CMPT 310 Midterm 1, Summer 2019

Toby Instructor

Last name *exactly as it appears on student card*	[redacted]
First name *exactly as it appears on student card*	[redacted]
SFU Student #	[redacted]
SFU email *ends with sfu.ca*	[redacted]

This is a **closed book exam**: notes, books, computers, calculators, electronic devices, etc. are **not permitted**. Do not speak to any other students during their exam or look at their work. If you have a question, please remain seated and raise your hand and a proctor will come to you.

	Out of	***Your Mark***
Agent Architecture	10	4
Search	10	4
Constraint Satisfaction	10	10
Short Answer	10	6.5
Total	40	24.5

Agent Architecture

a) (5 marks) Give the definition of a **rational agent**.

A rational agent is an agent that can observe and analyze the environment and decide on an action according to the constraints and percept history. maximizes its performans.

5

b) (5 marks) What is a **table-driven agent**, and how does it work? What is one **good** thing about such an agent? What are two different **bad** things about it?

A table driven agent has its actions defined in a provided table in the program. So the agent simply looks up in the table the action it should do.

3 or 5

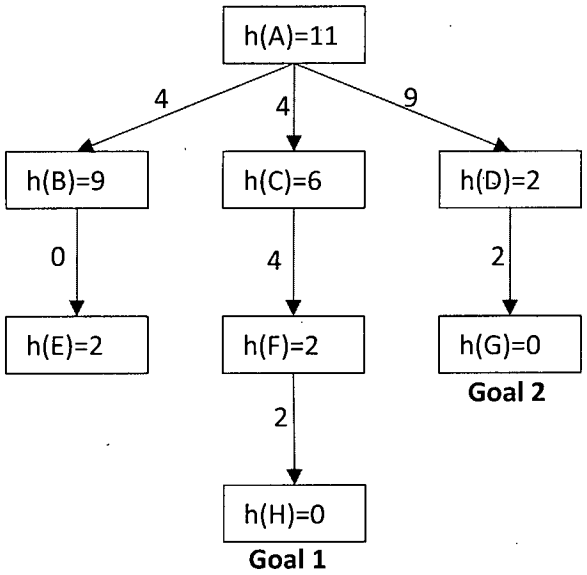
(V) The agent just needs to find the rule and behave according to it.
Similar to simple-reflex-agents: Simple and easy
→ simple and clear: small agents in simple domains can be implemented.

(X) The designer may not be able think of all the possible scenerios. Thus may cause the agent don't proceed on an action where the consequences can be fatal. This is also fails the completeness

(X) If the table is too long then the agent might not have enough time to go through the table to find the rule or learn the table. This also fails the time constraint.

(X) It takes massive amount of memory.

Searching



In the tree on the left, the starting node A is the root. The capital letter in each node is the node's name, and the number is the h-value for that node. Altogether, the h-values define a heuristic function h.

Each edge of the tree is labelled with its cost, and the two goal nodes, H and G, are marked.

For example, node G has an h-value of 0, and the cost of going from node D to node G is 2.

In the first few questions, a node is **visited** when it is removed from the frontier. **If there is a tie** about what node to visit next, always choose the node that comes first alphabetically.

a) (2 marks) If you start at node A, in what order will the nodes be visited by **uniform-cost search**?

A, B, C, E, F, D, H, G ✓ g(n)

b) (2 marks) If you start at node A, in what order will the nodes be visited by **greedy best-first search**?

A, D, G, C, F, H, B, E ✓ h(n)

c) (2 marks) Is the heuristic function h **admissible**? If not, why not?

ASK admissible is not overestimating

The heuristic function [h(n)] is admissible if the h(n) is either
- under-estimated or } consistent ✗ 0
- exactly estimated.

h(A) = 10 however it is given as 11 so that the h() is not admissible

If h(n) is over-estimated from node to the goal than it it is h(n) is inadmissible.

d) (2 marks) If you start at node A, in what order will the nodes be visited by **A* search**?

A, C, F, H, D, G, B, E
A, D, G, C, F, H, B, E ✗ 0 g(n)+h(n)

e) (2 marks) If you start at node A, what nodes (and in what order) will basic **hill-climbing** visit? The value of a node n is f(n)=11-h(n), and the higher the value of f the better.

? smallest h(n) in the expended choosen node.

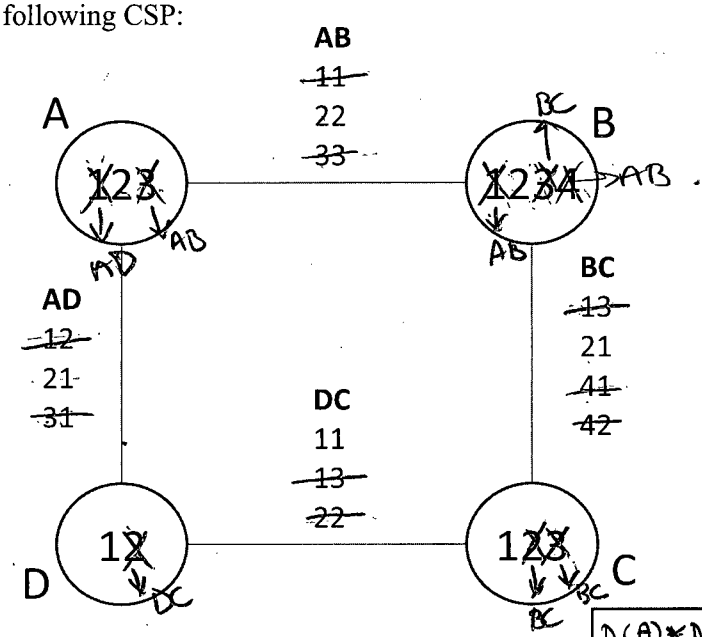
A, D, G, C, F, H, B, E ✗ 0

Basic hill-climb doesn't have strategy to restart. after finding a goal it reaches to the end Even if it re-starts it will find the same goal

4

Constraint Satisfaction

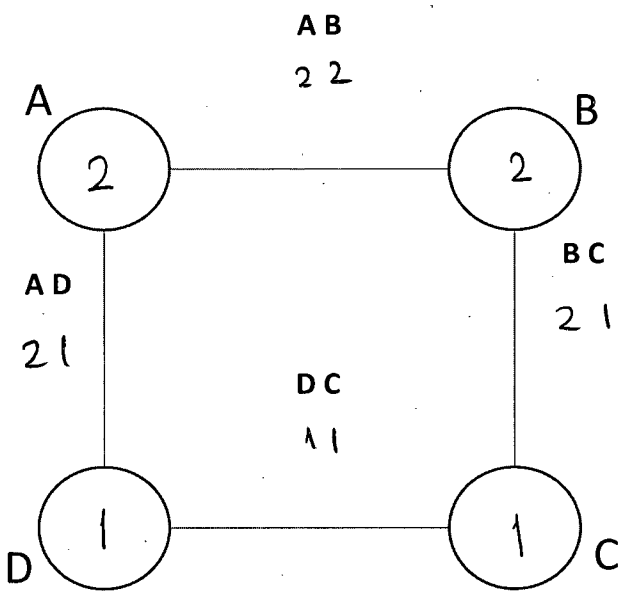
(8 marks) Consider the following CSP:



a) (1 mark) What is the size of the search space of the above CSP?

D(A)*D(B)*D(C)*D(D) = 3 x 4 x 3 x 2 = 72

b) (8 marks) Create an **arc consistent** version of the above CSP. Fill in the domains (in the circles) and constraints (under the corresponding letter pairs) here:



(1 mark) What is the size of the search space of the arc consistent CSP in b)?

1 x 1 x 1 x 1 = 1

Short Answer

a) (1 mark) What is the name of the main algorithm that most of the best traditional chess-playing programs used?	alpha-beta search algorithm 1
b) (1 mark) What is the name of the search algorithm used by the AlphaZero chess playing program?	Monte-Carlo search algorithm 0.5
c) (1 mark) *True* or *False*: AlphaZero learned to play chess by playing games against itself.	True 1
d) (1 mark) *True* or *False*: in practice, the major problem with A*-search when solving is that it runs out of memory. T	False 0 True
e) (1 mark) *True* or *False*: A*-search with an **inadmissible** heuristic on a finite graph sometimes may not find a goal node even though one exists. F	True 0 False
f) (1 mark) *True* or *False*: If you run the AC3 algorithm on an arc consistent CSP, then the CSP will not be changed.	True 1
g) (1 mark) *True* or *False*: In CSP backtracking search, the **minimum remaining values** (**MRV**) heuristic says that you should choose to next assign the node whose domain is the smallest.	True 1
h) (1 mark) *True* or *False*: When solving CSPs, **forward checking** is **not** useful with backtracking search, but **is** useful when making a CSP arc consistent.	False 1
i) (1 mark) *True* or *False*: The **min-conflicts** algorithm for solving CSPs is both incomplete and non-optimal. T	False 0
j) (1 mark) *True* or *False*: An agent can't be truly intelligent unless it is conscious.	Discussable but False 1

6.5