CMPT 310 Sample Midterm
Summer 2019

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

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	9	
Search	13	
Constraint Satisfaction	10	
Short Answer	10	

Total	42	

Agent Architecture

a) (2 marks) Give the definition **percept**, along with an example of two different ones.

A **percept** is an input the agent receives about the environment. Examples of percepts include: keystrokes, mouse movements, file contents, network packets, input from a camera, input from a sonar, …

b) (2 marks) Give the definition of **percept sequence**.

A **percept sequence** is a complete history of everything an agent has perceived.

c) (5 marks) What is a **simple reflex agent**, and how does it work? What is one **good** thing about such an agent? What are two different **bad** things about it?

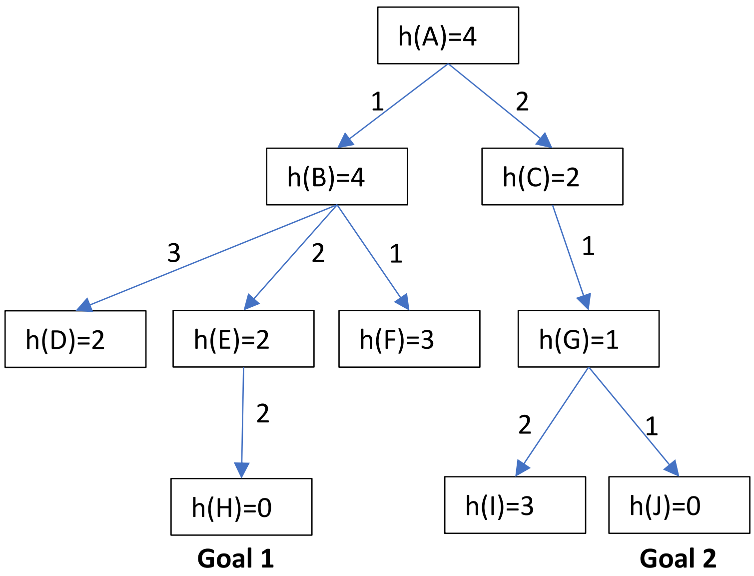
a simple reflex agent consists of condition-action rules of the form P -> act, where P is the current percept, and act is the action that should be performed

Pro: it's simple and clear; result in smaller programs than equivalent table-based agents

Con: they only consider the current percept, and have trouble in environments that are not fully observable

Con: can get stuck in infinite loops (could get unstuck by the introduction of some randomness into the rules)

Searching



In the tree on the left, 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 J, are marked.

For example, node J has an h-value of 0, and the cost of going from node C to node G is 1.

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) (3 marks) What are the values of g(H), g(I), and g(J), where g is the regular g-function as defined in A*-search and related algorithms.

g(H)=5, g(I)=5, g(J)=4

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

A B C F E G D J H I

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

A C G J I B D E H F

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

h is admissible because it never-overestimates the true cost to the nearest goal

e) (2 marks) What are f(H) and f(I), where f is the f-value function as defined in A*-search?

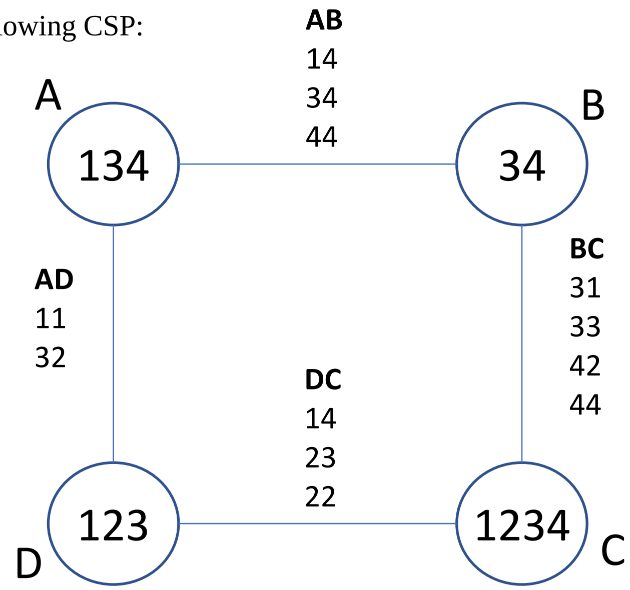
f(H)=g(H)+h(H)=5+0=5; f(I)=g(I)+h(I)=8

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

A C G J B E F H D I

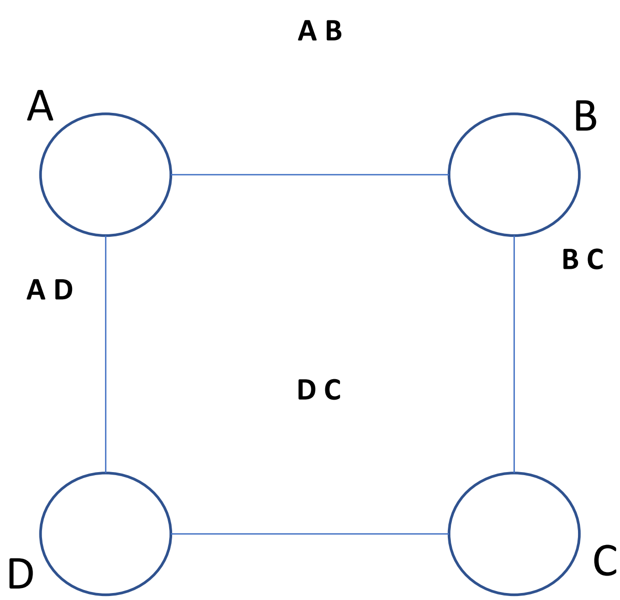
Constraint Satisfaction

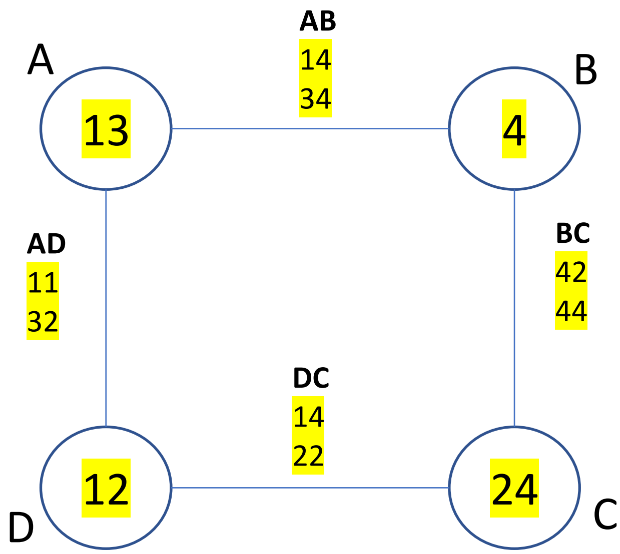
(8 marks) Consider the following CSP:



a) (1 mark) What is the size of the search space of the above CSP? 3*2*4*3=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:





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

2*1*2*2=8

Short Answer

a)	(1 mark) *True* or *False*: AlphaZero uses **alpha-beta search** as it is main search algorithm when playing chess.	False
b)	(1 mark) *True* or *False*: One of the interesting things about AlphaZero is that it can be used to create world-class players for some games other than chess.	True
c)	(1 mark) *True* or *False*: AlphaZero learned to play chess by playing games against other very strong chess programs (such as StockFish).	False
d)	(1 mark) *True* or *False*: Genetic algorithms can be described as a variant of stochastic beam search.	True
e)	(1 mark) *True* or *False*: If an A*-search heuristic is **consistent**, then it is also **admissible**.	True
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
g)	(1 mark) *True* or *False*: In CSP backtracking search, one very popular way of choosing the next variable to assign is to pick the variable with the largest domain.	False
h)	(1 mark) *True* or *False*: The **min-conflicts** CSP algorithm assigns variables one at a time.	False
i)	(1 mark) *True* or *False*: The **min-conflicts** algorithm for solving CSPs is an example of a **local search** algorithm.	True
j)	(1 mark) *Absolutely* or *No Way*: Only humans can be truly intelligent.	Either answer is okay: for this course it's a matter of opinion.