Roll number: Name:

CS 6380 – Artificial Intelligence – End semester exam – December 27, 2020

Maximum marks – 170 Weight – 50 %. **Rough sheets at end of paper** *Be Concise*

1. Draw arcs from left to right to relate each term on the LHS to *all* related terms on the RHS. [6]

A* - 2,1	1 systematic
Best First - 2,1	2 heuristic
Breadth First - 1	3 crossover
Branch & Bound - 1,2,6*	4 temperature
Hill Climbing - 2,7	5 population size
Depth First - 7	6 admissible
Simulated Annealing - 2,4,8,7	7 unsystematic
Genetic Algorithm - 5,8,7,3	8 probabilistic

* - underestimating heuristic

2. Strike out one option before you answer. "*I think that it is* ~~possible / not possible~~ *for machines to become intelligent because* … ". (give two reasons). [4]

a. Day after day we are exploring deeper into biology of brain & imitating it and the depth of brain is finite (atomic level imitation can generate a thinking machine)

b.

1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Total

3. Describe the Turing Test. Do you think it is a good test for what it is designed to test? [5]

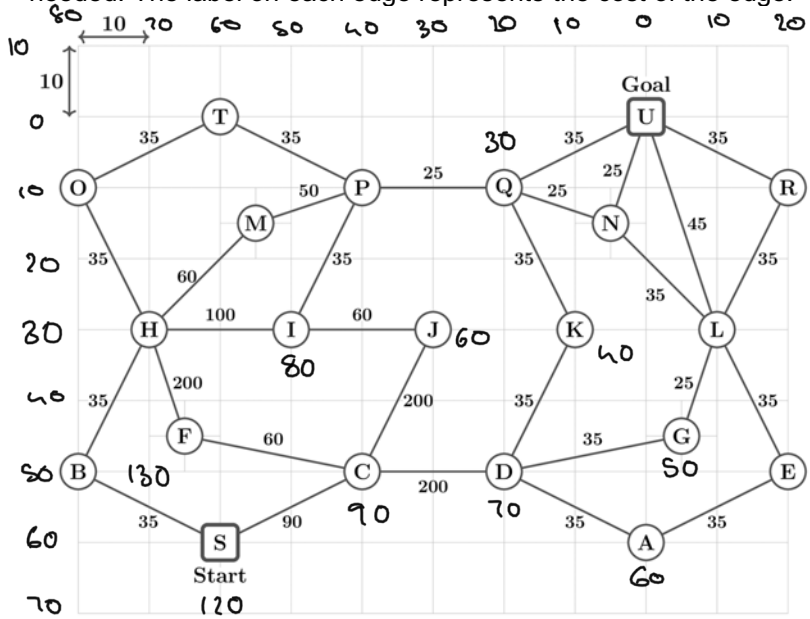
Turing test is a test to judge a machine's ability to exhibit intelligent behaviour that is equivalent to human. It' proposed by Alan Turing. The test involves 2 terminals interrogation side and response side. On the response side there are 2 sub divisions - one with an intellegent system and other with a human being. On the interrogation side there is a person who asks questions, get responses and judge the response whether it is produced by a man or machine. this is repeated many times and score of the system is calculated.

I think it is a good test but it has its limitations. this test still acts as a base point for judging the intelligence of a machine. One big problem is that the type of quries hugely affect the results - machines usually good with Yes/No queries but poor in conversational ones. Some systems have emerged to pass turing test by manipulating symbols and giving standard responses but they truly lack intelligence. So today researchess use other content specific tests rather than its abiliy to imitate humans.

4. Given the cost matrix below, show *three expansions* of the Branch and Bound algorithm in *the solution space* towards solving the TSP (traveling salesman problem). Use the heuristic “add cheapest allowed edge” when refining a node. Starting with the root label each node in your search tree with the lower bound estimated costs as discussed in the class. Mark the next node that is selected for refinement by the algorithm. [9]

	C	G	M	D	B
C	0	50	90	20	40
G	50	0	60	200	70
M	90	60	0	150	120
D	20	200	150	0	240
B	40	70	120	240	0

The following graph is an input for Questions 5 – 9. The nodes are placed on a grid where each side is 10 units. Node S is the start node and node U is the goal node. Use Manhattan distance as the heuristic function where needed. The label on each edge represents the cost of the edge.



5. List the nodes in the order inspected by Best First Search till it terminates. Does it find a path to the goal node? If yes list the path found along with its cost. [8]

Yes, It found path to the goal Node.

Path :- S C D K Q U

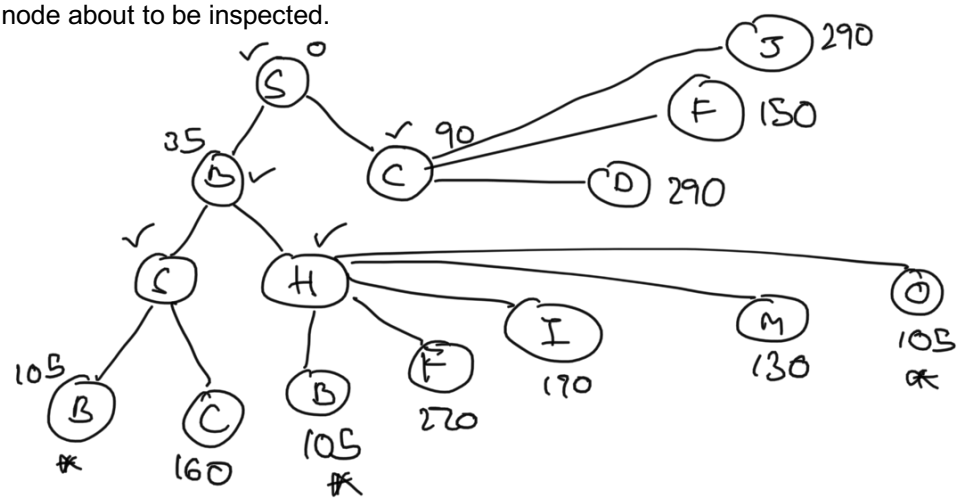
Cost :- 395

6. List the nodes in the order inspected by Hill Climbing till it terminates. Does it find a path to the goal node? If yes list the path found along with its cost. [8]

No path found to Goal

path :- S C J

7. Draw the search *tree* after 5 expansions of Branch&Bound. Label the nodes in the tree with the cost of the node. Mark the 6th node about to be inspected. [10]

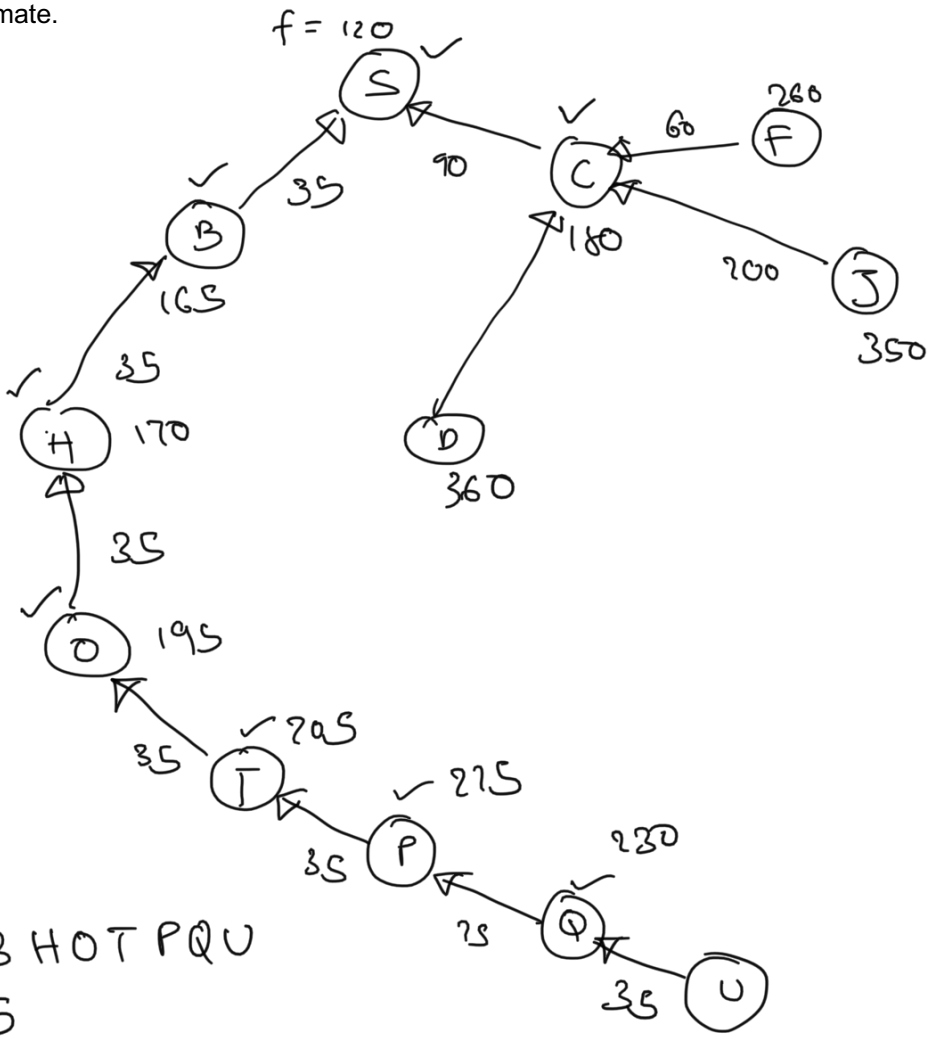


✓ → inspected nodes

✱ → Nodes to be inspected next (6th)

→ Any of them can be inspected

8. Draw the subgraph explored by Algorithm A* after it has terminated. Does it find a path to the goal node? If yes mark the path found and write down the cost of the path found. Label each node in your subgraph with the appropriate estimate. [12]



Yes

Path: S B H O T P Q U

Cost: 235

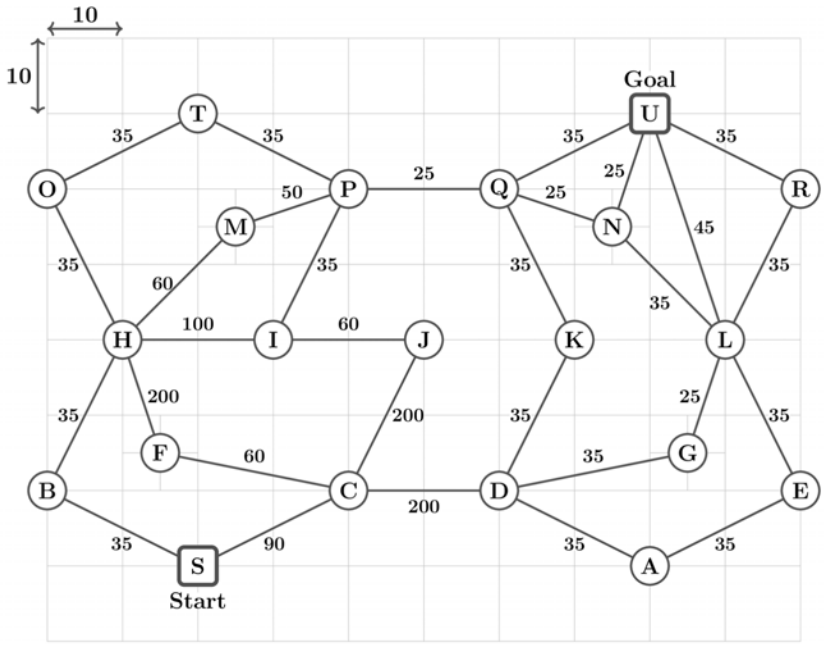
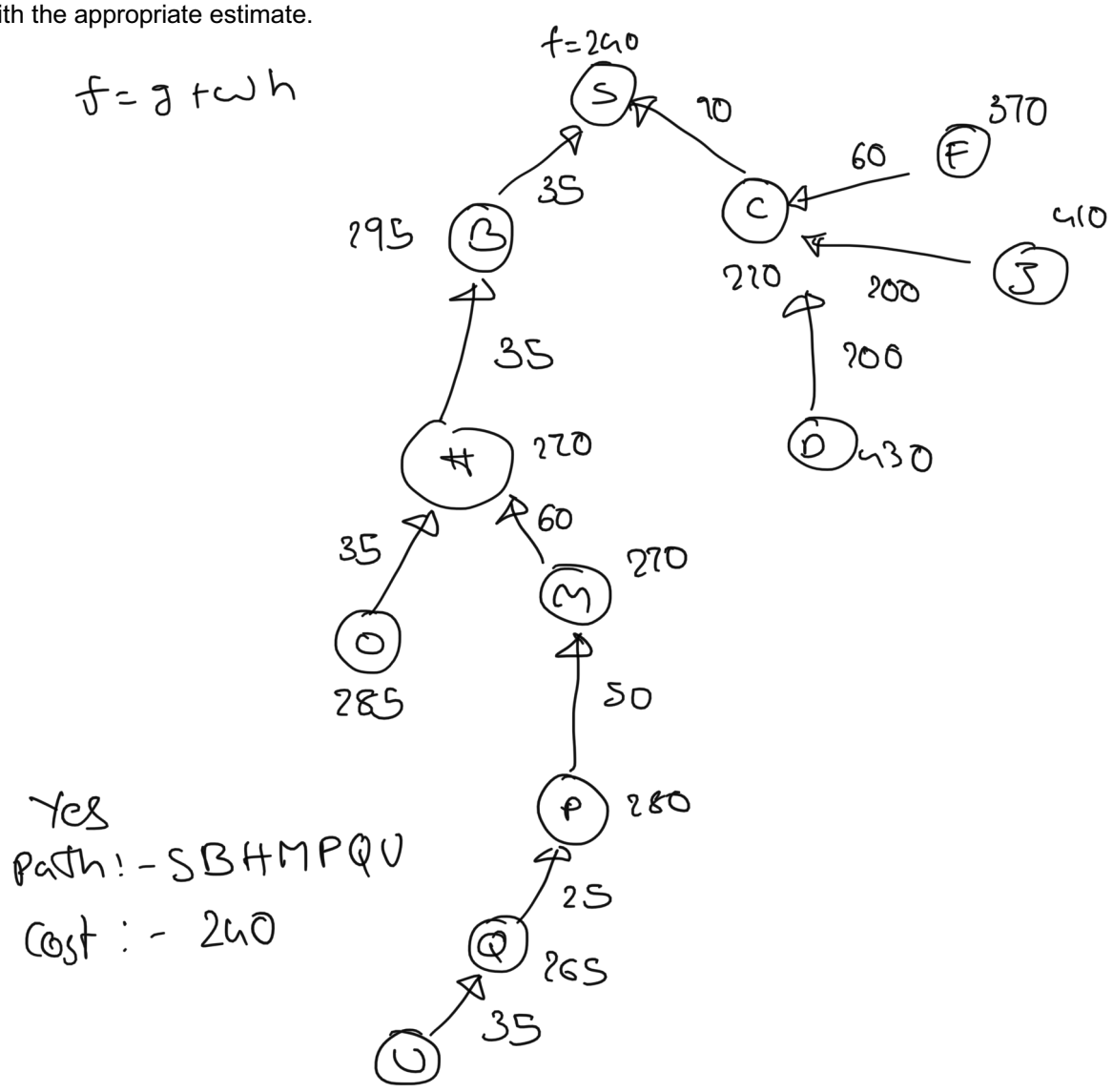


Figure repeated for convenience

9. Draw the subgraph explored by Algorithm wA* after it has terminated, when w=2. Does it find a path to the goal node? If yes mark the path found and write down the cost of the path found. Label each node in your subgraph with the appropriate estimate. [12]



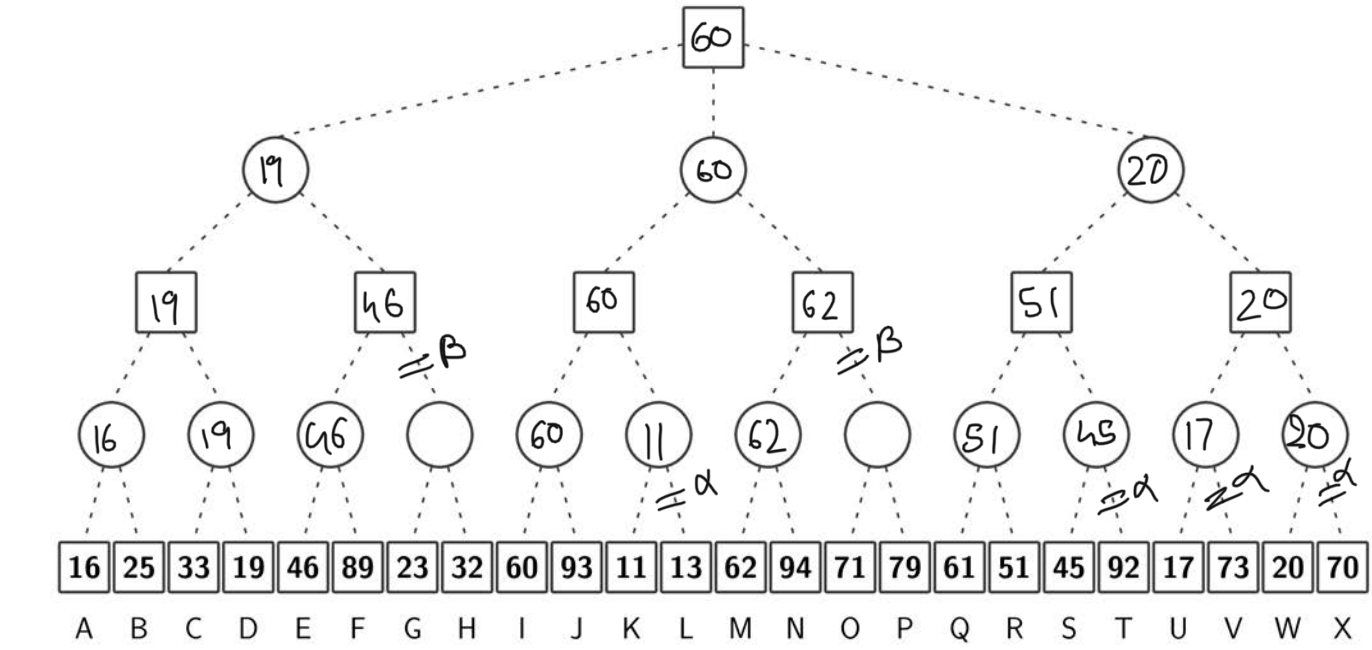
10. What improvement does the Smart Memory Graph Search algorithm make over the A* algorithm? How? Under what conditions? And at what cost? [6]

As memory is getting cheaper it is suggestible not to make recursive calls when A*_no pruning can easily solve the problem.

SMGS takes memory as a parameter too and only create relay layer when memory is the bottleneck. If the problem is small then it wont create any layers and allow A* to solve it directly.

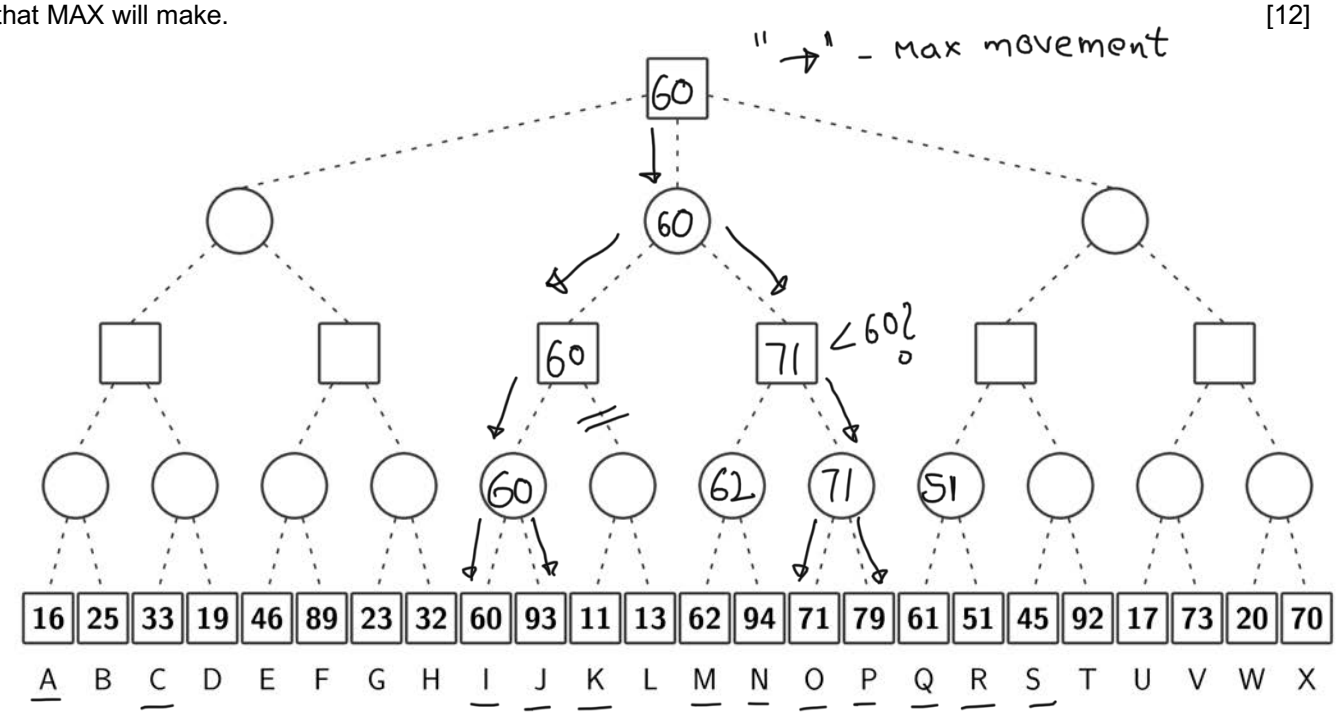
It identifies boundary nodes that can be potentially converted into relay nodes. These boundary nodes also stop search from leaking back.

11. For the following game tree, numbers in the leaf nodes are *e(J)* values. Show the *order* in which the leaf nodes are visited by the AlphaBeta algorithm searching from left to right. What is the value of the game tree? Mark the move that MAX will make. Also show the cutoffs and label them with alpha or beta as the case may be. [8]



→ A, B, C, D, E, F, I, J, K, M, N, Q, R, S, U, W

12. For the same game tree, show the *order* in which the leaf nodes are visited by the SSS* algorithm. Assume that the algorithm chooses the leftmost child at a decision point. What is the value of the game tree? Mark the move that MAX will make. [12]



A, C, I, K, Q, S, R, J, M, N, O, P

13. Given the planning problem described below show how *goal stack planning* with STRIPS operators will achieve the goal. You may choose *any order* where a choice has to be made. What is the plan found? [12]

Start state = {onTable(A), on(B,A), on(C,B), on(D,C), onTable(E), on(F,E), AE, clear(D), clear(F)}
Goal description = {on(C,E)}

14. For the (new) planning problem below but with a two-armed robot, draw the partial plan found that is an optimal solution plan *finishing the earliest* possible. What is the *makespan* of the plan? [8]

Start state = {onTable(A), on(B,A), on (C,B), on(D,C), onTable(E), on(F,E), AE(1), AE(2), clear(D), clear(F)}
Goal description = {on(C,D), onTable(B), on(D,F)}

15. Assume the following schema for Working Memory Elements

(Applicant ^name ^age ^gender)	gender: M/F/N
(Employed ^person ^employer)	employer: self/government/corporate/ngo
(Habits ^person ^hobby)	hobby: smoking/trekking/bridge
(Education ^person ^completed)	completed: highSchool/bachelors
(Eligible ^person ^loan)	loan: yes/no

A bank uses the following rules for deciding whether a person is eligible for a loan or not.

A. If the applicant's name is VM or NM he is eligible
B. If the applicant is in a corporate or government job and does not smoke s/he is eligible
C. If the applicant is a self-employed female she is eligible
D. If the applicant is a female graduate then she is eligible
E. If the applicant has finished high school and smokes then s/he is not eligible

Express the above rules in an OPS5 like language [6]

```
A: (P A
   (Applicant ^name <<VM NM>> ^name <x>)
   -->
   (make (Eligible ^person <x> ^loan yes))

B: (P B
   {(Employed ^person <x> ^employer <<corporate
    government>>)
    - (Habits ^person <x> ^hobby smoking)}
   -->
   (make (Eligible ^person <x> ^loan yes))

C: (P C
   {(Employed ^person <x> ^employer self)
    (Applicant ^name <x> ^gender F)}
   -->
   (make (Eligible ^person <x> ^loan yes))

D: (P D
   {(Applicant ^name <x> ^gender F)
    (Education ^person <x> ^completed bachelors)}
   -->
   (make (Eligible ^person <x> ^loan yes))
```

(P ∈ (education ^person <x> ^completed highSchool)
∧ (Habits ^person <x> ^hobby smoking))
→
(make (Eligible ^person <x> ^loan no))

16. Construct and draw a Rete Net for the above rules. [8]

17. Given the following Working Memory, list the Conflict Set for the above set of rules [12]

1) (Applicant ^name Kavita ^age 37 ^gender F)
2) (Applicant ^name Puran ^age 27 ^gender M)
3) (Applicant ^name Manisha ^age 26 ^gender F)
4) (Habits ^person Kavita ^hobby bridge)
5) (Habits ^person Puran ^hobby smoking)
6) (Education ^person Kavita ^completed bachelors)
7) (Education ^person Puran ^completed bachelors)
8) (Education ^person Manisha ^completed highSchool)
9) (Employed ^person Manisha ^employer corporate)
10) (Employed ^person Puran ^employer self)
11) (Employed ^person Kavita ^employer ngo)

Which element of the Conflict Set would be selected if the conflict resolution strategy is Specificity?

Which element of the Conflict Set would be selected if the conflict resolution strategy is Recency?

Who is/are eligible for a loan?

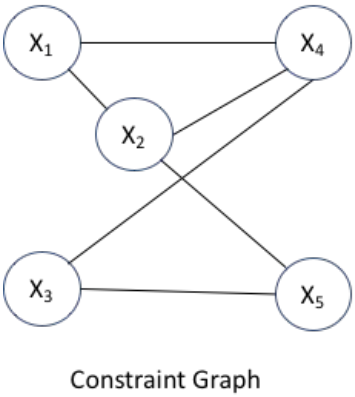
18. Express the above rules for loan eligibility in First Order Logic [6]

19. Draw the goal tree for the query "*Is there someone who is eligible for a loan?*" Mark one solution on it. [6]

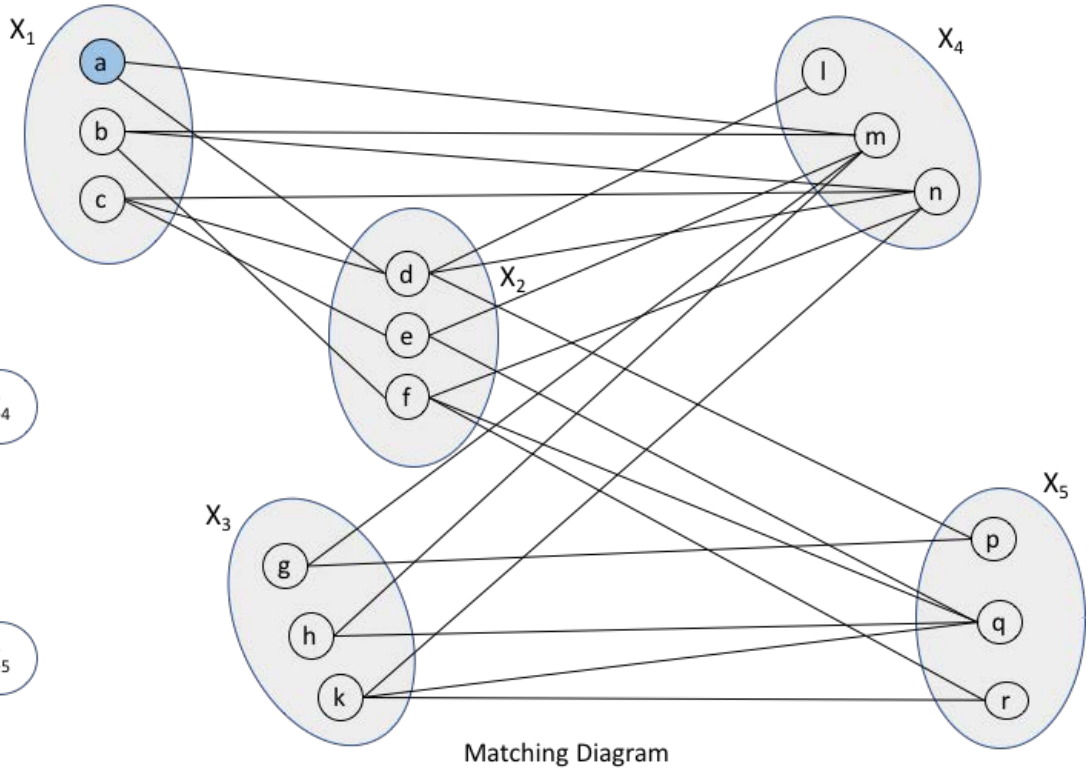
20. The figure below shows the constraint graph of a Binary CSP on the left and A PART of the matching diagram on the right. Please ASSUME a UNIVERSAL relation in the matching diagram where there is no constraint between variables in the constraint graph. The variables, and their values, are to be considered in *alphabetical* order.

    Algorithm Forward Checking is about to begin by assigning X_1=a. What are the next six values assigned to variables? Draw the matching diagram at this point.

    What is the first solution found by the algorithm? [12]



Constraint Graph



Matching Diagram