Shiva Saran
BE-B → 20
AIR

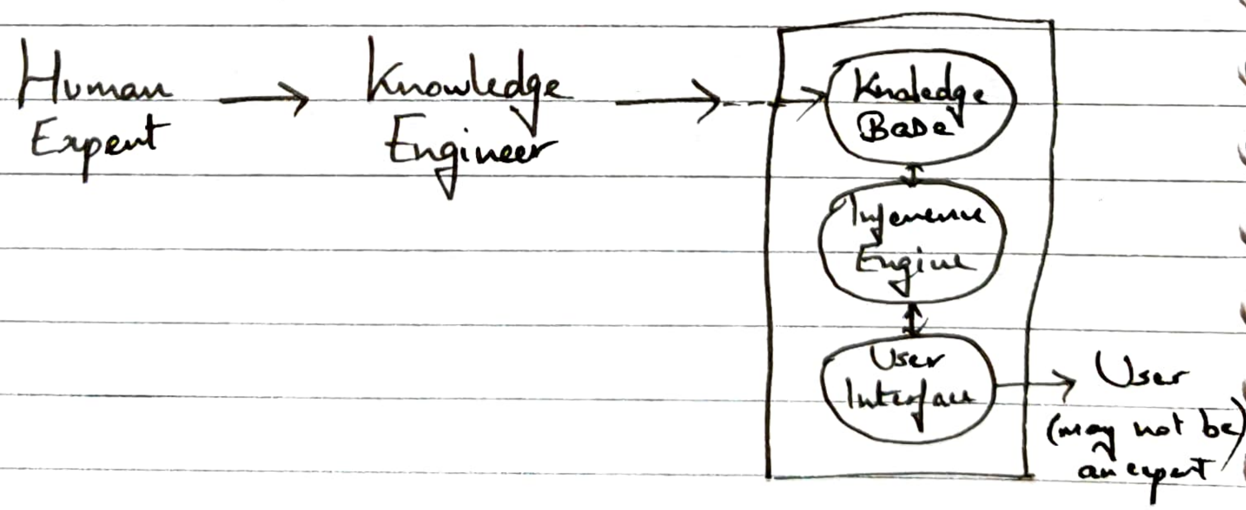
Date: __/__/____

* Assignment 2

* Question 1

1. Rule based expert system is also known as product systems of expert systems.
2. A rule based expert system is one whose knowledge base contains the domain knowledge coded in the form of rules.
3. Rules are used for representation of knowledge.
4. Rules are expressed as a set of if-then statements.
5. Rules tells what to do or what to conclude in different situation.

- Components of Expert System:



Date: __ / __ / ____

(i) Knowledge Base
It is a collection of rules or other information derived from the human expert.

(ii) Inference Engine
It is the main processing element of expert system.
· It chooses rules from agenda to fire.

(iii) User Interface
It is the method by which the expert system interacts with the user.

(iv) Working Memory
It contains the data that is received from the user during expert system session. Values in working memory are used to evaluate rules in the knowledge base.

• Advantages:
1. High reproducible.
2. Can be used in areas where human error is high.
3. If KB has correct info, low possibility of error.
4. Performance remains steady.
5. High speed response to a query.

Date: __ / __ / ____

- **Disadvantages / Limitations:**

1. Response of expert system ~~to~~ may get wrong if KB has less / wrong info.
2. Creative output not possible for different scenarios.
3. Maintenance & development costs are high.
4. Knowledge aquisition for designing is difficult.
5. For each domain, a specific ES is needed.
6. Can't learn itself, hence regular updates need.

* Question 2

Block World problem using STRIPS.

- It is known example to demonstrate planning using STRIPS (Standford Research Institute Problem Solver).
- Block world consists of:

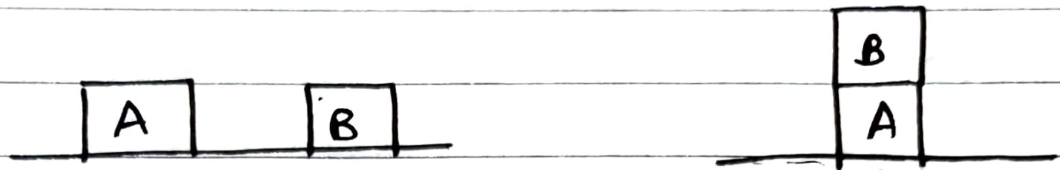
(i) A table

(ii) Identical blocks with unique letter.

(iii) Blocks can be put one on another to form a stack.

(iv) This stack is built with robot arm.

Eg. The arm can ~~pref~~ perform operations of lifting a single block at a time & placing it.



Initial State　　　　Goal state

- Predicates used here are:

On (A, table)
On (B, table) } Both blocks are on the table.

On (B, A) → B is on A.

Clear (A) → Block A has nothing on it.

Holding (B) → Robot's hand is holding block (B).

Empty_Hand : Robot's hand is not holding anything.

- State representation:-

1. On (A, table) ^ On (B, table) ^ Clear (A) ^ Clear (B) ^ Empty Hand → Initial Stage.
2. On (A, table) ^ Holding (B) ^ Clear (A)
3. On (A, table) ^ On (B, A) ^ Clear (B) ^ Empty_Hand (~)

* ~~Actoin~~ Actions:

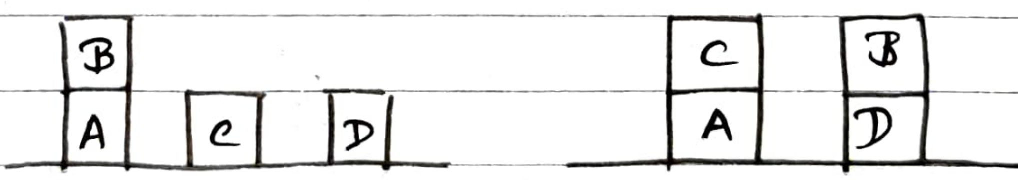
1. Unstack (B, A)
2. Stack (B, A)
3. Lift (B)
4. Place (B)

* Question 3

Goal Stack Planning

- Main problem with Backward State ~~pl~~ space search is goal regression is not sound & algorithm may be handling goal ~~dis~~ descriptions that are not consistent with any state.

- Basic idea of Goal stack planning is to handle interactive compound goals using goal stacks. Here the stack contains:
  - Goals
  - Operators — Add, Delete & Prerequiste lists
  - A database maintaining the ~~or~~ current sitation for each operator used.



Start | Goal

On (B,A) ∧ On (A, table) ∧ O(C, table) ∧ On (D, table) ∧ Empty_Hand.

On (C,A) ∧ On(B,D) ∧ on (A, table) ∧ On (D, table).

Date: __/__/____

The stack for goal will be:

On (C, A)

~~On (B, D)~~ On (B, D)

On (C, A) ^ on (B, D) ^ on (A, table) ^ On (D, table).

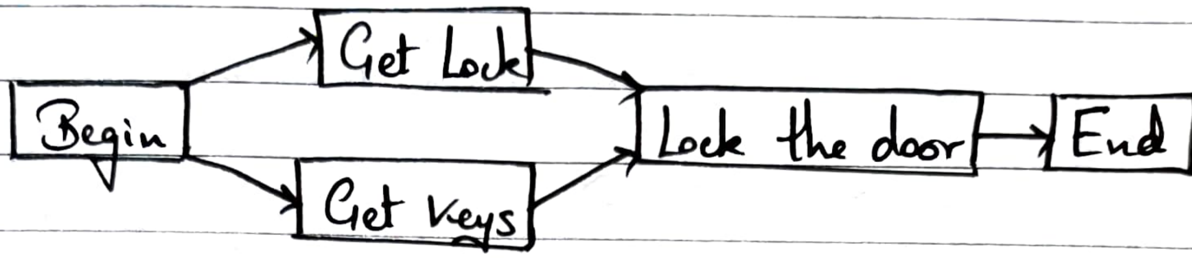
Here the last operation leads to final goal state where D & A alreed placed on table.

Question 4

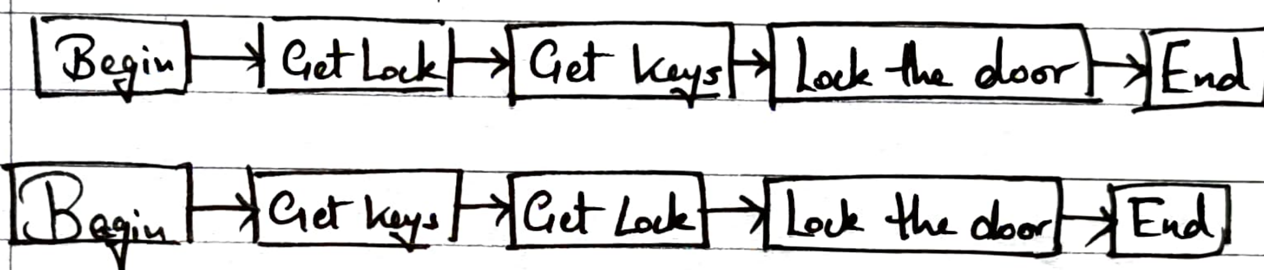
* Partial order planning

- The idea of partial order planning is to have a partial order between actions & only commit to an ordering between actions when forced.
- This is sometimes also called as non-linear planner which is misnomer because such planners often produce a linear plan.
- A partial ordering is a less-than relation that is transitive & asymmetric.
- A partial-order plan is a set of actions together with a partial ordering, representing a before relation on actions, such ~~as~~ that any total ordering of the actions, consistent with the partial ordering which will solve the goal from the initial state.

Eg.



(a) Partial order planning



(b) Total Order Planning.

Question 5

* Constraint Statisfaction Problem

- Constraints are the natural medium for people to express problems in many fields.
- Many real problems in AI can be modled as constraint satisfaction problem and are solved through search.
- Eg. of constraint
  The sum of 3 angles in triangle.

Date: __ /__ / ____

- Constraint is a logical relation among variables.
- Constraint satisfaction is a process of ~~dividing~~ finding a solution to a set of ~~problems~~ constraints.

Types of constraints:

① Unary — involves a single variable
Eg. South Africa ≠ green.

② Binary — involves a pair of variables
Eg. South Africa ≠ Washington.

③ Higher Order Constraints — Involves 3 or more variables.
Eg. Professors A, B, C can't be on committee.

④ Preference (Soft) Constraints)

Eg. Suppose you have CSP with variables A, B, C each with domain (1, 2, 3, 4). Suppose the contraints are $A < B$ & $B < C$.

→ Variables : {A, B, C}
Domain : {1, 2, 3, 4}
Constraint : ① $A < B$
② $B < C$