KOÇ UNIVERSITY
College of Engineering

COMP 341: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

Midterm 1

Fall 2022, 22/11/2022
Duration: 120 Minutes

Name: Solutions

ID: 00110001 00110000 00110000

- This exam contains 7 pages including this cover page and 5 questions. Check to see if any pages are missing. Put your initials on the top of every page, in case the pages become separated.
- By submitting this exam, you **agree** to fully comply with Koç University Student Code of Conduct, and accept any punishment in case of failure to comply. If you do not submit this exam on time, you will receive 0 credits.
- The exam is **closed book** and **closed notes**. You are **not** allowed to use any additional material including any electronic equipment such as computers and mobile phones.
- You are expected to be able provide clear and concise answers. Gibberish will not receive any credit. Your answers need to be readable by a human, illegible writing is not gradable.
- Read each question carefully and make sure to follow instructions. The written instructions take precedence over an answer that the instructor might give you during the exam, unless the instructor makes a class wide announcement.
- Do not write in the table below.

Question:	1	2	3	4	5	Total
Points:	24	28	16	12	20	100
Score:						

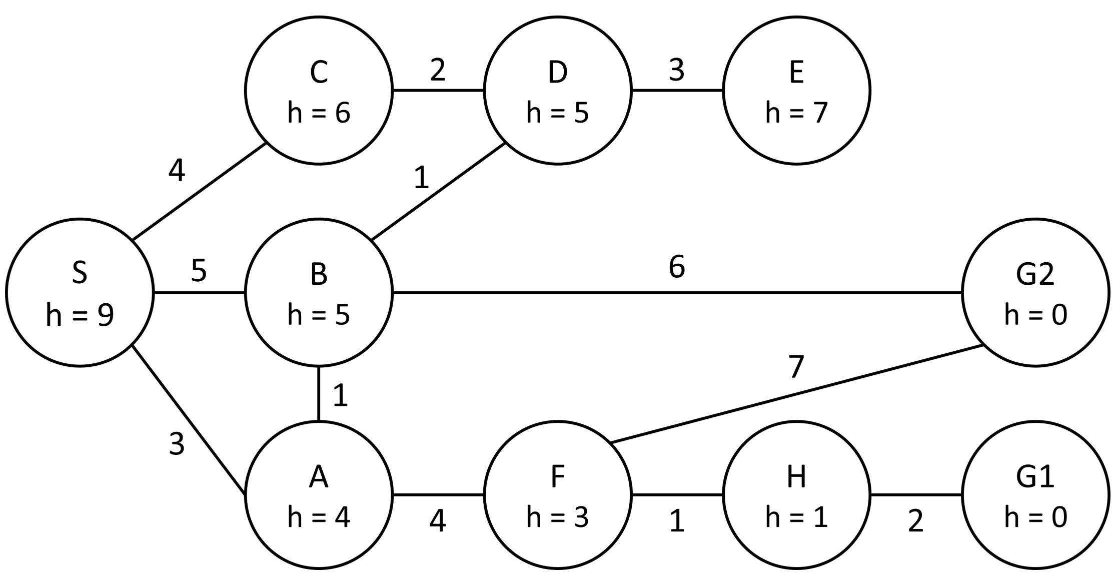
1. (24 points) True or False :

False	Reflex agents require a model of the environment.
True	Iterative deepening DFS has the same time complexity as regular DFS.
False	Uniform Cost Search always returns the same depth solution as Breadth First Search.
False	Genetic algorithms are a type of informed search algorithm.
True	Simulated annealing requires a model of the environment.
True	In constraint satisfaction problems, k consistency does not imply k-1 consistency.
Both	Only the value of the root state is guaranteed to be correct with alpha-beta pruning. You need to figure out the correct value of the first "deepest" state as well but this wasn't explicitly mentioned in the lecture notes so I am accepting both answers.
False	It is enough for expectiminimax evaluation functions to rank the goodness of states.

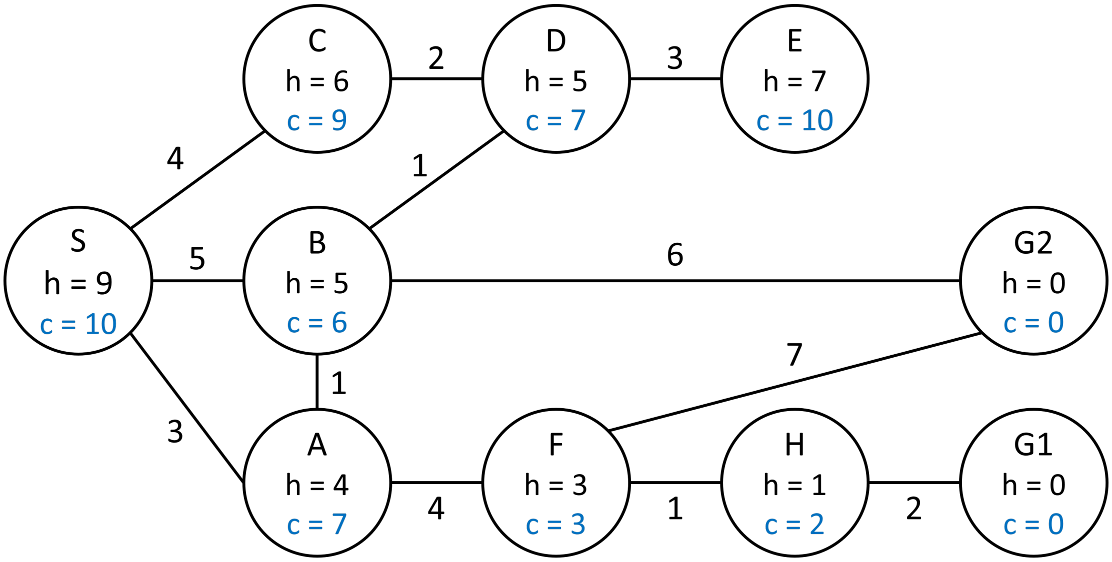
2. (28 points) You are given the weighted graph below, where **S** is the start state, and **G1** and **G2** are the goal states. The arcs represent transitions and cost of each transition is given next to the arcs. The numbers inside the nodes represent their heuristic value.

   You are asked to write the visiting/expanding order (order of popping from the frontier) of the nodes and the resulting solution paths with the given algorithms. The neighbors are added alphabetically to the frontier. The alphabetical order is: A,B,C,D,E,F,G1,G2,H,S. For algorithms using priority queues, break the ties alphabetically and assume that the priority values are updated when an existing node with a higher priority is being pushed.

   Warning: The arcs go both ways!



(repeated to give you an additional figure to work on)



(a) (6 points) Depth First Search:
Popped Node Order:
Stack based: S,C,D,E,B,G2
Recursive: S,A,B,D,C,E,G2
Resulting Path:
Stack based: S-C-D-B-G2
Recursive: S-A-B-G2

(b) (10 points) Uniform Cost Search:
Popped Node Order:
S,A,B,C,D,F,H,E,G1
Resulting Path:
S-A-F-H-G1

(c) (10 points) A* Search:
Popped Node Order:
S,A,B,C,D,F,H,G1
Resulting Path:
S-A-F-H-G1

(d) (2 points) Is the given heuristic admissible?
Yes it is

3. (16 points) You are given six variables, A, B, C, D, E, F each with the domain $\{1, 2, 3, 4, 5, 6\}$. You are also given the the following constraints on the variables:

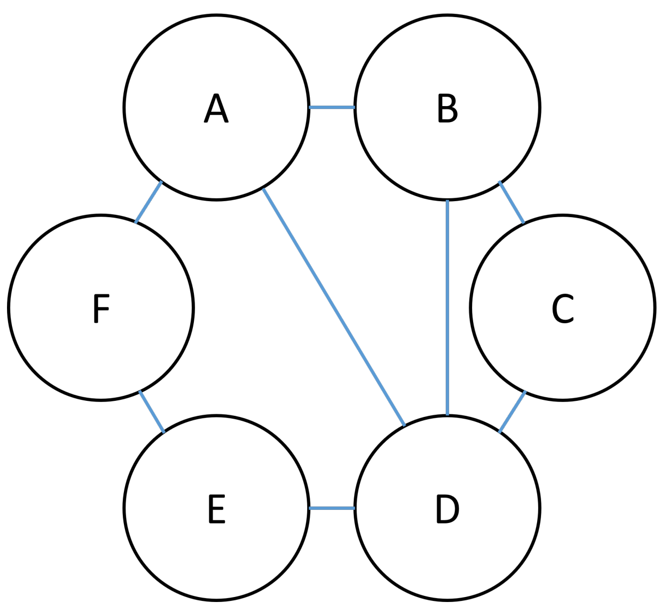
1. $A > 4$
2. $B < 6$
3. C is even
4. E is odd
5. $E + F$ is even
6. $C + D > 3$
7. $A \neq F$
8. $A \neq B$
9. $A \neq D$
10. $B \neq C$
11. $B \neq D$
12. $D \neq E$

(a) (2 points) On the grid below cross out the values from each domain that are eliminated by only enforcing **unary constraints** (i.e. constraints containing only a single variable).

A	~~1~~	~~2~~	~~3~~	~~4~~	5	6
B	1	2	3	4	5	~~6~~
C	~~1~~	2	~~3~~	4	~~5~~	6
D	1	2	3	4	5	6
E	1	~~2~~	3	~~4~~	5	~~6~~
F	1	2	3	4	5	6

0.5 per correctly applied unary constraint, -0.25 per incorrect application (no negative values)

(b) (4 points) Draw tansition arcs to build the binary constraint graph by only taking the **binary constraints** (i.e. constraints containing exactly two variable) into account. (Hints: There are 8 binary constraints so you need to have 8 arcs. We recommend you put the constraints on the arcs so that it is easier to solve the remaining problems.)



0.5 per correct arc, -0.25 per incorrect arc (no negative values)

(c) (1 point) According to the Minimum Remaining Value (MRV) heuristic, which variable should be assigned to first? Use the degree heuristic as a first and alphabetical order as a second tie-breaker. Perform this after enforcing unary constraints (part (a), making the CSP 1-consistent).

A since it only has 2 values remaining in its domain.

(d) (2 points) After enforcing unary constraints (part (a), making the CSP 1-consistent), what value would you assign to the variable D based on the Least Constraining Value (LCV) heuristic? Take the smallest value in case of a tie.

2,4 and 6 each only affect the variable of a single variable (B,B and A). We take 2 since it is the smallest

(e) (6 points) In addition to enforcing unary constraints (part (a)), now also run the arc consistency constraint propagation algorithm (AC3) on the initial problem. Cross out the values that are eliminated. Use the space if needed.

A	~~1~~	~~2~~	~~3~~	~~4~~	5	6
B	1	2	3	4	5	~~6~~
C	~~1~~	2	~~3~~	4	~~5~~	6
D	1	2	3	4	5	6
E	1	~~2~~	3	~~4~~	5	~~6~~
F	1	~~2~~	3	~~4~~	5	~~6~~

The black strikeouts carry on from part (a). Blue ones are done with AC3. (This is going to be short, the original intention was to have $C + D < 3$)

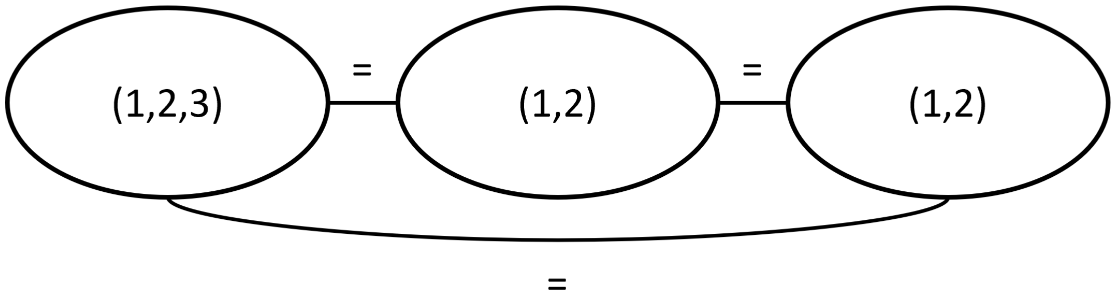
- When we look at the variable E and the constraint $E + F$ is even, we can remove 2,4 and 6 from F.

(f) (1 point) Give a complete assignment to values that does not violate any constraints. There are many viable answers...

A	B	C	D	E	F
6	4	2	1	3	5

4. (12 points) Answer the miscellaneous questions below

   (a) (6 points) Given the CSP below, answer the questions about consistency with True/False. The arcs represent "equal to" constraints.



   1-Consistent: True   2-Consistent: False   3-Consistent: True

   (b) (6 points) When would you stop a genetic algorithm, other than the "maximum number of generations reached"?

   This is open ended but one trivial, two obvious and one less obvious answers:

   - Maximum "duration" is reached (3 points since it is almost the same as max generations)
   - A minimum level of fitness is reached
   - No significant improvement in the population after several iterations
   - The "best" individual is a certain percent of the population

5. (20 points) Consider the following game. There are two players, X and Y. At each stage, the state of the game is an integer. When it is X's turn they have a choice of two possible moves:

   - Move A: $N := N - 23k_1$
   - Move B: $N := N - 10$

   If X performs A, then they roll a six-sided dice to determine the value of k_1. If the dice comes up 3 or less, $k_1 = -1$, otherwise, $k_1 = 1$. When it is Y's turn, they have a choice of two possible moves:

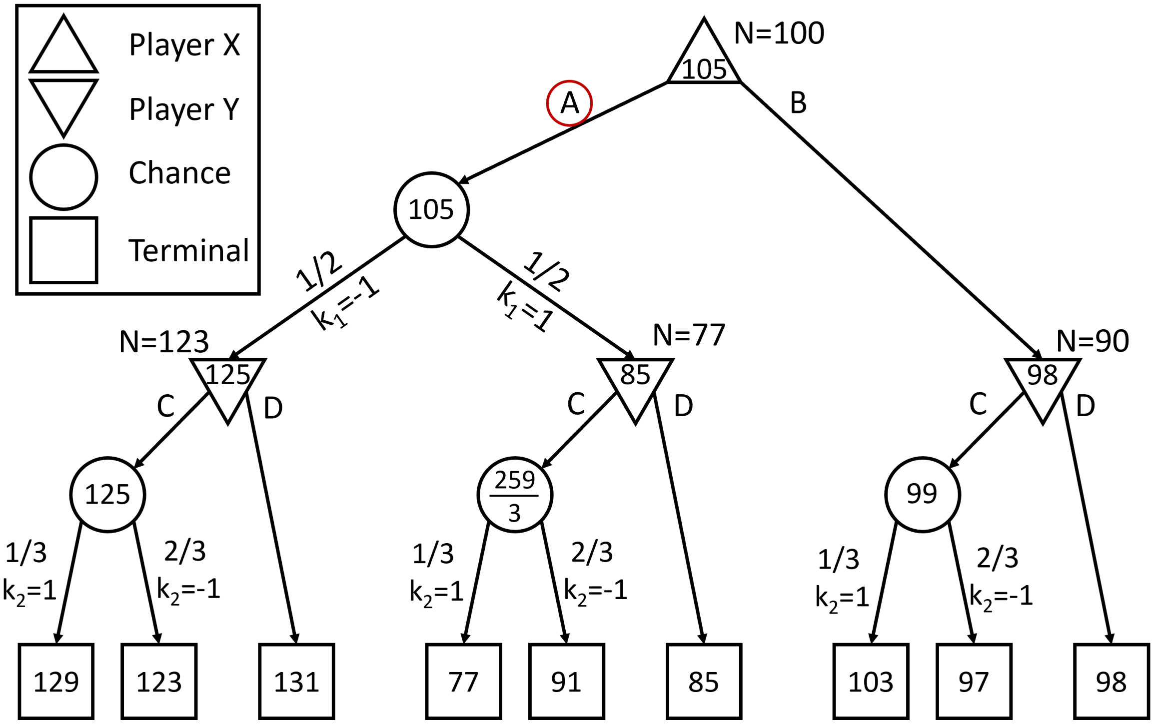
   - Move C: $N := N + (N + 3k_2)\%20$
   - Move D: $N := N + 8$

   Where % is the modulo operator, e.g. $5\%3 = 2$. If Y performs C, then they roll a six-sided dice to determine the value of k_2. If the dice comes up 2 or less, $k_2 = 1$, otherwise, $k_2 = -1$.

   The goal of X is to maximize N and the goal of Y is to minimize it. At the start N=100, and it is X's turn to play. The game will be played for 1 full turn, i.e., X will play one turn, then Y will play one turn and the game ends. The final value of a game is the value of N when it ends.

   (The questions continues on the next page)

Draw the game tree below. You should have nodes for X, nodes for Y, chance nodes and terminal nodes. Make the type of the node clear (e.g. use different shapes). Write the value of the states near each node and additionally the value of N near each player node. Mark the transitions with moves from the player nodes and with probabilities from the chance nodes. Mark the best starting move for X. (Hints: There should be 17 total nodes, 9 terminal nodes, 4 chance nodes)



- 12 N calculations (9 terminal,3 min): 0.25 per = 3
- 16 transition markings: 8 moves, 8 from chance, 0.2 per = 3.2,
- Chance values (using their calculated values): 3 for min, 1 for max, 1.5 per, total 6
- 3 Min and 1 max values: 1 per, total 4
- True action 1 point
- Clear node markings 0.5 per type (all or nothing): 2
- Generic clarity: 1 point