\vee)

IF $IPK = \text{Cukup}(0,52)$ AND $Gaji = \text{Kecil}(1)$ THEN $NK = \text{Tinggi}(0,52)$

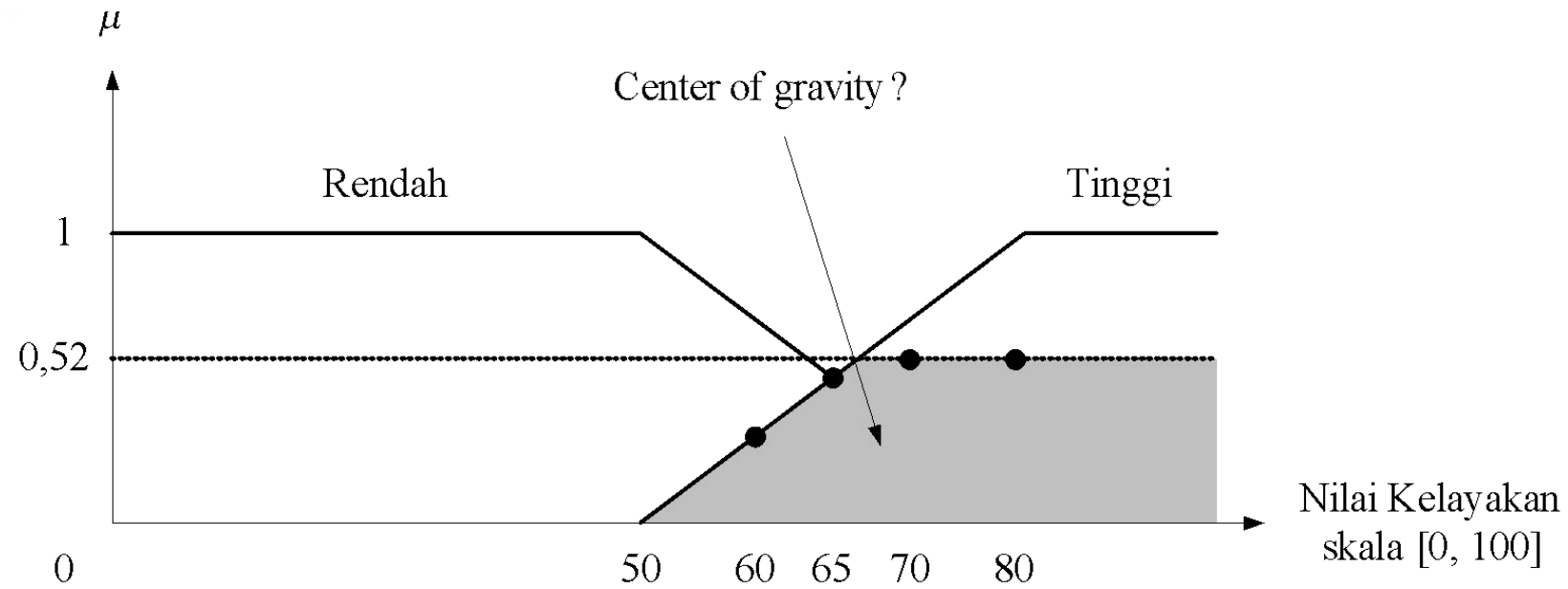
IF $IPK = \text{Cukup}(0,52)$ AND $Gaji = \text{Sedang}(0)$ THEN $NK = \text{Rendah}(0)$

IF $IPK = \text{Besar}(0,48)$ AND $Gaji = \text{Kecil}(1)$ THEN $NK = \text{Tinggi}(0,48)$

IF $IPK = \text{Besar}(0,48)$ AND $Gaji = \text{Sedang}(0)$ THEN $NK = \text{Tinggi}(0)$



$NK = \text{Rendah}(0)$
 $NK = \text{Tinggi}(0,52)$



$$y^* = \frac{60(1/3) + 65(1/2) + (70 + 80)(0,52)}{(1/3) + (1/2) + (0,52)2}$$

$$y^* = \frac{20 + 32,5 + 78}{2,87334} = 69,66$$

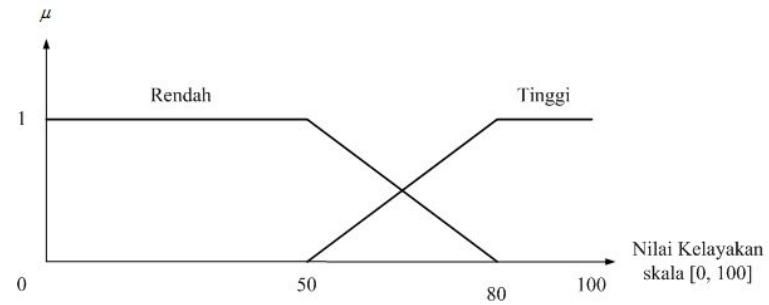
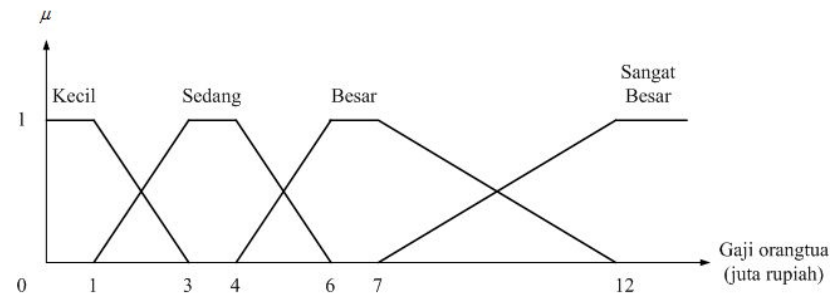
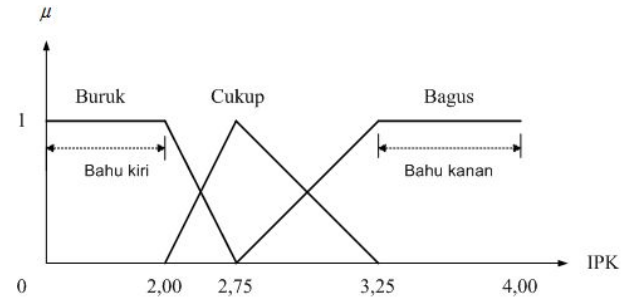


Keputusan Model Mamdani

- Mahasiswa B dengan $IPK = 2,99$ dan Gaji orangtuanya sebesar 1 juta rupiah per bulan memperoleh Nilai Kelayakan sebesar **69,66**.
- Lebih besar dibandingkan dengan Nilai Kelayakan mahasiswa A yang sebesar **52,39**.
- Jadi, mahasiswa B layak mendapatkan beasiswa.



Model Mamdani



IPK \ Gaji	Kecil	Sedang	Besar	Sangat Besar
Buruk	Rendah	Rendah	Rendah	Rendah
Cukup	Tinggi	Rendah	Rendah	Rendah
Bagus	Tinggi	Tinggi	Tinggi	Rendah



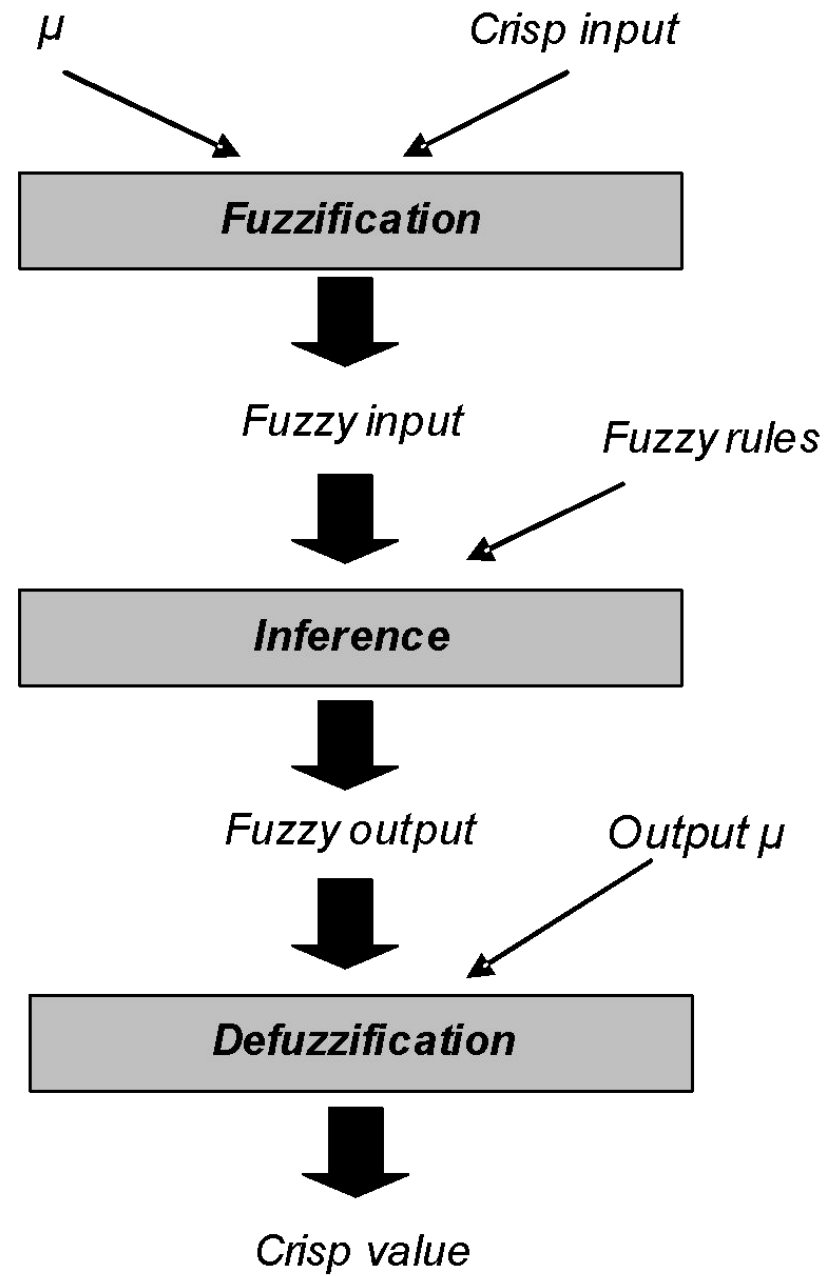
Model Sugeno

- Model ini sering digunakan untuk membangun sistem kontrol yang membutuhkan respon cepat
- Proses perhitungannya sangat sederhana sehingga membutuhkan waktu relatif cepat sehingga sangat sesuai untuk sistem kontrol
- Bagaimana jika digunakan untuk masalah pemberian beasiswa?



Masalah: Pemberian Beasiswa

Mahasiswa	IPK	Gaji Ortu (Rp/bulan)
A	3,00	10 juta
B	2,99	1 juta





Fuzzification & Rule Evaluation

- Misalkan proses *fuzzification*-nya sama persis dengan model Mamdani.
- Misalkan *Rule* yang digunakan juga sama persis dengan model Mamdani.



Mahasiswa A

IF $IPK = \text{Cukup}(0,5)$ AND $Gaji = \text{Besar}(0,4)$ THEN $NK = \text{Rendah}(0,4)$

IF $IPK = \text{Cukup}(0,5)$ AND $Gaji = \text{Sangat Besar}(0,6)$ THEN $NK = \text{Rendah}(0,5)$

IF $IPK = \text{Bagus}(0,5)$ AND $Gaji = \text{Besar}(0,4)$ THEN $NK = \text{Tinggi}(0,4)$

IF $IPK = \text{Bagus}(0,5)$ AND $Gaji = \text{Sangat Besar}(0,6)$ THEN $NK = \text{Rendah}(0,5)$

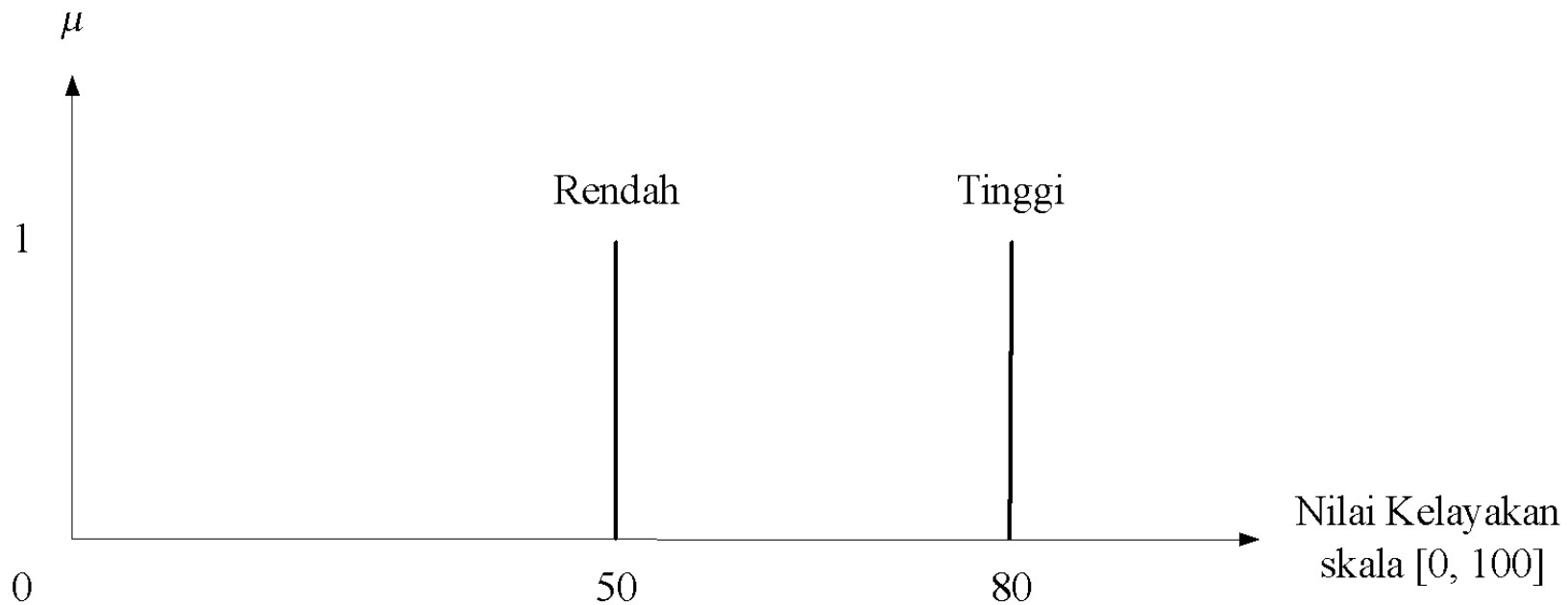


$NK = \text{Rendah}(0,5)$

$NK = \text{Tinggi}(0,4)$

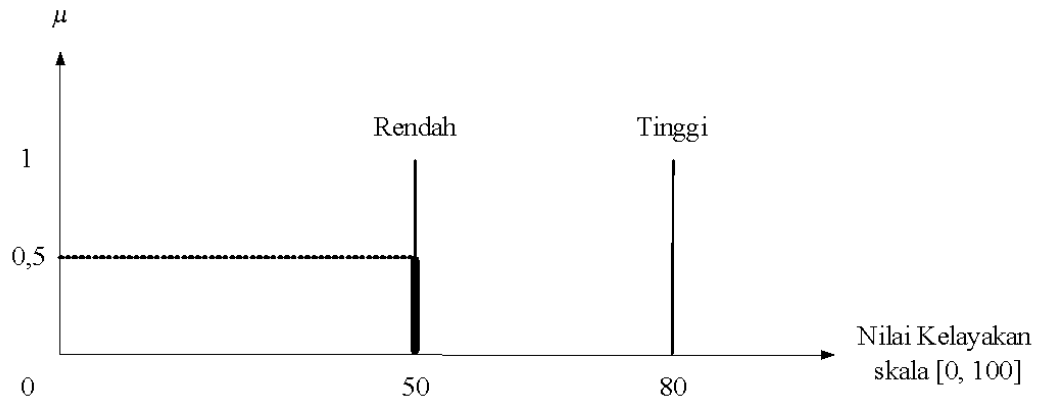


FK *singleton* untuk Nilai Kelayakan

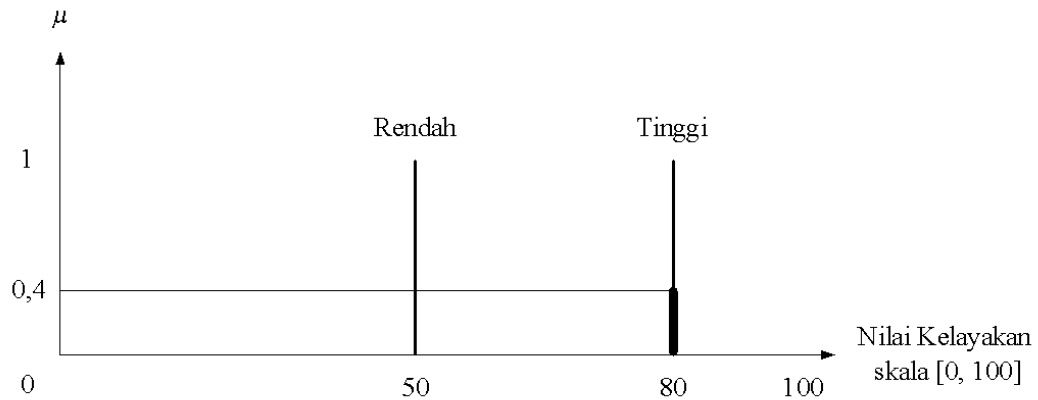




Untuk mahasiswa A



(a)

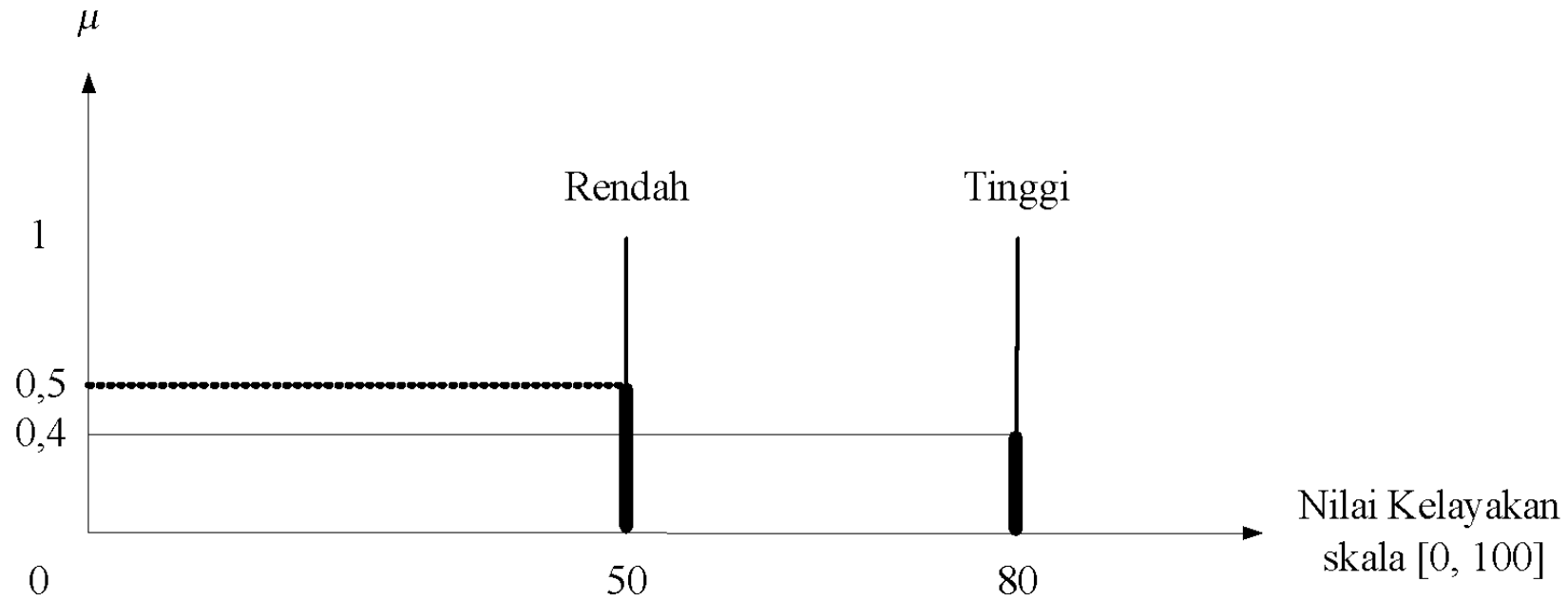


(b)

- NK = Rendah (0,5)
- NK = Tinggi (0,4)



Proses Composition





Defuzzification: Weighted Average

$$y^* = \frac{(0,5)50 + (0,4)80}{(0,5) + (0,4)} = 63,33$$



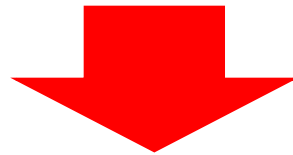
Mahasiswa B

IF $IPK = \text{Cukup}(0,52)$ AND $Gaji = \text{Kecil}(1)$ THEN $NK = \text{Tinggi}(0,52)$

IF $IPK = \text{Cukup}(0,52)$ AND $Gaji = \text{Sedang}(0)$ THEN $NK = \text{Rendah}(0)$

IF $IPK = \text{Besar}(0,48)$ AND $Gaji = \text{Kecil}(1)$ THEN $NK = \text{Tinggi}(0,48)$

IF $IPK = \text{Besar}(0,48)$ AND $Gaji = \text{Sedang}(0)$ THEN $NK = \text{Tinggi}(0)$

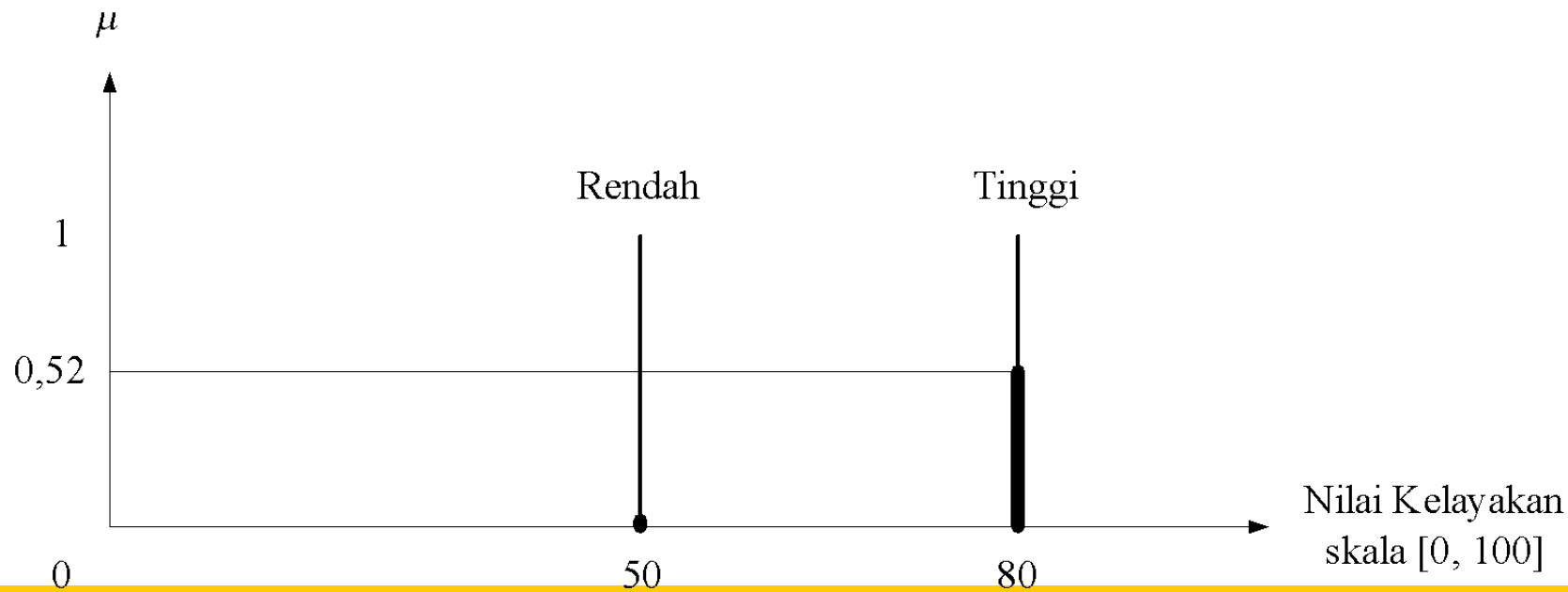


<p>$NK = \text{Rendah}(0)$ $NK = \text{Tinggi}(0,52)$</p>



Untuk Mahasiswa B

- NK = Rendah (0)
- NK = Tinggi (0,52)





Defuzzification: Weighted Average

$$y^* = \frac{(0)50 + (0,52)80}{0 + 0,52} = 80$$

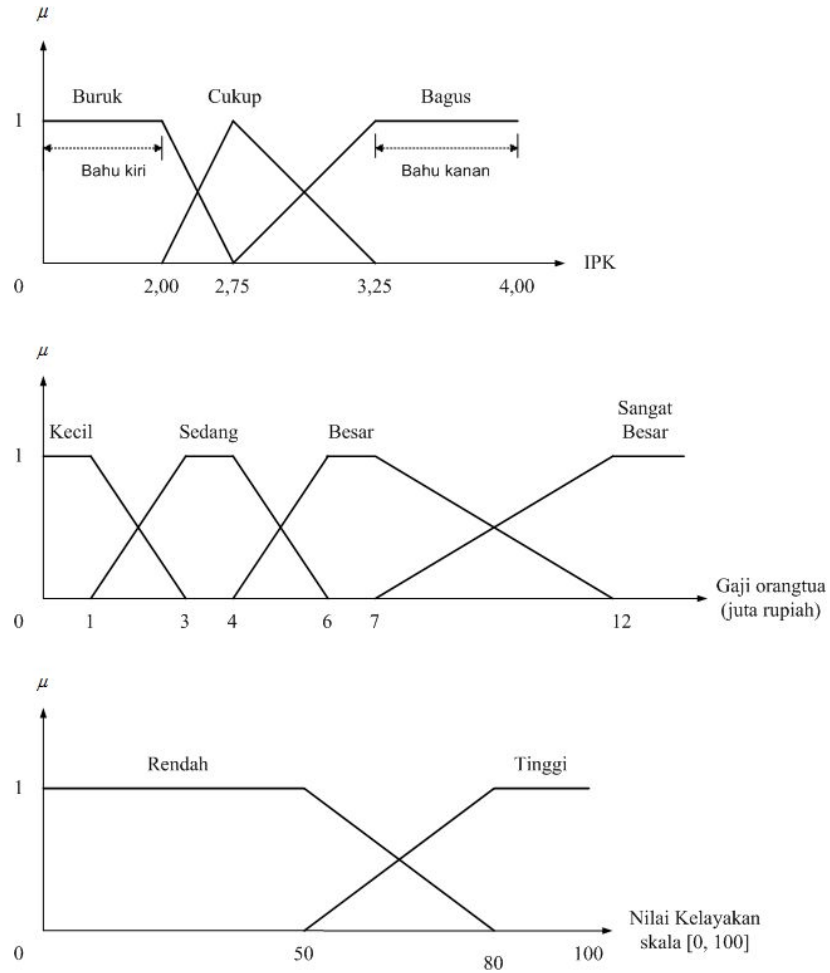


Keputusan Model Sugeno

- Mahasiswa B dengan $IPK = 2,99$ dan Gaji orangtuanya sebesar Rp 1 juta per bulan memperoleh Nilai Kelayakan sebesar **80**.
- Lebih besar dibandingkan dengan Nilai Kelayakan mahasiswa A yang sebesar **63,33**.
- Jadi, mahasiswa B layak mendapatkan beasiswa.

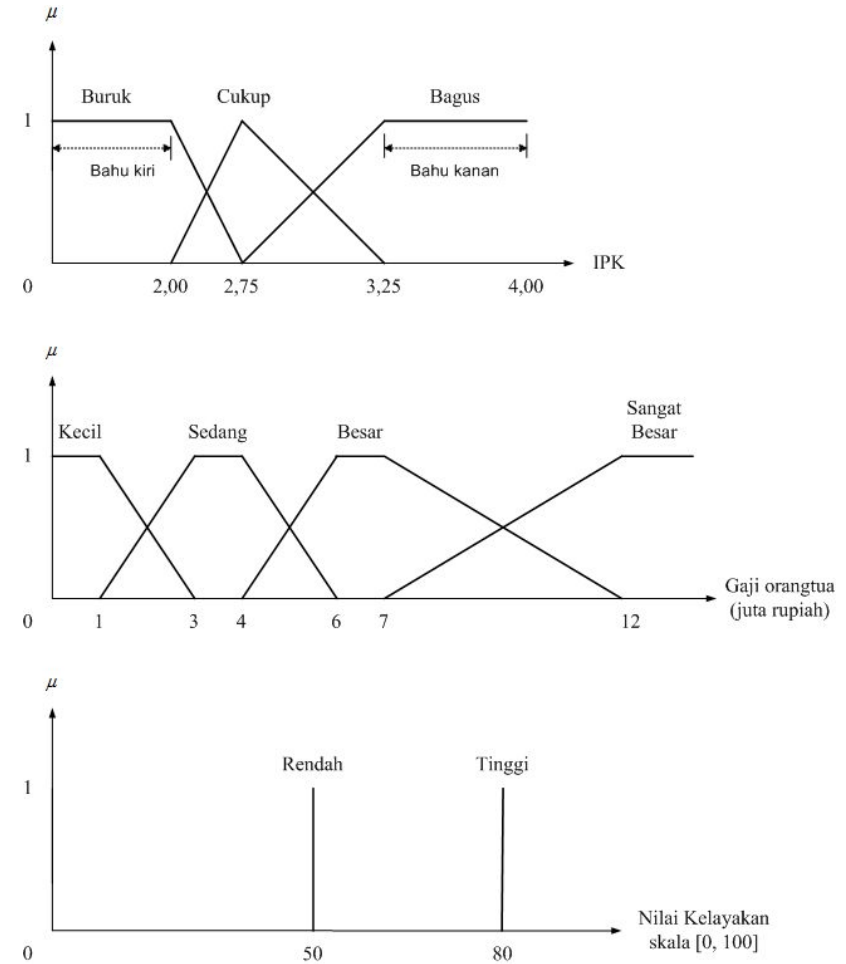


Model Mamdani



IPK \ Gaji	Kecil	Sedang	Besar	Sangat Besar
Buruk	Rendah	Rendah	Rendah	Rendah
Cukup	Tinggi	Rendah	Rendah	Rendah
Bagus	Tinggi	Tinggi	Tinggi	Rendah

Model Sugeno



IPK \ Gaji	Kecil	Sedang	Besar	Sangat Besar
Buruk	Rendah	Rendah	Rendah	Rendah
Cukup	Tinggi	Rendah	Rendah	Rendah
Bagus	Tinggi	Tinggi	Tinggi	Rendah



Nilai Kelayakan mahasiswa A & B

Mahasiswa	Nilai Kelayakan mendapat beasiswa	
	Model Mamdani	Model Sugeno
A	52,39	63,33
B	69,66	80
Selisih A dan B	17,72	16,67



REFERENSI

Suyanto. (2021). *Artificial intelligence: Searching, reasoning, planning, dan learning*. Informatika Bandung.

Politeknik Elektronika Negeri Surabaya. (2017). *Algoritma Pencarian Heuristik*. Departemen Teknik Informatika dan Komputer.



TERIMA KASIH