CSE 440

Lecture 01 : 29.01.2023

- What is AI ?

  - aggregate / global capacity
  - to act purposefully
  - to think rationally
  - to deal effictively with environment

$\boxed{\text{Chap 1: intro}}$

Lecture 02 : 31.01.2023

Turing test : if the interrogator can't tell if it's computer / human.

↳ Acting humanly

Thinking humanly : know very little how brain works.

Thinking rationally : all intelligent behaviour are not included.

Acting rationally : most optimal

Lecture 03 : 5.02.2023

$\boxed{\text{Chap 2: Agent}}$

Agent :

actuator : output

sensor : input

perceiving it's environment

acting upon that environment

Rational Agent : (thinks logically), maximizes the expected value of the performance measure

omni - scient : সবজান্তে
(সব) (জানে)

clair - voyant : ভবিষ্যতের সব জানা
(দেখে)

Lecture 04 : (Env types) 7.02.2023

• fully-observable	see/hear/perceive all
• Partially-observable	partial information (AI)
• Single-Agent	only one intelligent agent
• Multi-Agent	• co-operative • competitive
• Deterministic	100% probability (দাবা)
• Stochastic	random ([illegible]/[illegible]),
• Episodic	independent
• Sequential	dependent
• Discrete	set of actions
• Continuous	action happen continuously
• Static	fully static doesn't exist
• Dynamic	changing environment

Example : (chess play)

1. Fully - observable
2. Multi - Agent
3. Deterministic
4. Sequential
5. Discrete
6. Static

Lecture 05 : 12.02.2023

$\boxed{\text{• Simple reflex agents :}}$ (lowest level AI)

Example : tic - tac - toe

↳ only take decision

$\boxed{\text{• Model - based :}}$ (enviroment represtation)

↳ make a decision + update

no specific goal

$\boxed{\text{• Goal - based :}}$ (agent has a ~~gole~~ goal)

↳ decisions are taken to reach the goal. Example : vaccum cleaner

$\boxed{\text{• Utility - based :}}$ (choose the best actions)

↳ how happy with a state.

human agent is a mixture of all types of agents.

PEAS of **Part - Picking robot :**

P : % of parts in correct bins

E : conveyor belt, parts , bins

A : jointed arm, hand

S : camera, jointed angle sensor

Lecture 06 :

14.02.2023

State : Possible combination

Action : Blank space more

Time and space complexity –

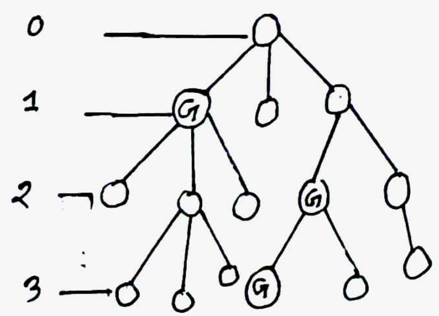
b – how many children (max)

(for agent) – no. of action

d – level

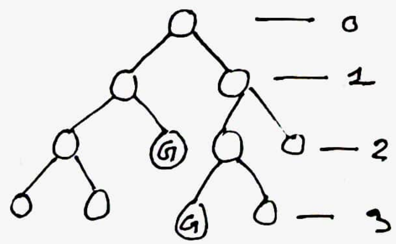
m – maximum depth

Example 1:



b – 3
d – 1
m – 3

Example 2:



b – 2
d – 2
m – 3

BFS – left to right (→) [Queue] /
DFS – up to down (↓) [Stack]

infinite (b) – BFS (incomplete)
infinite (d) – DFS (incomplete)

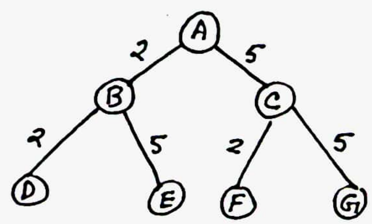
Lecture 07 :

19.02.2023

BFS — complete ✓   Time (b^d)
optimal ✓   Space (b^d)

DFS — complete ✗   Time (b^m)
optimal ✗   Space (bm)

Uniform - cost - Search



Priority Queue

A
B(2), C(5)
D(4), C(5), E(7)
C(5), E(7)
E(7), F(7), G(10)

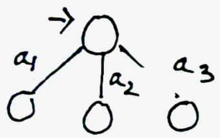
Lecture 08 : 26.02.2023

Heuristic Search

↳ Approximate result

↳ informed search

Blind-first-Search :



$a_1 = a_2 = a_3$

for desirable approach → heuristic

Example : (puzzle problem)

$h_1(S) = 8$

$h_2(S) = 18$ → closer to actual cost

↳ admissible

DFS – Expand the deepest node.

BFS – Expand the shallowest node.

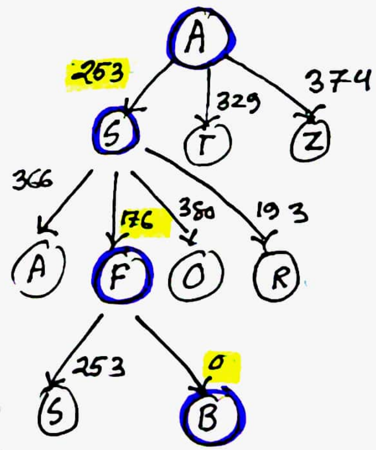
Uniform – Expand the least cost node.

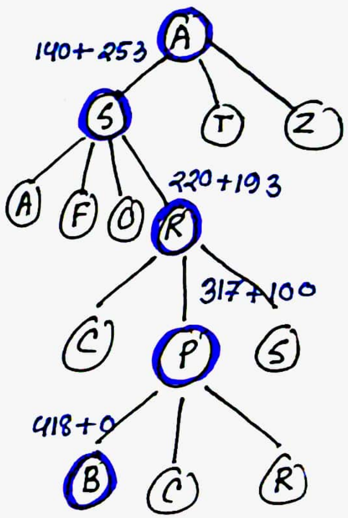
Best-First-Search :

Greedy – takes optimal solution first

A* – actual cost + heuristic cost

Example :





Greedy – not optimal

$f(n) = h(n)$

↳ heuristic cost

A* – optimal

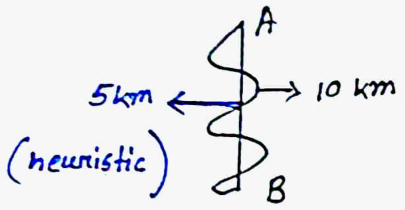
$f(n) = g(n) + h(n)$

Lecture 09 :

26.02.2023

If heuristic is admissible $\longrightarrow$ A^* optimal

Example :



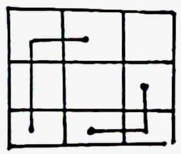
Why A^* optimal ?

→ effective branching factor reduce

$f(n) = g(n) + h(n)$

manhatten distance : horizontal / vertical

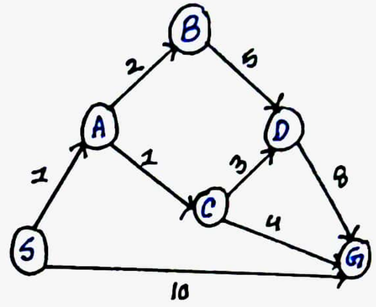
Example :



Admissible heuristic → go for a non-restrict approach.

Lecture 10 :

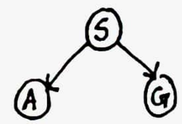
5.03.2023



Heuristic table

nodes	h(n)
S	5
A	3
B	4
C	2
D	6
G	0

BFS : (weight ✗)

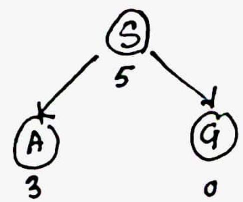


Order of visit node	Queue
S	S
A	A G
G	G B C

NOTE : BFS is optimal → path cost equal

$\therefore$ $solut^n$ path : S – G
$\therefore$ $Solut^n$ cost : 10

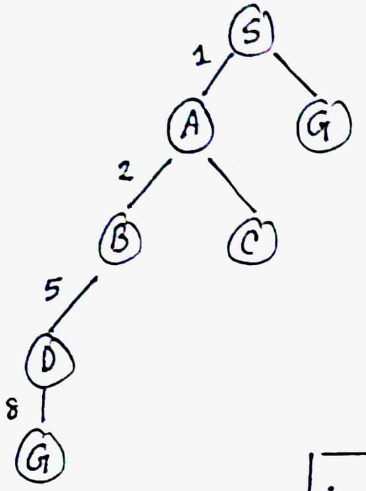
Greedy : (heuristic table ✓)



Order of visit node	PQueue
	S(5)
S	~~S~~ G(0)
G	

$\therefore$ $solut^n$ path : S – G
$\therefore$ $solut^n$ cost : 10

DFS : (weight value X)



Order of visited node

S
A
B
D
G

∴ $solut^n$ path : S - A - B - D - G

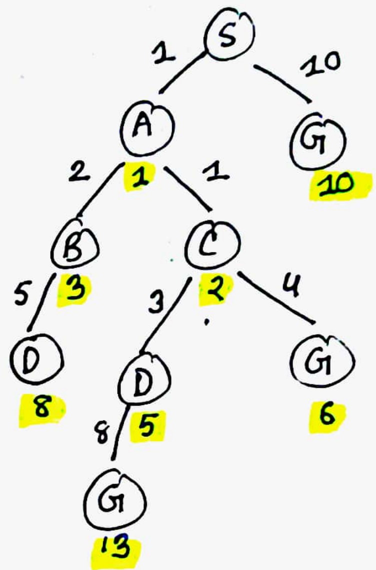
Stack

S
A G
B C G
D C G
G C G

∴ $solut^n$ cost :

1 + 2 + 5 + 8
= 16

Uniform Cost Search (weight value ✓)



Order of visited nodes

S
A
C
B
D
G

∴ $solut^n$ path : S-A-~~G~~C-G

∴ $solut^n$ cost : 1+1+4 = 6

PQueue :

$\langle S,0 \rangle$

$\langle A,1 \rangle$, $\langle G,10 \rangle$

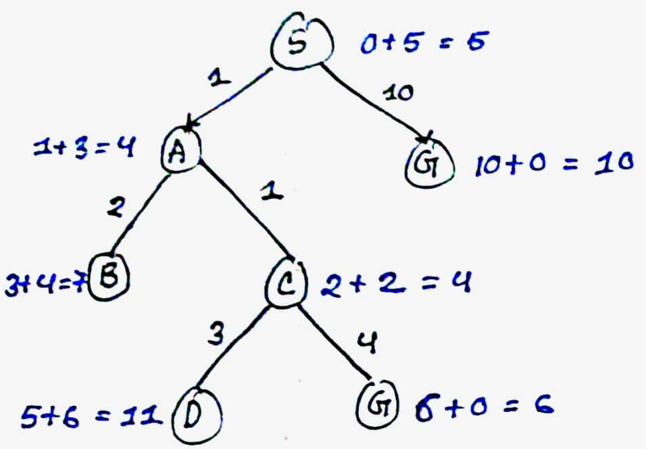
$\langle C,2 \rangle$, $\langle B,3 \rangle$, $\langle G,10 \rangle$

$\langle B,3 \rangle$, $\langle D,5 \rangle$, $\langle G,6 \rangle$, $\langle G,10 \rangle$

$\langle D,5 \rangle$, $\langle G,6 \rangle$, $\langle D,8 \rangle$, $\langle G,10 \rangle$

$\langle G,6 \rangle$, $\langle D,8 \rangle$, $\langle G,10 \rangle$, $\langle G,13 \rangle$

A* search : (heuristic + weight ✓)



Order of visited node

S
A
C
G

PQueue :

<S, 5>

<A, 4> , <G, 10>

<C, 4> , <B, 7> , <G, 10>

<G, 6> , <B, 7> , <G, 10> , <D, 11>

∴ Solut^n path : S – A – C – G

∴ Solut^n cost : 1 + 1 + 4 = 6

Lecture 11 — 7.03.2023

Adversial search – Game (difficult)

↳ unpredictable opponent

↳ time limit

Example : (Tic-Tac)

Terminal utility –

- X wins (+1) MAX
- O wins (–1) MIN
- draw (0)

Env types :

FO, MA, D, Static, Seq, Discrete

Property of Minimax :

- Complete ✓
- Optimal ✓

- branching factor of chess : b ≈ 35
- maximum depth : m ≈ 100

Lecture 12 :

12·03·2023

Questions for Quiz :

1· PEAS description

2· Types of agent

3· Environment types with justification

4· Practised maths in Lecture 10·

5· Branching factor ? Depth size ?

which searching algorithm best for memory ?

6· Blind / Heuristic search ?

α- β purning example : ($b^m = b^{m/2}$)

≥ 3 / MAX — $\alpha \uparrow$

≤ 2 / MIN — $\beta \downarrow$

$\alpha \geq \beta$ $\longrightarrow$ Prune

↳ order matters

MAX update α

Better approach :

evaluation func + cut-off search

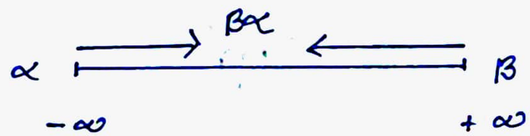
↳ unfinished game evaluate

Example :

Eng — 170 (20)

Ban — 70/2 (10)

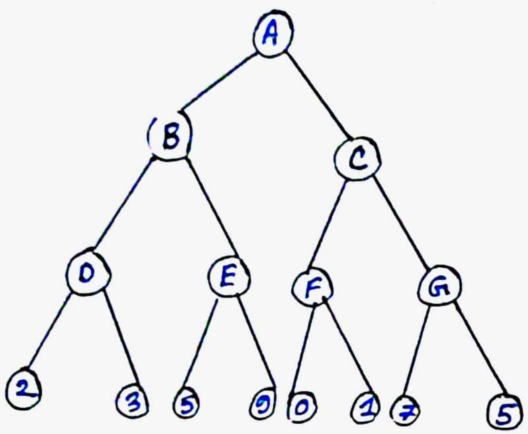
Eng (-1) Tie (0) Ban (+1)



Lecture 13 :

19·03·2023

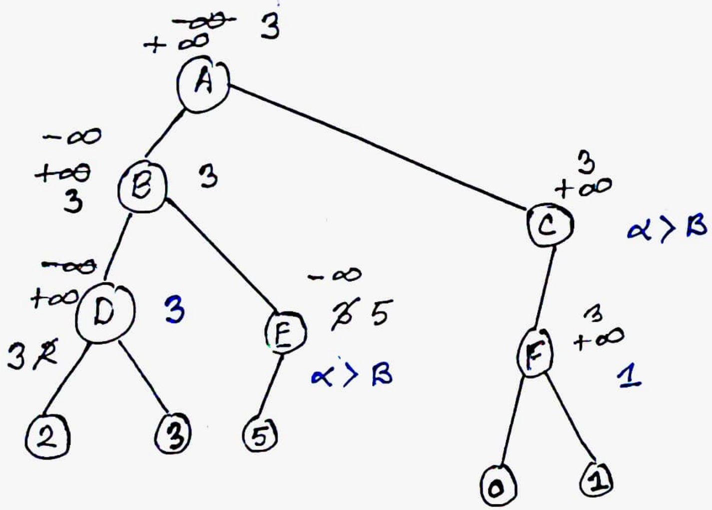
Example :

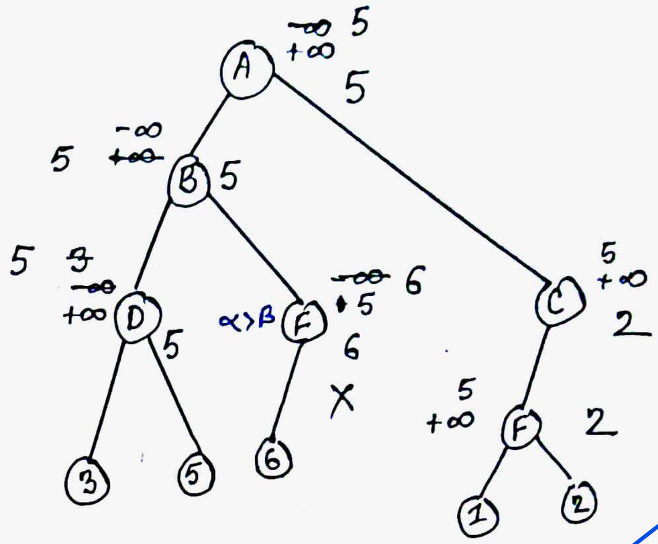


MAX

MIN

MAX

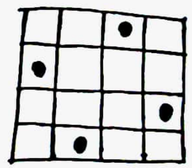




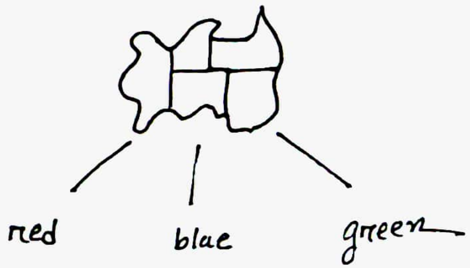
Lecture 14 :

21·03·2023

4-queens problem :



Back-tracking Example :



Heuristic Solution :

- Minimum remaining value

→ less value ~~of~~ remaining ✓

- Degree heuristic

→ remaing value = equal then check node connection (high)

- least constrain

Forward checking :

Lecture 15 :

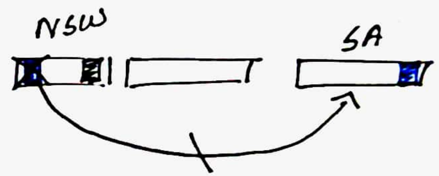
28.03.2023

Arc consistancy :

$X \longrightarrow Y$

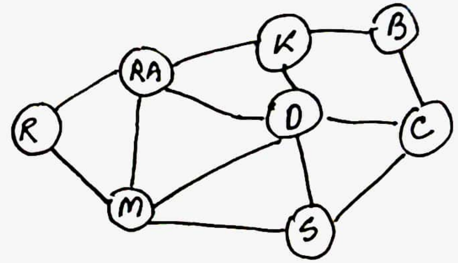
for every value of X,

there is some value of Y.



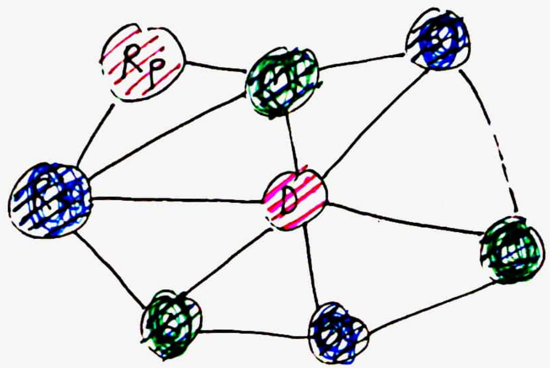
→ not consistant

Problem :



D = 6

M = 4



Lecture 21: 30.04.2023

Propositional logic :

statement that have truth value

knowledge base :

knows everything as True .

In-Class - Example :

mythical = y

mortal = R

mammal = M

horned = H

magical = G

1. $y \Rightarrow \neg R$
2. $\neg y \Rightarrow (R \land M)$
3. $(\neg R \lor M) \Rightarrow H$
4. $H \Rightarrow G$
5. $G \land H = \boxed{\alpha}$

CNF format :

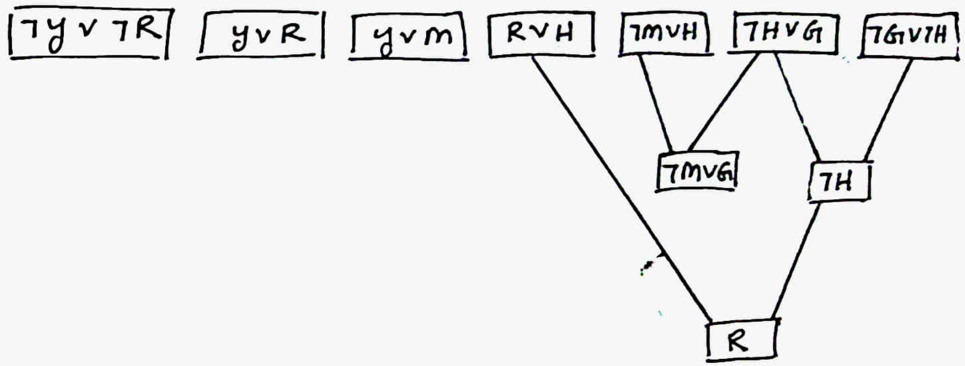
$\boxed{KB \land \neg\alpha}$

↳ not satisfied

$\therefore \alpha = T$

1. $\neg y \lor \neg R$
2. $y \lor (R \land M) \equiv (y \lor R) \land (y \lor M)$
3. $(R \land \neg M) \lor H \equiv (R \lor H) \land (\neg M \lor H)$
4. $\neg H \lor G$
5. $\neg\alpha$: $\neg G \lor \neg H$

Resolution :



Lecture 22 : 7.05.2023

Predicate → return True value
(Boolean value)

Example :

- Rahim is smart

↳ Smart(x)

Smart(Rahim)

- At(x, y) : "x is at y"

Example :

- Domain NSU

$\forall x$ smart(x)

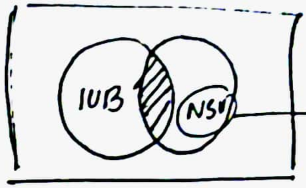
$\exists x$ smart(x)

$\boxed{\forall x - \Rightarrow}$

- Domain World

$\forall x$ At(x, NSU) $\Rightarrow$ Smart(x)

- $\exists x$ At(x, IUB) $\wedge$ Smart(x) $\boxed{\exists x - \wedge}$



→ Subset

Properties of quantifiers :

Loves(x, y) : x loves y

- $\exists x \forall y$ Loves(x, y) : There are some people who loves all. (universal love)
- $\forall y \exists x$ Loves(x, y) : All people is loved by someone.

$\neq$

- $\forall x \forall y$ loves(x, y) : Every p loves Every p
- $\exists x \exists y$ loves(x, y) : Some p loves Some p

$\neq$ ~~· $\exists x$ $\forall y \exists x$ loves(x, y) : Everyone is loved by someone~~

- $\forall x\, \exists y$ loves(x, y) : Everyone loves someone

- $\exists y\, \forall x$ loves(x, y) : some people are loved by someone

$\exists x\ \neg$ likes(x, ice-cream)

$\neg \forall x$ likes(x, ice-cream)

$\neg \exists x$ likes(x, Brocoli)

$\forall x\ \neg$ likes(x, Brocoli)

Lecture 23: 9.05.2023

8.10:

$\boxed{\exists x : \wedge}$

$\boxed{\forall x : \Rightarrow}$

Occupation(P, O)

Customer(P_1, P_2)

Boss(P_1, P_2)

a. occupation(Emily, Surgeon) $\vee$ occupation(Emily, lawyer)

b. Occ(Joe, Actor) $\wedge\ \exists x$ Occ(Joe, x) $\wedge\ x \neq$ Actor

c. $\forall x$ Occ(x, surgeon) $\Rightarrow$ Occ(x, Doctor)

d. $\neg$ Customer(Joe, lawyer) ×

$\hookrightarrow \neg \exists x$ Customer(Joe, x) $\wedge$ Occ(x, lawyer)

e. $\exists x \{ \text{Boss}(x, \text{Emily}) \land \text{Occ}(x, \text{lawyer}) \}$

~~f. $\exists x \{ \text{Occ}(x, \text{lawyer}) \land$~~
~~$\forall x \{ \text{Customer}(x, \text{doctor})$~~

f. $\exists x \; \text{Occ}(x, \text{lawyer}) \land [\forall y \; \text{Cus}(y, x)$

$\Rightarrow \text{Occ}(y, \text{Doctor})]$

g. $\forall x \; \text{Occ}(x, \text{Surgeon}) \Rightarrow [\exists y \; \text{Occ}(y, \text{law})$

$\land \text{Cus}(x, y)]$

Lecture 24 :

14.05.2023

<u>8.11</u> :

- $\text{Friends}(x, y)$: x and y are friends
- $\text{Speakslanguage}(x, L)$: x speaks L
- $\text{Understand}(x, y)$: x understand y

$$\boxed{\forall x \forall y \forall l \equiv \forall x, y, l}$$

a. people speaking the same language can understand each other.

c. (i) $\forall x, y \; \text{Understand}(x, y) \land \text{Understand}(y, x)$

$\Rightarrow \text{Friends}(x, y)$

Transitive relation :

$a > b$ and $b > c$

$\therefore a > c$

$\forall x, y, z$

(ii) $\text{Friend}(x, y) \land \text{Friend}(y, z) \Rightarrow \text{Friend}(x, z)$

(11) Given that,

$S(x)$
$F(x)$
$A(x, y)$

a. $A(\text{Lois}, \text{Professor})$

b. $\forall x\ S(x) \Rightarrow A(x, \text{Professor Gross})$

c. $\forall x\ F(x) \Rightarrow A(x, \text{Professor Miller}) \wedge \vee$

$A(\text{Professor Miller}, x)$

d. ~~$\exists x\ S(x)$~~ $\exists x\ \forall y\ ((S(x) \wedge F(y)) \Rightarrow$

$\neg A(x, y) \equiv \exists x\ S(x) \wedge [\forall y\ (F(y) \Rightarrow \neg A(x,y)]$

e. ~~$\exists x\ F(x)$~~ $Ex\ F(x) \wedge [\forall y\ S(y) \Rightarrow \neg A(y, x)]$

d. $\exists x S(x) \wedge [\neg Ey\ F(y) \wedge A(x, y)]$

e. $\exists x F(x) \wedge [\neg Ey\ S(y) \wedge A(y, x)]$

f. $\exists x\ S(x) \wedge [\forall y\ F(y) \Rightarrow A(x, y)]$

g. $\exists x\ F(x) \wedge [\forall y\ F(y) \Rightarrow A(x, y)]$ X

h. $\exists x\ S(x) \wedge [\forall y\ F(y) \Rightarrow \neg A(y, x)]$

g. $\exists x\ F(x) \wedge [\forall y\ F(y) \wedge y \neq x \Rightarrow A(x, y)]$

Lecture 25 : (uncertainty) 16·05·2023

- partially observable
- dynamic Environment
- stochastic Environment

Agent :

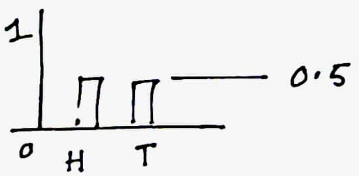
- Expected outcome (Probability)

Axioms of Probability :

- $0 \leq P(A) \leq 1$
- $P(\text{True}) = 1 \mid P(\text{False}) = 0$
- $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$

Random value : (Prob· distribution)

Example : Coin toss $= X$



Lecture 26 : May 23

- $P(A)$
- $P(A \mid B)$
- $P(A \wedge B) = P(A \mid B)\, P(B) = P(B \mid A)\, P(A)$

↳ Baye's Rule : $\dfrac{P(B \mid A)\, P(A)}{P(B)} = P(A \mid B)$

Prior Probability : (unconditional) $P(A)$

Posterior Probability : (conditional) $P(A \mid B)$

$$P(\text{cause} \mid \text{effect}) = \frac{P(\text{effect} \mid \text{cause}) \cdot P(\text{cause})}{P(\text{effect})}$$

↳ effect less $\propto$

Independence :

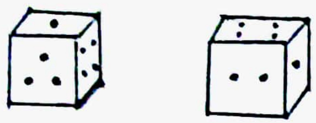
$$P(A \wedge B) = P(A) \cdot P(B)$$

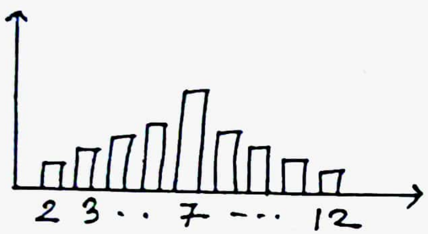
$$P(A \wedge B \wedge C) = P(A) \cdot P(B) \cdot P(C)$$

$A \perp B$

Example :

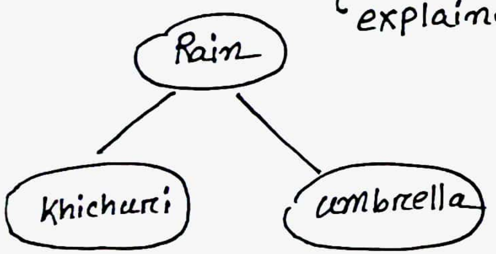
01318980252





Conditional Independence :

'explained away'



Day	K	U
1	✓	✓
2	✓	✓
3	✗	✗
4	✗	✗

$\boxed{K \perp U \mid R}$

↳ if rain is given only then

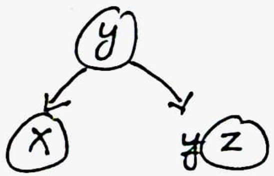
- $P(K \mid U, R) = P(K \mid R)$
- $P(U \mid K, R) = P(U \mid R)$

Example :

$\Rightarrow$ explained away

$R \perp J \mid P$

Building Block :



$x \perp z \mid y$

Lecture 27 : 28.05.23

↳ (Joint probability)

From table —

$P(A = T) : 0.3 + 0.1 + 0.05 + 0.15$

$P(B = F, C = F) : 0.1 + 0.3$

Problem :

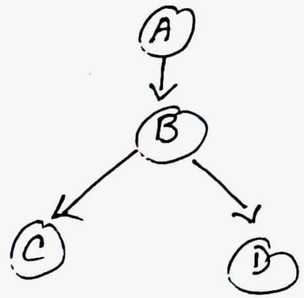
$$P(A \mid \neg B) : \frac{P(A, \neg B)}{P(\neg B)}$$

$$= \frac{0.3 + 0.1}{0.1 + 0.2 + 0.3 + 0.1} = \frac{0.4}{0.7}$$

As,

$$\boxed{P(A, \neg B) = P(A \mid \neg B) \cdot P(\neg B)}$$

A Bayesian Network : (DAG)

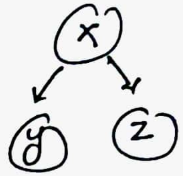


$A \perp C \mid B$

$A \perp D \mid B$

$C \perp D \mid B$

$P(A \mid C, B) = P(A \mid B)$



$P(x) \left(P(y \mid x) \; P(z \mid y, x) \right)$

↳ $P(x) \; P(y \mid x) \; P(z \mid x)$

$P(A, B) = P(A \mid B) \; P(B)$

$P(A, B, C) = P(A) \; P(B \mid A) \; P(C \mid B, A)$

↳ if A, B is give

then prob. of C

$P(J \mid A) + P(J \mid \neg A)$