Artificial inteligence

Date: / / Page no:

Unit → 1

Q-1 overview of AI

AI usually define as science of making computer do things that required intelligence done by humain

(II) AI is a branch of science which deal with helping machine, find solution to complex problem. in humain like fashion

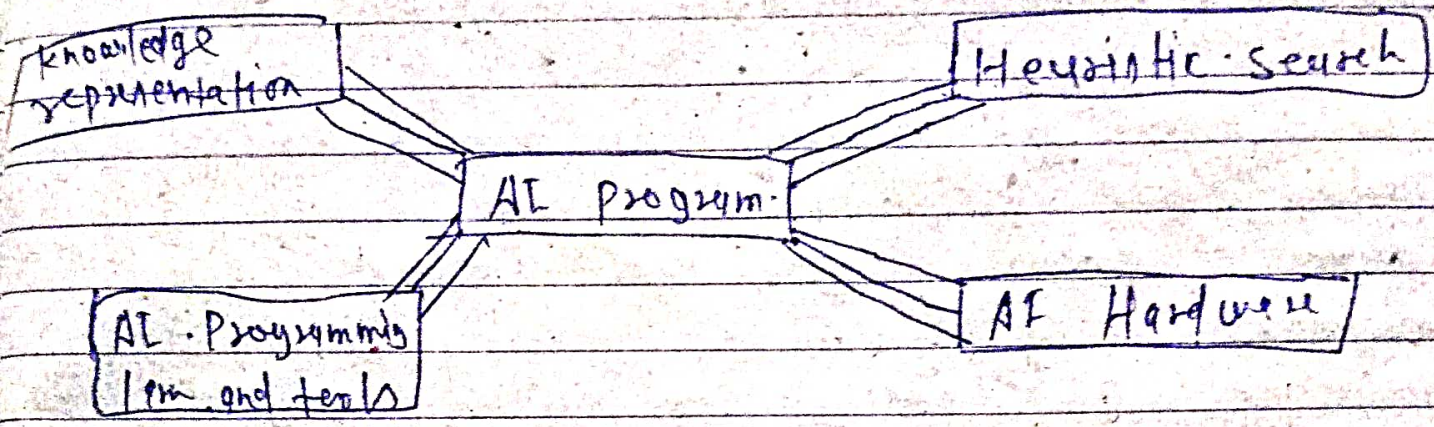
(III) AI is composed with two wordels AI Artificial and intelligence. where Artificial define "man made" and intelligence define "thinking power" hence AI means man made thinking power.

(4) Application of AI

(I) game playing
(II) speech recognization
(III) Hand writting recog.
(IV) Natural lang. process.
(V) computer vision
(VI) Intelligent robot
(VII) Expert system
(VIII) classification
(IX) in Astronomy
(X) in Healthcare
(XI) in finance, Data security
(XII) in social media
(XIII) in transport.

Date / / Page no.

Q2 Component of AI



Q.3 Lisp ⇒ List processing

(I) John mccarthy discover LISP in 1958-59. Shortly after developing FORTRAN

(II) It was implemented by Steve Russell on an IBM 704 Computer.

(III) It is a second oldest high level programming language in the world.

(IV) It is suitable for AI programms. as it process symbolic information effectively

feature of lisp →

(I) It is a machine dependent language.
(II) It allow updating the program
(III) It provide High level debugging
(IV) It support object oriented programming.
(V) It support all type of data type, such as tree, hash table, array, vector, list, set, object.
(VI) It support decision making stmt like if, when, case
(VII) It also support iterating stmt like do, loop.
(VIII) By using lisp we can also create our own function
(IX) It support input output functions.

Date: / / Page no:

Application build in CLISP →

(i) CN2
(ii) Yahoo store
(iii) Auto card
(iv) Emacs

Expression → lisp expression are called symbolic expression or S-expression.

- the Sexpression are composed of three object list, string and atoms.

LISP program can run on interpretor or compiler

Hello world program in lisp

Syntax

(write-line 'string')

(write-line "Hello world")

Q.4 Basic Building Block

There are 3 Basic Building Block

(I) Atom → An atom is a number or string in contigeous character. It includes numbers and special characters.

ex name , 12345 , *Hellow* , BCD*123

(II) List → A list is a sequence of Atoms and list closed in paranthesis ().

ex (I am a list)
(a b (c de) fgH)

(III) string → String is a group of characters enclosed in double ~~closed~~ quotation mark ("...").

ex "I am string"
"a, ba c * # %."

Q.5 LISP Data types

Date: / / Page no:

LISP data types can be categories as

① scaler type → Number, character, symbol

② Data structure → list, vector, string

① It is not necessary to define a data type for a LISP variable.

② Any variable can take any LISP object as its value.

type of → function return the type of a object.

ex (setq x 10)
(setq y 3.12)
(setq ch nil)
(setq n 123.78)
(setq bg 11.0e+4)
~~(setq)~~
(setq z 64/2)

(print x)
(print y)
(print ch)

Date: / / Page no:

```
(print n)
(print bg)
(print r)
```

output →

```
10
3.12
NIL
123.78
110000.0
32
```

Q.6 LISP manipulating function

(I) car → It taken a list as argument and return the first element.

(II) cdr → It taken a list or argument and return the list except first element.

(III) conn → this function will take two argument one in list and a value. and insert the value in the list at first place. and return the new list.

(III) list → It taken any number of arguments and return a list with the argument an member of list.

(IV) append → It merge two or more list into one.

(V) last → this function return a list that contain a last element.

(VI) member → this function is used to return the list based on member value.

Date / / Page no

gt takes two argument member value and list.
If member value present in the list then it will return the list else it return NIL.

(vii) reverse → gt takes a list and return it in reverse order.

program..

```
(write (car '(a b c d e f)))
(terpri)
(write (cdr '(a b c d e f)))
(terpri)

(write (cons 'a '(b c)))
(terpri)

(write (list 'a '(bc) (ef)))
(terpri)

(write (append '(A B) '(c d) '(e f)))
(terpri)

(write (last '(a b c d (ef))))
(terpri)
```

Date: / / Page no:

```
(write (member 'a' (a b c d)))
 (terpri)
(write (reverse '(a b c d e)))
```

Output →

```
a
b c d ef
a b c
(a (BC) (ef))
(A B c d ef)
(e f)
(a, b, c d)
(e d c b a)
```

Date: / / Page no:

Q.7

predicates (P) → Predicates are the functions that test there argument.

If the condition is false gt return NIL and if the condition is true gt will return Non-NIL value true.

(I) atom:
(II) equal:
(III) evenp:
(IV) OddP
(V) zeroP:
(VI) NULL:
(VII) listp:
(VIII) greaterP:
(IX) lessp:
(X) NumberP:
(XI) symbolP:
(XII) IntegerP:
(XIII) floatP:
(XIV) characterP:
(XV) StringP:
(XVI) arrayp:
(XVII)
(XVIII)

example

(write (atom 'abcd)) → True
(terpri)
(write (equal 'a 'b)) → NIL
(terpri)
(write (evenp '10)) → True
(terpri)
(write (oddp '5)) → true.
(terpri)
(write (zerop 0.0001)) → NIL
(terpri)
(write (integerp '10)) → True.

Q.8 Property list

Date: / / Page no:

① A property list consist of entries where every entries consist of a key called Indicator and a value called property.

for example, we have a object person we would like this person object to have properties like 'name', height, weight, address, age etc.

when a symbol is created initially its property list is empty.

Various property list function

① Get function →

Syntax → get symbol Indicator

get search the property list to find an indicator equivalent to indicator

If found value is returned or default value is returned.

(write (self (get 'book' title) (AI)))

Date: / / Page no.

(II) self function → It is used with getting to create new property value pair.

Syntax self ((get function) property / value)

(III) remprop → remprop function is used to remove the property equallent to the indicator

Syntax → remprop Symbol indicator

(IV) getf →

getf search the plist stored in place for indicator equivalent to an indicator. If one is found then the corresponding value is returned otherwise default is returned.

Syntax → get place indicator.

(V) remf → remf remove the property equivalent to the given indicator for the given place.

Syntax

remf place indicator.

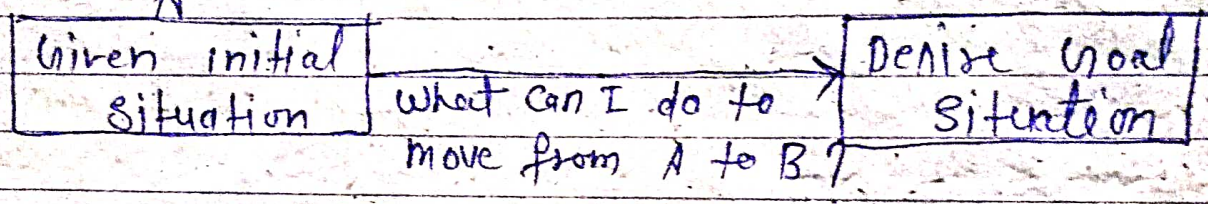
Unit → 2

Problem Solving

Date: / / Page no:

Q.1 problem solving in AI...

problem solving in AI may be characterized as a systematic search through a range of possible action in order to reach the goal or solution.



- problem solving in AI is used to achive objective or resolve a particular ~~st~~ situation.

problem solving is a process of designing and carriging out a set of step to reach the goal

problems like

(i) chess
(ii) N-queen
(iii) water-Jug problem.

Date / / Page no.

Q.2

Search and Control Strategies.

State space Search → Set of all possible states.

Control strategies →

Control strategy by which we comes to know, which rule is to be applied next during the process of reaching the solution to a problem

Control strategies requirement

(i) It should be cause motion
(ii) It should be systematic.

Q.3 water jug problem

Date: / / Page no:

(4, 3)

(1) fill 4 gallon jug $(x, y) \rightarrow (4, y)$ $x < 4$

(2) fill 3 gallon jug $(x, y) \rightarrow (x, 3)$ $y < 3$

(3) Empty 4 gallon jug $(x, y) \rightarrow (0, y)$ $x > 0$

(4) Empty 3 gallon jug $(x, y) \rightarrow (x, 0)$ $y > 0$

(5) pour water from 3 gall. jug to fill four gall. jug $(x, y) \rightarrow (4, y-(4-x))$ $0 < x+y \geq$ and $y > 0$

(6) pour water from 4 gall. jug to fill 3 gall. jug $(x, y) \rightarrow (x-(3-y), 3)$ $0 < x+y \leq 3$ and $y > 0$

(7) pour all water from 3 gal jug into 4 gall jug. $(x, y) \rightarrow (x+y, 0)$ $0 < x+y \leq 4$ and $y \geq 0$

(8) pour all water from 4 gall. jug into 3 gall jug. $(x, y) \rightarrow (0, x+y)$ $0 < x+y \leq 3$ and $x \geq 0$

Date: / / Page no:

Result →

x	y	Rule
0	0	
0	3	2
3	0	7
3	3	2
4	2	5
0	2	3

Ans

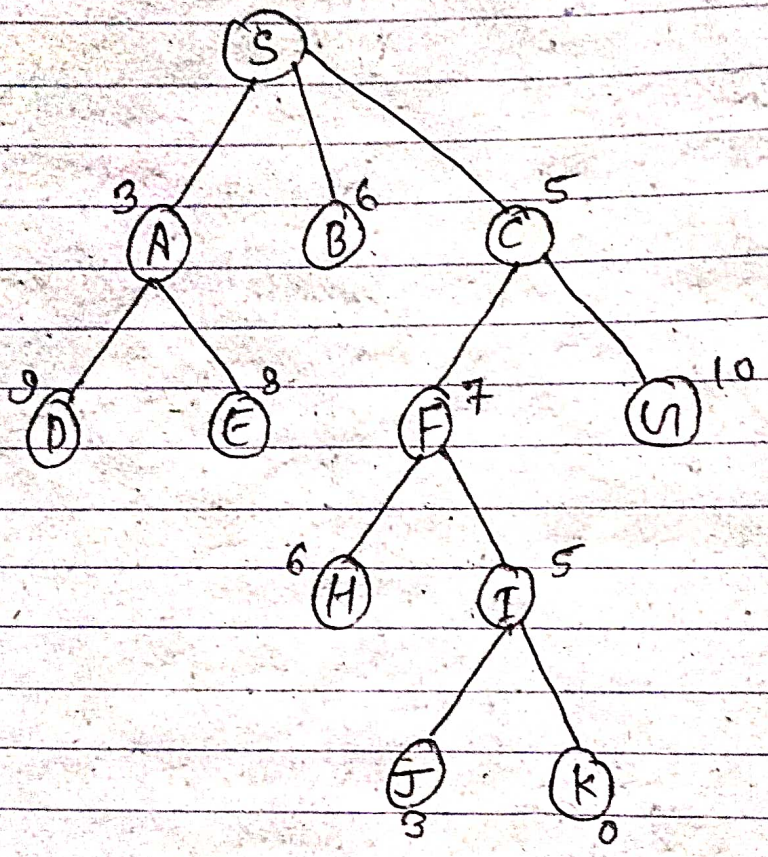
Date: / / Page no:

step No	Node being expended	children	Available Node	selected Node
1	S	(A:3) (B:6) (C:5)	(A:3) (B:6) (C:5)	(A:3)
2.	A	(D:9) (E:8)	(B:6) (C:5) (D:9) (E:8)	(C:5)
3	C	(F:7) (G:10)	(B:6) (D:9) (E:8) (F:7) (G:10)	(B:6)
4	B	—	(D:9) (E:8) (F:7) (G:10)	(F:7)
5	F	(H:6) (I:5)	(D:9) (E:8) (G:10) (H:6) (I:5)	(I:5)
6	I	(J:3) (K:0)	(D:9) (E:8) (G:10) (H:6) (J:3) (K:0)	(K:0) ↑ Goal Search Stop.

Q.4

Date: / / Page no:

Best first search



Q.5 Depth first search & BFS

BFS → It is used to find the shortest path in the graph.

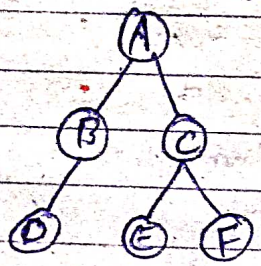
It uses queue data structure that follows first in first out.

It is slower then DFS

It is a vertex based technique.

required more memory

ex



output ⇒ A, B, C, D, E, F

DFS → It is an edge based technique

It uses stack data structure.

It follows last in first out.

It is suitable when the solution is away from the source.

It is faster than BFS

DFS uses Backtracking

required less memory

we may trapped in infinite loop.

Q.6 A* Algorithm

Date: / / Page no:

A* Algorithm is one of the best and popular technique in path finding and graph traversals.

f(N) = function ~~for~~ used for estimating least cost.

g(N) = estimate least cost from start to n

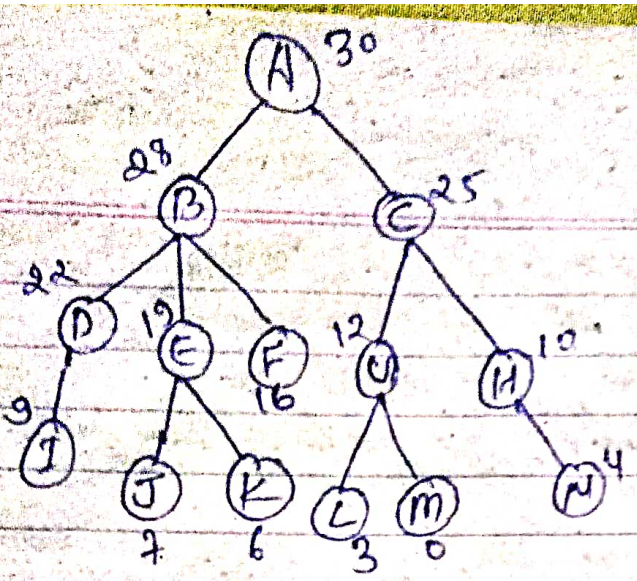
h(N) = ~~least~~ estimate least cost from n to goal

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

It is an inform search algorithm.

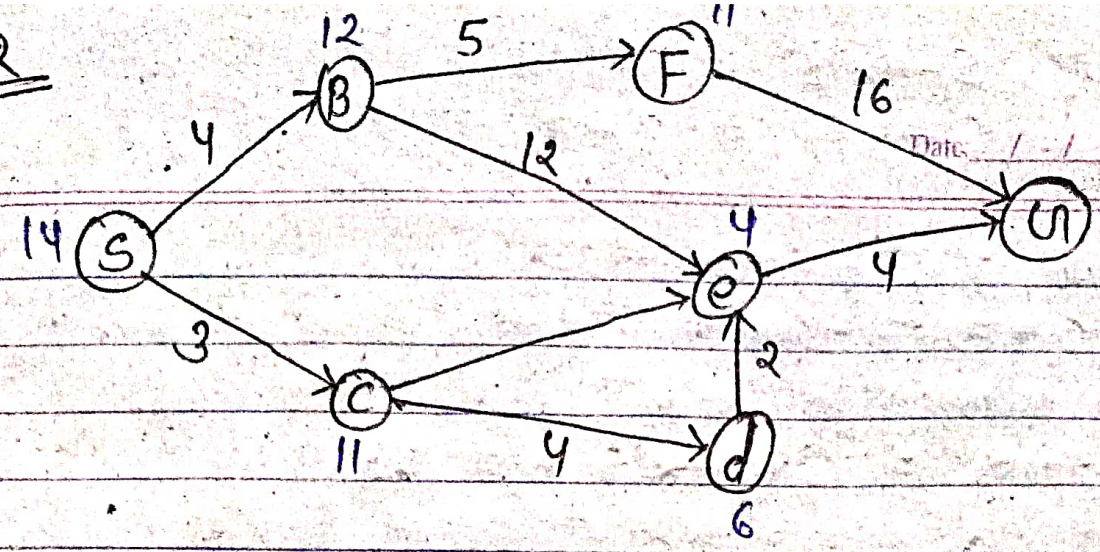
A* algorithm is a Breath first search algorithm in which a cost is associated with unknown node.

example



Node selected	openlist	clone list
A	A	empty list
C	BC	A
H	BGH	AC
N	BGN	ACH
G	BG	ACHN
m	BLm	ACHNG
L	BL	ACHNGm
B	B	ACHNGmL
F	DEF	ACHNGmLB
E	DE	ACHNGmLBF
K	DJK	ACHNGmLBFE
J	DJ	ACHNGmLBFEK
D	D	ACHNGmLBFEKJ
I	I	ACHNGmLBFEKJD
		ACHNGmLBFEKJDI

Q.2



Node selected	openlist	close list
S	S	empty
C	B C	S
D	B D E	S C
E	B E	S C d
B	B	S C d e
F	F	S C d e b F

Best path

S C d e

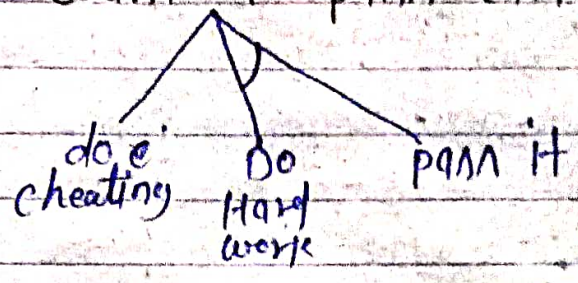
3 + 4 + 6 + 2 + 4 = 19

Q.7 AO* Algorithm

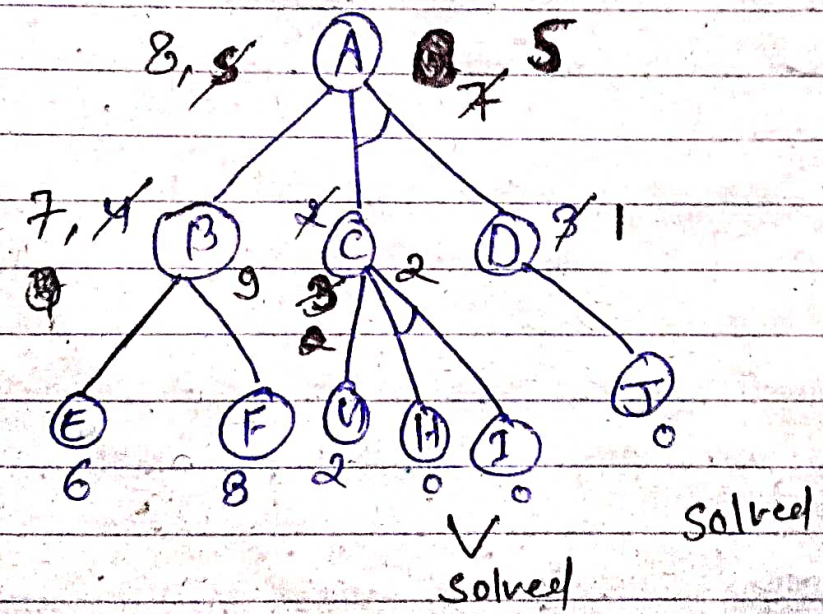
Date / / Page no.

It is a ~~hue~~ heuristic search Algo- in AI. It uses the concept of AND-OR graph to ~~decom~~ ~~solve~~ decomplish any complex problem.

Q. ⇒ want to pass exam



AO* does not explore all the solution paths once it get a solution.



Solved graph / tree

Date: / / Page no:

Step 1 ⇒

$f(A-B) = g(B) + h(B) = 1+4 = 5$

$f(A-C-D) = g(C) + h(C) + g(D) + h(D)$

$1+2+1+3 = 7$

Step 2 ⇒

$f(B-E) = g(E) + h(E) = 1+6 = 7$

$f(B-F) = 1+8 = 9$

$f(A-B) = g(B) + \text{updated } h(B) = 1+7 = 8$

Step 3 ⇒

$f(C-G) = 1+2 = 3$

$f(C-H-I) = 1+0+1+0 = 2$

Step 4 ⇒

$f(D-J) = 1+0 = 1$

$f(A-C-D) = 1+2+1+1 = 5$

Finally solved graph value are. (A = 5)

A = 5	D = 1	G = 0
B = 7	E = 6	H = 0
C = 2	F = 8	I = 0

Q.8 Heuristic search

Date: / / Page no:

(i) A heuristic in a technique that is used to solve a problem faster than the other classic method.

(ii) It is a technique that improve the efficiency of search process.

(iii) The purpose of heuristic search is to guide the search process. to find the most efficient path among all the path.

(iv) Heuristic function

f(n) = function used for estimating least cost

g(N) = estimate. least cost from start to current node n.

h(n) = estimate the least cost from current node to goal.

Q.2

Hill climbing Algo.

(i) It is also called optimization algorithm that uses simple heuristic function

(ii) the amount of distance node from the goal

(iii) Infact there is ~~perticular~~ practically no difference between Hill climbing and depth first search. except that the children of the node has been updated.

(iv) It is a technique for optimizing the mathematical problem.

(v) Hill climbing is widely used when a good heuristic is available.

- problem with Hill climbing

(i) local maxima / maximum
(ii) Plateau
(iii) Ridge.

Date: / / Page no:

(i) Local maxima → It is a state that is better than all its neighbours but it is not better then some other state farther away.

(ii) Plateau ⇒ It is a condition in which all neighbour have the same value. it is not possible to determine the best direction.

(iii) Ridge ⇒ the algorithm stop when it reaches this state because any point on a ridge can look like a peak because movement in all direction is downward.

this is an area in a path which must be traversed v. carefully.

Q.10 Constraint Satisfaction problem

CSP Consist of three component V, D, C.

V is a set of variable {V_1, V_2, V_3, V_n}

D is a set of Domain {$D_1, D_2, D_3 - - - D_n$}

C is a set of constraints that specify allowable combination of value.

C = {scope, rel}

Scope → scope is a set of variable that participate in constraints.

rel → rel is a relation that define the value that variable can take.

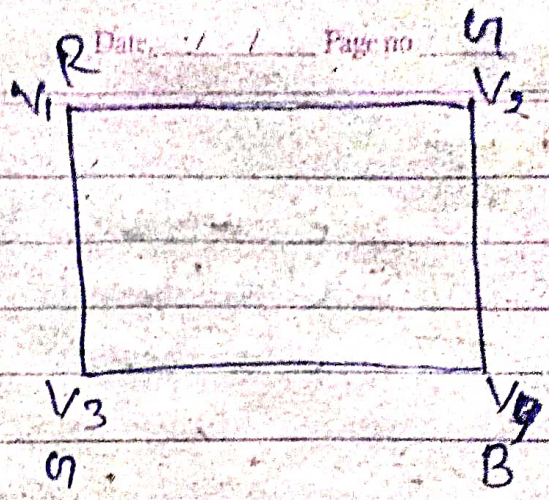
{C_1, C_2, C_3} → Rule

Algo. for CSP

$V_0 = \{ V_1, V_2, V_3, V_4 \}$

$D = \{ Red, Green, Blue \}$

$C = \{ 1 \neq 2, 1 \neq 3, 1 \neq 4, 2 \neq 4, 3 \neq 4 \}$



1	2	3	4
R, G, B	R, G, B	R, G, B	R, G, B
R	G, B	G, B	G, B
R	G	G, B	B
R	G	G	B

Q.11 Crypt Arithmetic problem Page no: Date:

```
  S E N D
  M O R E
---------
M O N E Y
```

```
  9 5 6 7
  1 0 8 5
---------
1 0 6 5 2
```

10652

```
  C R O S S
  R O A D S
-----------
D A N G E R
```

```
  9 6 2 3 3
  6 2 5 1 3
-----------
1 5 8 7 4 6
```

```
  D O N A L D
  G E R A L D
-------------
R O B E R T
```

```
  5 2 6 4 8 5
  1 9 7 4 8 5
-------------
7 2 3 9 7 0
```

```
  F O R T Y
    T E N
    T E N
---------
S I X T Y
```

```
  2 9 7 8 6
    8 5 0
  - 8 5 0
---------
  3 1 4 8 6
```

unit → 5

Date / / Page no

Q.1 Bayes theorem

(I) Bayes theorem is also known as Bayes Rule, Bayes law or Bayes reasoning.

(II) which determine the probability of an event ~~with uncertain knowledge~~ based on prior knowledge that might be related to event.

(III) the Bayes theorem was named after the British mathematician thomas Bayes.

(IV) It is a way of calculating the value of ~~P(A|B)~~ P(B|A) with the help of knowledge P(A|B)

formula →

$$P(A) = \frac{\text{Number of favorable outcome}}{\text{total number of fav. outcome}}$$

Date: / / Page no:

Now

$$P(A \cap B) = P(A|B)\,P(B) \quad \text{---} \quad (1)$$

$$P(A \cap B) = P(B|A)\,P(A) \quad \text{---} \quad (2)$$

from equation (1) and (2)

$$P(A|B)\,P(B) = P(B|A)\,P(A)$$

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

or

$$P(B|A) = \frac{P(A|B)\,P(B)}{P(A)}$$

Q2 Bayesian Network

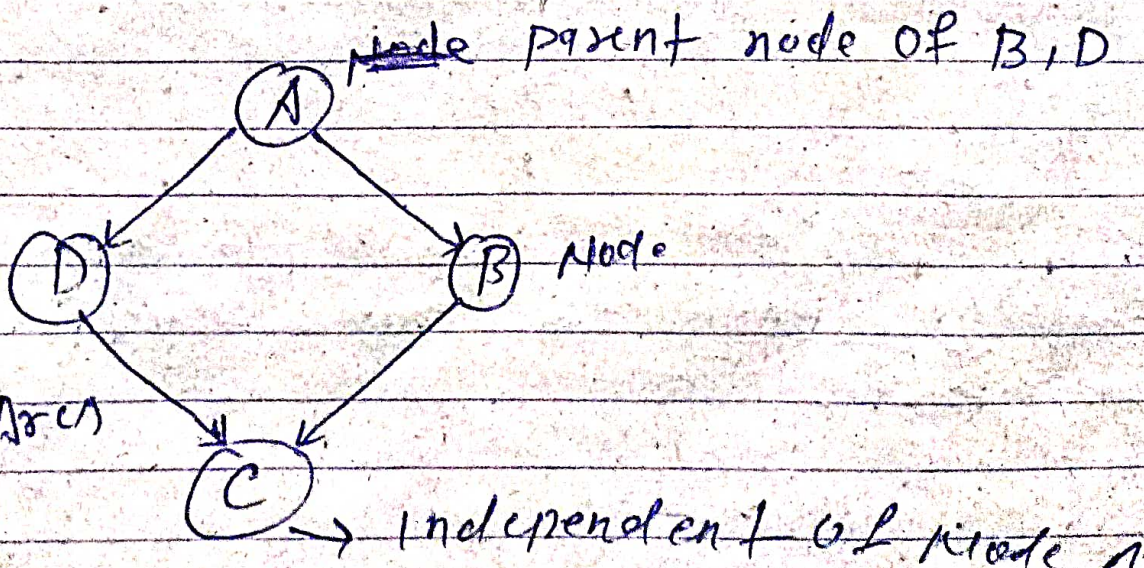
Date. / / Page no.

(I) A Bayesian network is a ~~probabilistic~~ graphical model which represent a set of variable and their conditional dependencies using a graph.

(II) It is also called Bayes network, belif network and decision network.

(III) these network are build from probability distribution and also use probability theory for prediction.

(IV) A Bayesian network is made of nodes and ~~arcs~~ arcs.



Date: / / Page no:

Q.3 Expert System

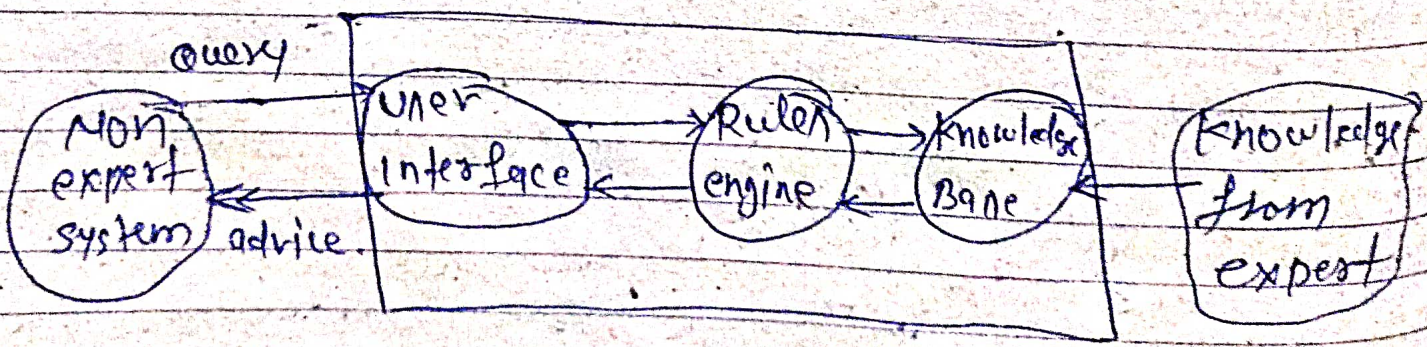
(i) An expert system is a computer program that is designed to solve complex problem and to provide decision making ability like a human.

(ii) It perform this by extracting knowledge from its knowledge base

(iii) It was developed in 1970.

(iv) the performance of expert system is based on its knowledge stored in knowledge base.

Architecture



Date: / / Page no:

(1) User interface → with the help of user interface the user can interact with expert system. with the help of UI the expert sytem takes query as an input and pass it to the rule engine.

(11) Rule of Engine → It is known as the brain of expert system. It is the main processing unit of the expert system. with the help of Engine the system extract the knowledge from knowledge base.

(111) Knowledge base → the knowledge base is a type of storage that stores knowledge.

It is similar to database that contain information and rule of perticular domain, or subject.

Application

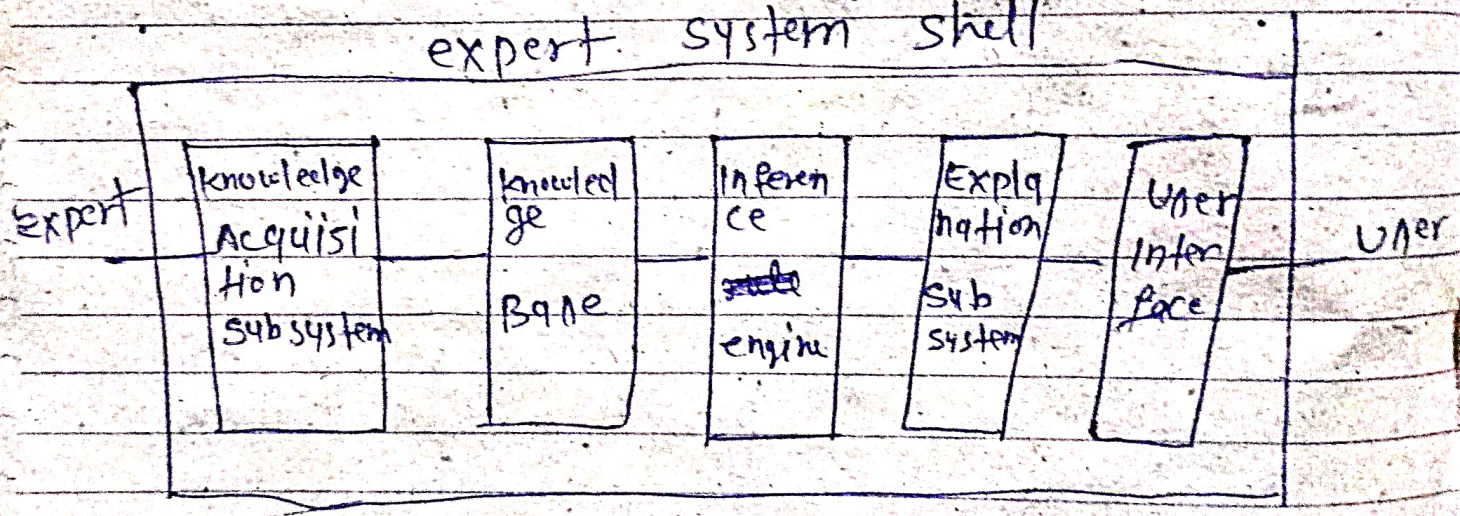
(1) designing and manufacturing domain
(11) the knowledge domain.
(111) the finance domain.
(1v) planning and scheduling

Q.4 Expert system shell

Date: / / Page no:

(I) An Expert system shell is a special purpose tool, designed based on the requirement of a particular application.

(II) It contains the basic component of expert system.



Components of expert system shell

(I) knowledge Acquisition subsystem
(II) knowledge Base
(III) inference rule engine.
(IV) Explanation subsystem
(V) User Interface.

Date: / / Page no:

① Knowledge Acquisition subsystem →

It is a subsystem that helps the expert to build the knowledge Base.
It collect the knowledge from different experts and store into knowledge base using knowledge representation language.

② knowledge base → It is a collection of factual and heuristic knowledge.

③ ~~Interf~~ inference Engine → It is used for manipulating symbolic information and knowledge in knowledge base.

④ Explanation subsystem → this explanation subsystem explain the solution or result to the user.

⑤ User interface → It provide the interaction b/w System and user.

Q.3 Learning

(I) learning in an application of AI that provide system the ability to automatically learn and improve its performance through its experience.

(1) Rote learning in AI →

(i) Rote learning in a memorization technique

(ii) A model simple memorize the information without understanding the context

(iii) in rote learning we store the computed data so that we do not have to recompute it again.

(iv) It is a simplest type of learning. without any modification. simply data copied into knowledge base.

(II) learning by induction in AI

(1) induction learning is a type of machine learning that is used to identify the patter

in data.

the induction learning algo. create a model based on a set of training data that are used to predict the new data.

(iii) Explanation - Based learning →

(1) EBL is a type of machine learning where a computer or AI system learn ~~by understanding~~ ~~to~~ from previously solved problem.

It is a problem solving method.

Q.6 Knowledge Acquisition Date: / / Page no:

(I) Knowledge Acquisition is a process of extracting knowledge from data.

(II) this can be done manually or using various technique such as machine learning.

there are many diff. method of knowledge Acquisition.

(I) rule Based system
(II) Decision tree.
(III) Artificial ~~Neural~~ neural network
(IV) fuzzy

(I) Rule Based system → Rule Based system uses a set of rules or heuristics to make decision.

(II) Decision tree → It is another method of knowledge Acquisition which uses IF-Then-else statement to make decision.

Date: / / Page no:

(iii) ANN → the ANN learn like the way Human brain learn. they learn the data and make decision Based on data.

(iv) FUZZY → It uses fuzzy set theory to make decision.

Task in K.N.

(i) collect → Collecting knowledge from expert.

(ii) Interprit → Identifying key point.

(iii) Analyze → forming Strategies to solve problem.

(iv) designing → forming better understanding of problem.

Q.7 Mysin

Date: / / Page no:

(I) Mysin is a rule-Based expert system, which is designed for diagnosing infection and selecting the Right antibiotic..

(II) It is a computer program that was developed in 1970's at Stenford University.

(III) Mysin was written in LISP programming language.

(IV) Firstly mysin ask the question about patient and ~~diagnose~~ then comparing the answer to database to know the infection.

Unit → 4

Q.1 Natural language processing.

(I) NLP stand for Natural language processing.

(II) NLP is one of the most interesting application of AI.

(III) In NLP we develop programs that can understand the natural language.

(IV) In NLP we use Natural language to instruct computer to perform any task.

(V) Naturally we have to use translater that convert natural language into machine language.

Natural language → NLP → machine language.

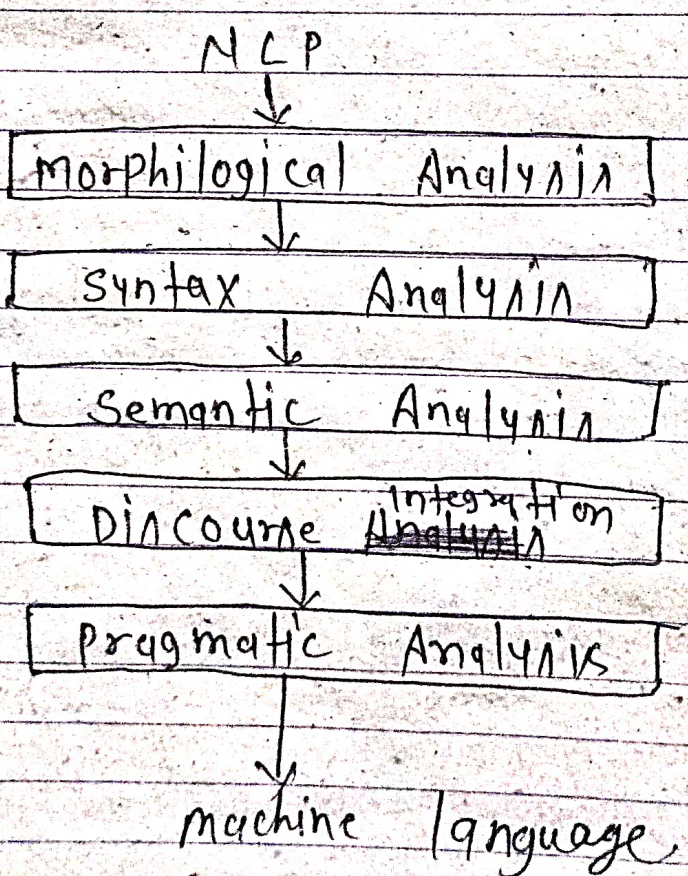
Date: / / Page no:

working of NLP

the working of NLP is similar to the working of compiler.

first it checks validity of ~~and~~ words and then check the syntax of sentence.

it has five phase.



Date: / / Page no:

① morhilogical Analysis → ① It assume program as collection of words.
② It divide the paragraph into words and check their validity.
③ It uses hash table.

② Syntax Analysis → ① It assume program as collection of sentence ② It devide the paragraph into sentences. and check their grammer.

③ Semantic Analysis → It is used to check weathe the ~~given~~ sentente of paragraph is meaningfull or not.

④ Discourse Intyration → Discouree integration depends upon the sentences that proceeds it.

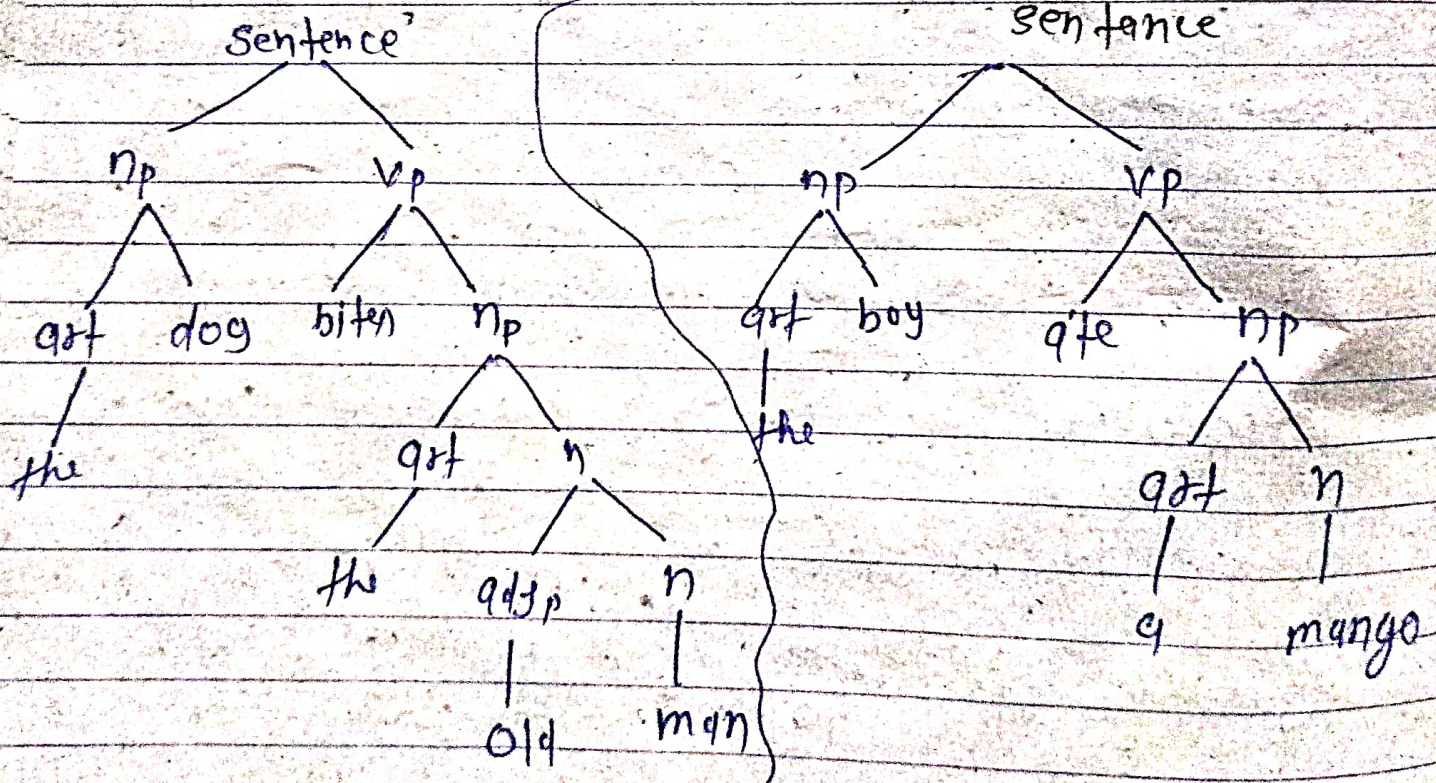
⑤ Pragmatic Analysis → Its perpose it to get the meaning of thone sentences which are not clear.

Q.2 Parsing techniques: Date: / / Page no:

these are five rule.

(I) Sentence → np vp
(II) np → n
(III) np → art n
(IV) vp → v
(V) vp → v np

ex The Dog bites the old men.



Q.2 the boy ~~art~~ ate a mango.

0.3 Game playing

Date: / / Page no:

(I) Game playing is an important domain of AI.

(II) Game does not required much knowledge the only knowledge we need is the rule, legal moves and the winning and loosing condition of a game.

(III) Searching techniques likes BFS (Breath first search) are not accurate for this. as the branching factor is very High.

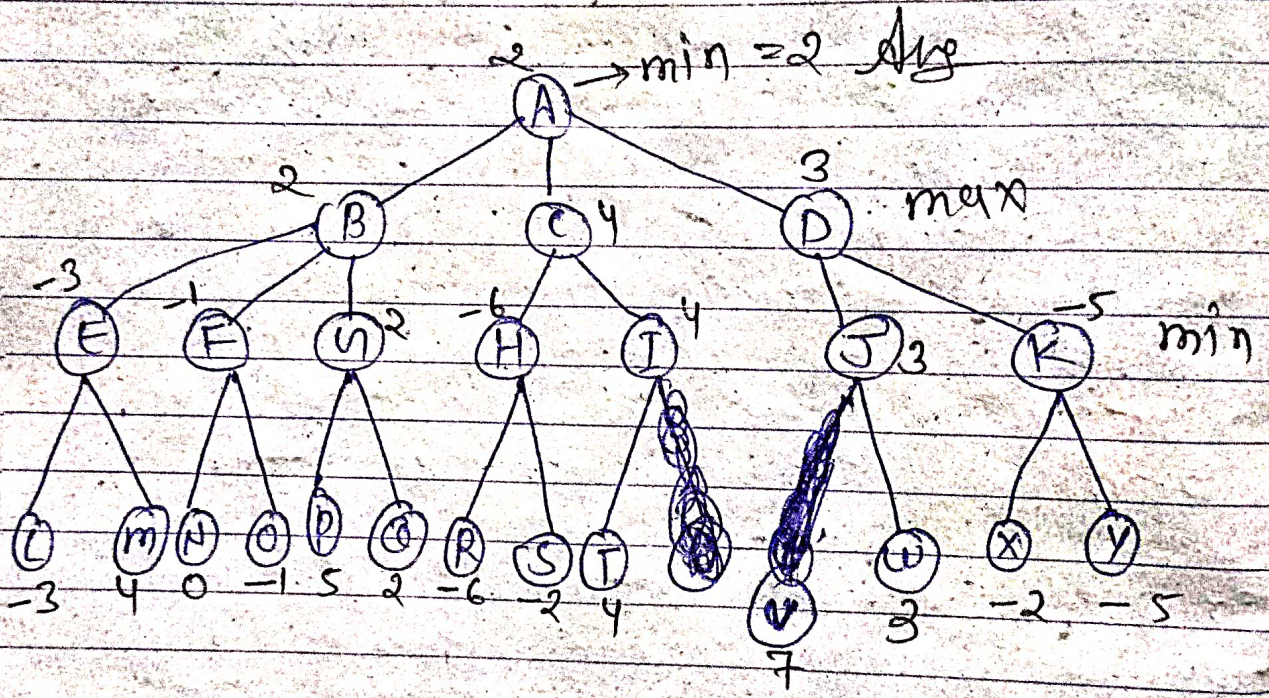
(IV) So searching will take lot of time. so we need another search algorithm.

(I) minimax Algorithm
(II) alpha beta cutoff / Alpha-Beta pruning.

minimax → is a recursive or backtracking algorithm which is used in game playing and decision making.

Q.4 min max Algorithm.

(I) It is a backtracking algorithm.
(II) Best move strategy used.
(III) max will try to maximize its utility.
(IV) min will try to minimize its utility.

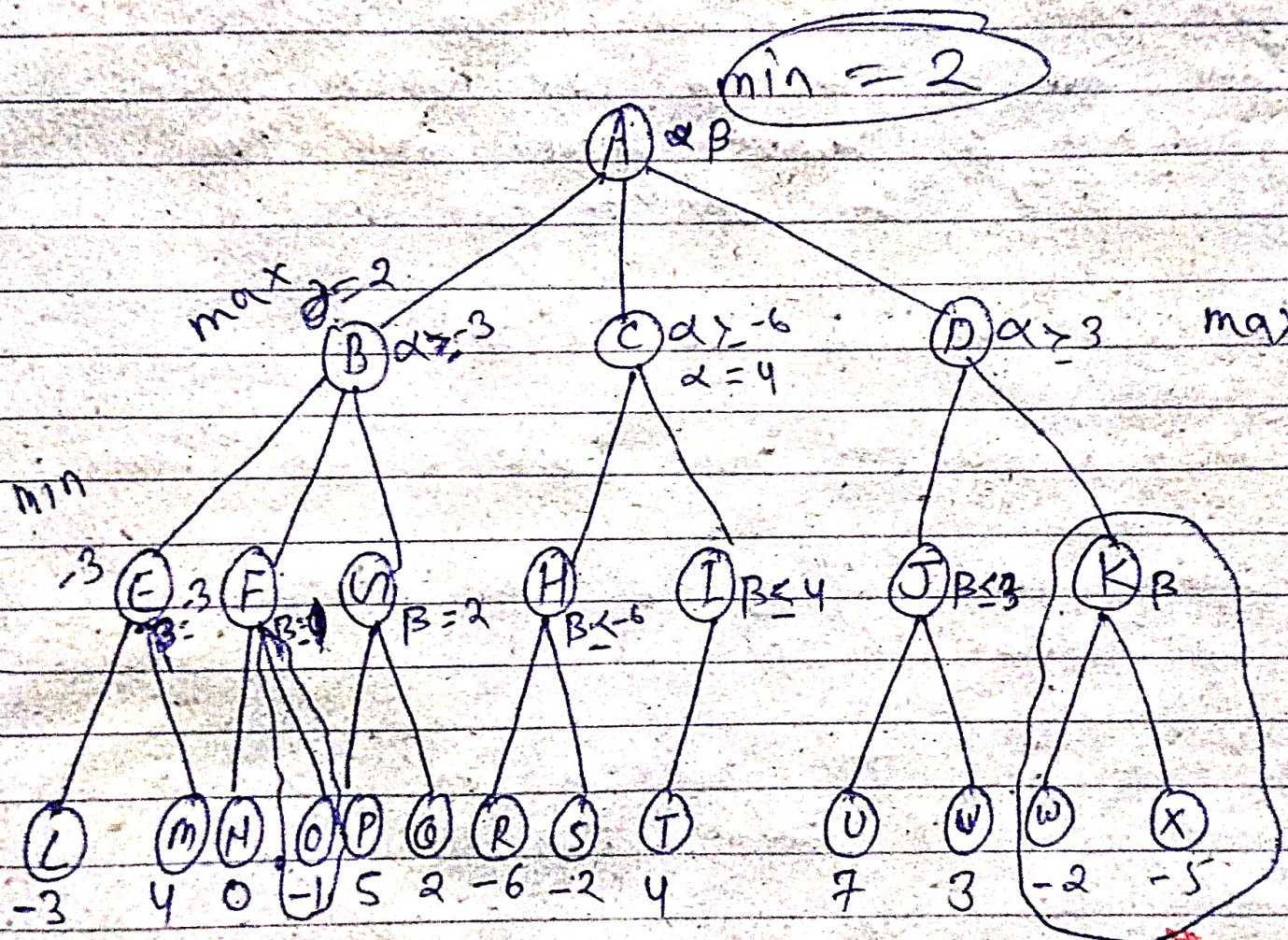


Q.5 Alpha - Beta pruning

Date: / / Page no:

→ It search by exploring less number of nodes.

(α) Alpha = max node, Beta (β) = min. node



Nodes = {K, X, W} that have been pruned

Pruned Efficiently

Q.6 The Block world problem

(I) there is a flate surface on which blocks are placed.

(II) there are numbers of ~~blocks~~ square blocks all of same size.

(III) they can be stacked one upon other.

(IV) there is a robot arm that can manipulate the blocks.

It can perform the action.

(I) STACK (A, B) → place Block A on block B.

the arm must holding A and surface of B must be clear.

(II) Upstack (A, B) → pickup the block A from top of the block B.

the arm must be empty and block A must have no blocks on the top of it.

(III) pickup (A) → Pickup the block A from the table and hold it.
the arm must be empty and there must be no block on top of A

(IV) Putdown (A) → Put block A on the table.
the arm must have been holding block A.

predicates are: →

(I) ON (A,B) – Block A is on block B

(II) ONTABLE (A) – Block A is on the table

(III) CLEAR (A) → there is nothing on top of A

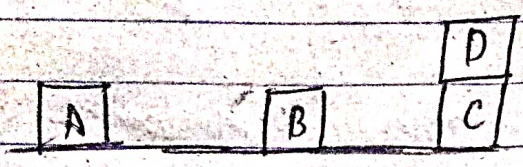
(IV) HOLDING (A) → the arm is holding block A

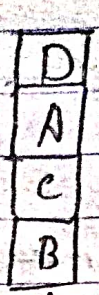
(V) ARM EMPTY → the arm is holding nothing

problem

Date / / Page no.

current state





Goal state

Solution → the function that need to define the current state.

(I) ONTABLE (A)
(II) CLEAR (A)
(III) ONTABLE (B)
(IV) CLEAR (B)
(V) ONTABLE (C)
(VI) ON (D, C)
(VII) CLEAR (D)

ARM EMPTY

the function that robot perform to solve the problem.

Date: / / Page no:

~~(I) UPSTACK (D, C)~~

~~(II) PU~~

~~(II) PICKUP (C)~~

(I) UPSTACK (D, C)

(II) PUTDOWN (D)

(III) PICKUP (C)

(IV) STACK (C, B)

(V) PICKUP (A)

(VI) STACK (A, C)

(VII) PICKUP (D)

(VIII) STACK (D, A)

the function used ~~are~~ to define goal stack.

(I) ONTABLE (B)
(II) ON (C, B)
(III) ON (A, C)
(IV) ON (D, A)
(V) CLEAR (D)
ARM EMPTY