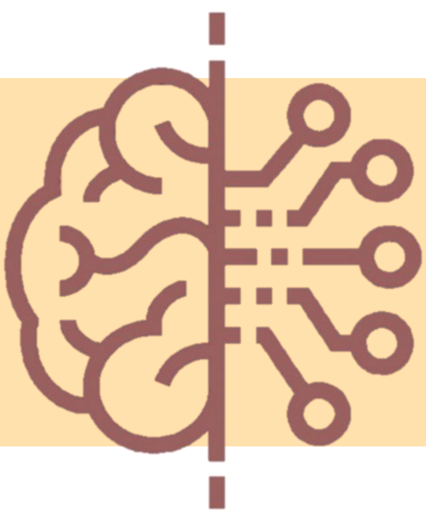
Mid Term Test
Revision



Artificial Intelligence

School of Computing
Universiti Teknologi Malaysia

Principles of Responsible AI

	Challenge or Risk	Example
Fairness	Bias can affect results	A loan-approval model discriminates by gender due to bias in the data with which it was trained
Reliability & Safety	Errors may cause harm	An autonomous vehicle experiences a system failure and causes a collision.
Privacy & Security	Data could be exposed	A medical diagnostic bot is trained using sensitive patient data, which is stored insecurely
Inclusiveness	Solutions may not work for everyone	A predictive app provides no audio output for visually impaired users
Transparency	Users must trust a complex system	An AI-based financial tool makes investment recommendations - what are they based on?
Accountability	Who's liable for AI-driven decisions?	An innocent person is convicted of a crime based on evidence from facial recognition – who's responsible?

Exercise #1

Example of application	Principles of Responsible AI	Reason
1. An AI-powered facial recognition system used for security purposes consistently misidentifies individuals with darker skin tones more frequently than those with lighter skin tones. This bias occurs due to the lack of diversity in the dataset used to train the algorithm, which primarily consisted of images of individuals with lighter skin tones.		
2. A healthcare AI system responsible for diagnosing medical conditions encounters a situation where it fails to provide accurate diagnoses for a specific subset of patients. Upon investigation, it is discovered that the AI model was not adequately trained on data from a certain demographic or geographical region, leading to unreliable predictions for those populations.		
3. An investment firm utilizes an AI-driven algorithm to make trading decisions automatically. However, due to a flaw in the algorithm's logic, the system makes several erroneous trades, resulting in significant financial losses for the firm and its clients. Upon investigation, it is found that the error was a result of inadequate testing and oversight during the algorithm's development phase.		

Exercise #1

Example of application	Principles of Responsible AI	Reason
4. A company uses an AI-driven tool to screen job candidates for interviews. The tool automatically filters resumes based on various criteria, including qualifications, experience, and keywords. However, some candidates notice discrepancies in the selection process, as certain resumes with similar qualifications are either accepted or rejected inconsistently.		
5. A popular voice assistant AI collects and stores voice recordings and personal information from users to improve its functionalities and personalize user experiences. However, a security breach occurs, leading to unauthorized access to the stored data by a third-party entity.		
6. An AI-powered language translation tool primarily focuses on translating text from widely spoken languages but overlooks the inclusion of regional dialects and languages spoken by smaller communities. As a result, speakers of these dialects or languages face difficulties using the tool, hindering their access to vital information and resources available in mainstream languages.		

Exercise #2: Knowledge Representation

Convert the English sentences in parts (I) – (IV) into standard predicate logic sentences when given the following propositions:

A: x is creating content

T: x is selfie

P: x has a Tik Tok account

C: x likes to eat

F: x likes traveling

No	English sentence	Predicate Logic
I.	Someone is creating content and selfie.	
II.	Not all creating content must likes traveling or likes to eat.	
III.	Every Tiktokers has a Tik Tok account is creating content.	
IV.	Some people likes traveling will like to eat.	

Exercise #2 : Knowledge Representation

Convert the English sentences in parts (I) – (IV) into standard predicate logic sentences when given the following propositions:

p: It is sunny this afternoon
q: it is colder than yesterday
r: We will go swimming
s: We will take a canoe trip
t: We will be home by sunset

No	English sentence	Predicate Logic
I.	It is not sunny this afternoon and it is colder than yesterday.	
II.	We will go swimming only if it is sunny.	
III.	If we do not go swimming, then we will take a canoe trip.	
IV.	If we take a canoe trip, then we will be home by sunset.	

Activity: Universal FOL conversion

Remember the basic syntax: ∀*x* constant(*x*) → predicate(*x*)

English sentence	FOL sentence
All birds fly ∀ x variable sentences ∀ x (x is bird) (bird is fly)	**∀ x bird (x) → fly (x)**
Every man respects his parent ∀ x variable sentences ∀ x (x is man) (man respects parent)	**∀ x man (x) → respect (x,parent)**

Activity: Universal FOL conversion

Remember more syntax: ∀*x* predicate(*x*,constant) → predicate(*x*)

English sentence	FOL sentence
Everyone learn AI is intelligent ∀ x variable sentences ∀ x (x is everyone) (everyone learn AI) (everyone is intelligent)	**∀ x Learn(x, AI) → Intelligent(x)**
Santokh learn AI is intelligent (Santokh learn AI) (Santokh is intelligent)	Learn (Santokh, AI) → Inteligent (Santokh)
Every customer likes KFC is happy ∀ x variable sentences Let customer is *x*, ∀ x (x is customer) (customer like KFC) (customer is happy)	**∀ x Like (x,KFC) → happy (x)**
Sara likes KFC is happy (Sara like KFC) (Sara is Happy)	**?**

Activity: Universal FOL conversion

Remember more syntax: ∀*x* predicate(*x*,constant) → predicate(*x*)

English sentence	FOL sentence
Everyone learn AI is intelligent **∀ x variable sentences** **∀ x (x is everyone)** (everyone learn AI) (everyone is intelligent)	**∀*x* Learn (*x*, AI) → Intelligent (*x*)**
Santokh learn AI is intelligent **(Santokh learn AI)** (Santokh is intelligent)	**Learn (*Santokh*, AI) → Intelligent (*Santokh*)**
Every customer likes KFC is happy **∀ x variable sentences** **Let customer is *x*, ∀ x (x is customer)** (customer like KFC) (customer is happy)	**Let customer is *x*, ∀*x* Like (*x*, KFC) → Happy (*x*)**
Sara likes KFC is happy **(Sara like KFC) (Sara is Happy)**	**Like (*Sara*, KFC) → Happy (*Sara*)**

The mistake

- What does this statement mean:

  $\forall x\ Taking(x,AI) \wedge Smart(x)$

- Common mistake: using $\wedge$ as the main connective with $\forall$:

  $\forall x\ Taking(x,AI) \wedge Smart(x)$

  means "Everyone is taking AI and everyone is smart"

Existential Quantification

MEMORIZE THIS SYNTAX

- Syntax: $\exists$*variables sentence*
- Someone taking AI is smart:

  $\exists x\ Taking(x,AI) \wedge Smart(x)$
- Semantics: $\exists x\ S$ is equivalent to the *disjunction of instantiations* of *S*

$$
\begin{aligned}
& (Taking(Ann,AI) \wedge Smart(Ann)) \\
\vee \quad & (Taking(John,AI) \wedge Smart(John)) \\
\vee \quad & \ldots
\end{aligned}
$$

- Typically, $\wedge$ is the main connective with $\exists$.

Activity: Existential FOL conversion

Remember the basic syntax: ∃ x variable sentences

English sentence	FOL sentence
Some boys play cricket ∃ x variable sentences (*x* is a boy) and (*x* play cricket) and this is true for some *x*	**∃ x boy(x) ∧ play(x,cricket)**
Someone like broccoli is healthy ∃ x variable sentences (*x* is someone) (*x* like brocolli) and (x is healthy) and this is true for someone *x*	**∃ x Like(x, brocolli) ∧ healthy(x)**
Anyone like broccoli ∃ x variable sentences (*x* is anyone) (*x* like brocolli) and this is true for any *x*	**∃ x Like(x, brocolli)**

The mistake

- What does this mean:

  $\exists x\ Taking(x, AI) \rightarrow Smart(x)$

- Common mistake: using $\rightarrow$ as the main connective with $\exists$:

  $\exists x\ Taking(x, AI) \rightarrow Smart(x)$ is true if there is anyone who is not taking AI!

Properties of Quantifiers

- $\forall x \forall y$ is the same as $\forall y \forall x$
- $\exists x \exists y$ is the same as $\exists y \exists x$
- *$\exists x \forall y$ is not the same as $\forall y \exists x$*

$\exists x \forall y Loves(x, y)$

$\forall y \exists x Loves(x, y)$

Properties of Quantifiers

- $\forall x \forall y$ is the same as $\forall y \forall x$
- $\exists x \exists y$ is the same as $\exists y \exists x$
- *$\exists x \forall y$ is not the same as $\forall y \exists x$*

$\exists x \forall y Loves(x, y)$

"There is a person who loves everyone in the world"

$\forall y \exists x Loves(x, y)$

"Everyone in the world is loved by at least one person"

Quantifier Duality

Each quantifier can be expressed using the other quantifier and negation:

- $\forall x$ *Likes(x,IceCream)* is equivalent to $\neg\exists x\ \neg$*Likes(x,IceCream)*
- $\exists x$ *Likes(x,Broccoli)* is equivalent to $\neg\forall x\ \neg$*Likes(x,Broccoli)*

Everyone likes ice cream == Not someone not liking ice cream (nobody do not like ice cream)

Someone like broccoli == Not everyone not liking broccoli

Proof methods

Proof methods divide into (roughly) two kinds:

- *Model checking*:
  - Truth table enumeration (sound and complete for propositional logic)
  - Heuristic search in model space (sound but incomplete)
- *Application of inference rules*:
  - Legitimate (sound) generation of new sentences from old
  - A *proof* is a sequence of inference rule applications
  - Inference rules can be used as operators in a standard search algorithm!

Proof Method using Inference Rules

The proof process can be viewed as a *search* in which the operators are *inference rules*:

- *Modus Ponens (MP)*

$$\frac{\alpha, \quad \alpha \rightarrow \beta}{\beta} \qquad \frac{\textit{Takes(Joe,AI)} \quad \textit{Takes(Joe,AI)} \rightarrow \textit{Cool(Joe)}}{\textit{Cool(Joe)}}$$

- *And-Introduction (AI)*

$$\frac{\alpha \quad \beta}{\alpha \wedge \beta} \qquad \frac{\textit{Cool(Joe)} \quad \textit{CSMajor(Joe)}}{\textit{Cool(Joe)} \wedge \textit{CSMajor(Joe)}}$$

- *Universal Elimination (UE)*

$$\frac{\forall x \alpha}{\alpha\{x/\tau\}} \qquad \frac{\forall x \ \textit{Takes(x,AI)} \rightarrow \textit{Cool(x)}}{\textit{Takes(Pat,AI)} \rightarrow \textit{Cool(Pat)}}$$

τ must be a ground term: a term with no variables

Exercise #3: Proof Method using Inference Rules

You are given the following scenario:

If Nor has a fever, then Nor has red eyes symptom. If Nor has fatigue, then Nor has a fever. Nor has fatigue.

Using the above scenario, construct a proof using inference rules for: “Nor has red eyes symptom”
Use the following notation to represent the problem:
D: Nor has red eyes symptom
E: Nor has a fever
F: Nor has fatigue

Refutation resolution

- The previous example was easy because it had very few clauses
- When we have a lot of clauses, we want to *focus* our search on the thing we would like to prove
- We can do this as follows:
  - Assume that our fact base is consistent (we can't derive NIL)
  - Add the *negation* of the thing we want to prove to the fact base
  - Show that the fact base is now inconsistent
  - Conclude the thing we want to prove

Proof Method - Resolution

- Resolution is a sound and complete inference method for first-order logic.
- Resolution is a *refutation* procedure: to prove that $KB \models \alpha$, we show that $KB \wedge \neg\alpha$ is unsatisfiable
- The knowledge base and $\neg\alpha$ are expressed in universally quantified, conjunctive normal form
- Like in propositional logic, the resolution inference rule combines two clauses to make a new one:

$$\begin{tikzcd} C_1 \arrow[dr] & & C_2 \arrow[dl] \\ & C & \end{tikzcd}$$

- Inference continues until an empty clause is derived (contradiction)

Conjunctive normal form (CNF) is an approach to Boolean logic that expresses formulas as conjunctions of clauses with an AND or OR. Each clause connected by a conjunction, or AND, must be either a literal or contain a disjunction, or OR operator. CNF is useful for automated theorem proving.

In conjunctive normal form, statements in Boolean logic are conjunctions of clauses with clauses of disjunctions. In other words, a statement is a series of ORs connected by ANDs.

For example:
(A OR B) AND (C OR D)
(A OR B) AND (NOT C OR B)
The clauses may also be literals:
A OR B
A AND B

Literals are seen in CNF as conjunctions of literal clauses and conjunctions that happen to have a single clause. It is possible to convert statements into CNF that are written in another form, such as disjunctive normal form.

Inference rules, equivalence

$(\alpha \wedge \beta)$	$\equiv$	$(\beta \wedge \alpha)$	commutativity of $\wedge$
$(\alpha \vee \beta)$	$\equiv$	$(\beta \vee \alpha)$	commutativity of $\vee$
$((\alpha \wedge \beta) \wedge \gamma)$	$\equiv$	$(\alpha \wedge (\beta \wedge \gamma))$	associativity of $\wedge$
$((\alpha \vee \beta) \vee \gamma)$	$\equiv$	$(\alpha \vee (\beta \vee \gamma))$	associativity of $\vee$
$\neg(\neg\alpha)$	$\equiv$	α	double-negation elimination
$(\alpha \Rightarrow \beta)$	$\equiv$	$(\neg\beta \Rightarrow \neg\alpha)$	contraposition
$(\alpha \Rightarrow \beta)$	$\equiv$	$(\neg\alpha \vee \beta)$	implication elimination
$(\alpha \Leftrightarrow \beta)$	$\equiv$	$((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))$	biconditional elimination
$\neg(\alpha \wedge \beta)$	$\equiv$	$(\neg\alpha \vee \neg\beta)$	de Morgan
$\neg(\alpha \vee \beta)$	$\equiv$	$(\neg\alpha \wedge \neg\beta)$	de Morgan
$(\alpha \wedge (\beta \vee \gamma))$	$\equiv$	$((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))$	distributivity of $\wedge$ over $\vee$
$(\alpha \vee (\beta \wedge \gamma))$	$\equiv$	$((\alpha \vee \beta) \wedge (\alpha \vee \gamma))$	distributivity of $\vee$ over $\wedge$

Use any to convert sentence into CNF form

Approach to convert FOL into CNF

1. Convert to negation normal form
   a. Eliminate implications & equivalence
   b. Move NOTs inwards using De Morgan
2. Standardize variables
3. Skolemization (to eliminate all existential quantifiers)
4. Drop all universal quantifiers
5. Distribute ORs inwards over ANDs

1. Convert to negation normal form

Eliminate implications & equivalence:

convert $p \rightarrow q$ to $\neg p \vee q$ and

convert $p \leftrightarrow q$ to $(p \vee \neg q) \wedge (\neg p \vee q)$

Approach #1.1: Implication elimination

Eg "Any mothers who love all babies, are also loved by some mothers"

$$\forall x\ \big(\forall y\ Babies(y) \rightarrow Loves(x,y)\big) \rightarrow (\exists y\ Loves(y,x))$$

$$\forall x\ \big(\forall y\ \neg Babies(y) \vee Loves(x,y)\big) \rightarrow (\exists y\ Loves(y,x))$$

$$\forall x\ \neg\big(\forall y\ \neg Babies(y) \vee Loves(x,y)\big) \vee (\exists y\ Loves(y,x))$$

1. Convert to negation normal form

De Morgan Laws:

convert $\neg(p \vee q)$ to $(\neg p) \wedge (\neg q)$;
convert $\neg(p \wedge q)$ to $(\neg p) \vee (\neg q)$;
convert $\neg\neg p$ to p;
convert $\neg(\forall x\, p(x))$ to $\exists x\, \neg p(x)$;
convert $\neg(\exists x\, p(x))$ to $\forall x\, \neg p(x)$

Approach #1.2: Der Morgan Laws

Eg "Any mothers who love all babies, are also loved by some mothers"

$$\forall x\, \neg\big(\forall y\, \neg Babies(y) \vee Loves(x,y)\big) \vee (\exists y\, Loves(y,x))$$
$$\forall x\, \big(\exists y\, \neg\, (\neg\, Babies(y) \vee Loves(x,y)\big) \vee (\exists y\, Loves(y,x))$$
$$\forall x\, \big(\exists y\, \neg\neg\, Babies(y) \wedge \neg Loves(x,y)\big) \vee \big(\exists y\, Loves(y,x)\big)$$
$$\forall x\, \big(\exists y\;\, Babies(y) \wedge \neg Loves(x,y)\big) \vee \big(\exists y\, Loves(y,x)\big)$$

2. Standardize variables

For FOL that use the similar variable name twice, change the name of ONE of the variables to avoid confusion

Approach #2.1: Standardize Variables

$\forall x\, p(x) \vee\ \exists x\, q(x)$ change into $\forall x\, p(x) \vee\ \exists y\, q(y)$ and
$\forall x\, p(x) \vee \forall y\, q(y)$ change into $\forall x\, p(x) \vee \forall x\, q(x)$

Eg "Any mothers who love all babies, are also loved by some mothers"

$$\forall x\, \big(\exists y\ \ Babies(y) \wedge \neg Loves(x,y)\big) \vee \big(\exists y\, Loves(y,x)\big)$$
$$\forall x\, \big(\exists y\ \ Babies(y) \wedge \neg Loves(x,y)\big) \vee \big(\exists z\, Loves(z,x)\big)$$

3. Skolemization (eliminate all ∃)

Approach #3.1: Skolemization

- Move all quantifiers to the left, without changing their relative positions.

  Convert $p \wedge (\forall x\, q(x))$ to $\forall x (p \wedge q(x))$;
  convert $p \vee (\forall x\, q(x))$ to $\forall x (p \vee q(x))$;
  convert $p \wedge (\exists x\, q(x))$ to $\exists x (p \wedge q(x))$;
  convert $p \vee (\exists x\, q(x))$ to $\exists x (p \vee q(x))$

- Use Skolem function
  - If $\exists x\, p(x)$ then just pick one; call it x'
  - If the existential quantifier is under control of a universal quantifier, then the picked value has to be a function of the universally quantified variable: $\forall x, \exists x, p(x, y)$ then change to $\forall x, p(x, y(x))$

Eg "Any mothers who love all babies, are also loved by some mothers"

$$\forall x \left(\exists y\ Babies(y) \wedge \neg Loves(x,y)\right) \vee \left(\exists z\ Loves(z,x)\right)$$
$$\forall x\, \exists z \left(\exists y\ Babies(y) \wedge \neg Loves(x,y)\right) \vee Loves(z,x)$$
$$\forall x\, \exists z\, \exists y \left(Babies(y) \wedge \neg Loves(x,y)\right) \vee Loves(z,x)$$
$$\forall x\ \exists y \left(Babies(y) \wedge \neg Loves(x,y)\right) \vee Loves(g(x),x)$$
$$\forall x\ \left(Babies(f(x)) \wedge \neg Loves(x,f(x))\right) \vee Loves(g(x),x)$$

4. Drop all universal quantifiers

By now, all the quantifiers have become universal ∀, therefore we can drop them because we can take for granted that all variables are universally quantified

Approach #4.1: Drop all universal quantifiers

Eg "Any mothers who love all babies, are also loved by some mothers"

$$\forall x \quad \big(Babies(f(x)) \wedge \neg Loves(x, f(x))\big) \vee Loves(g(x), x)$$

$$\big(Babies(f(x)) \wedge \neg Loves(x, f(x))\big) \vee Loves(g(x), x$$

5. Distribute ORs inwards over ANDs

Distribute Ors inwards over ANDs to create a conjunction of disjuncts.

- Replace $p \vee (q \wedge r)$ with $(p \vee q) \wedge (p \vee r)$

Eg "Any mothers who love all babies, are also loved by some mothers"

$$\left(Babies(f(x)) \wedge \neg Loves(x, f(x))\right) \vee Loves(g(x), x)$$

$$Babies(f(x)) \vee Loves(g(x), x)) \wedge (\neg Loves\left(x, f(x)\right) \vee Loves(g(x), x)$$

Unification

- A *substitution* σ unifies atomic sentences p and q if $p\sigma = q\sigma$

p	*q*	σ
Knows(John,x)	*Knows(John,Jane)*	*x/Jane*
Knows(John,x)	*Knows(y,Mary)*	*y/John,x/Mary*
Knows(John,x)	*Knows(y,Mother(y))*	*y/John,x/Mother(John)*

- Idea: Unify rule premises with known facts, apply unifier to conclusion
- E.g., if we know *q* and the rule: *Knows(John,x)* $\rightarrow$ *Likes(John,x)*, we conclude:
  - *Likes(John,Jane)*
  - *Likes(John,Mary)*
  - *Likes(John,Mother(John))*

Example 1: Resolution refutation

Given the axioms and its notation:

M: Mary studies late at night.

M → D: If Mary studies late at night, she feels drowsy in the morning.

D → F: If Mary feels drowsy in the morning, she drinks coffee.

~F: Mary doesn't drink coffee.

You are required to perform a refutation resolution to prove that "Mary studies late at night".

Solution:

1. Convert sentences into CNF and put into Knowledge Base (choose any approach #1-#5)
2. Negate goal and add this knowledge into Knowledge Base
3. Run proof tree until get NIL, otherwise the goal is false

Exercise 4: Exhaustive Search

2. Based on Figure 1, let node A is a start node, and the goal node is E. Perform a **breadth-first search**, then list down the order of nodes to be visited (OPEN and CLOSED list) from the starting node to the goal node in Table below.

Iteration	OPEN	CLOSED
0		
1		
2		

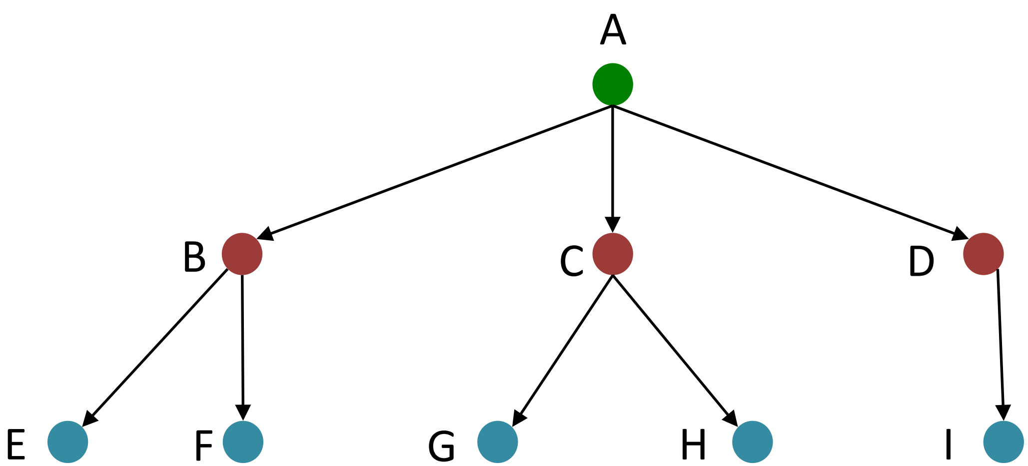


Figure 1. Gas piping network

3. Produce the sequence of nodes for the gas can be flowed from gate A to gate E.

Exercise 5: Exhaustive Search

1. Based on Figure 1, let node A is a start node, and the goal node is E. Perform a **depth-first search**, then list down the order of nodes to be visited (OPEN and CLOSED list) from the starting node to the goal node in Table below.

Iteration	OPEN	CLOSED
0		
1		
2		
3		
4		
5		

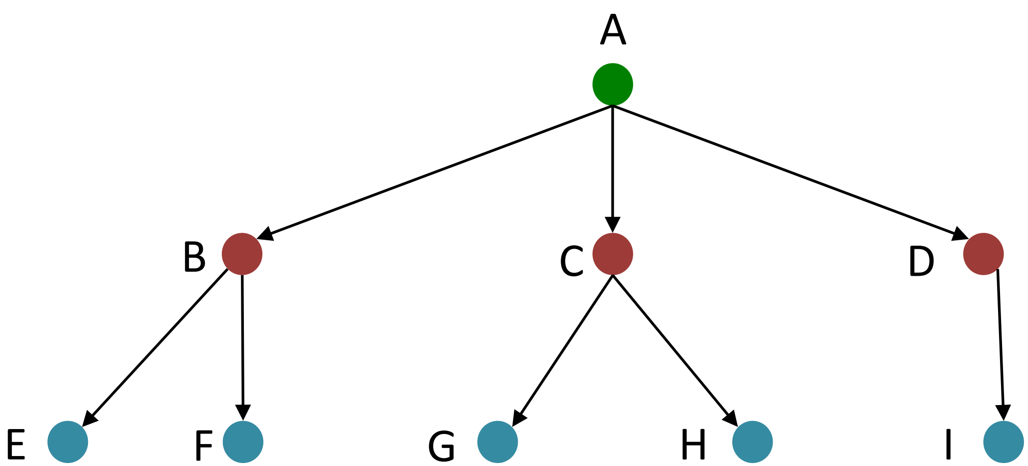


Figure 1. Gas piping network

2. Produce the sequence of nodes for the gas can be flowed from gate A to gate E.