

**1. [10%] General AI Knowledge**

C6A1285E-9863-4A55-A6E3-1A635A3D8669

561-F15-midterm1

#287

2 of 11



For each of the statements below, fill in the bubble T if the statement is always and unconditionally true, or fill in the bubble F if it is always false, sometimes false, or just does not make sense.

1	<input type="radio"/> T	<input checked="" type="radio"/> F
2	<input checked="" type="radio"/> T	<input type="radio"/> F
3	<input checked="" type="radio"/> T	<input type="radio"/> F
4	<input type="radio"/> T	<input checked="" type="radio"/> F
5	<input type="radio"/> T	<input checked="" type="radio"/> F
6	<input checked="" type="radio"/> T	<input type="radio"/> F
7	<input checked="" type="radio"/> T	<input type="radio"/> F
8	<input type="radio"/> T	<input checked="" type="radio"/> F
9	<input checked="" type="radio"/> T	<input type="radio"/> F
10	<input type="radio"/> T	<input checked="" type="radio"/> F

F 1- Backgammon is deterministic.

T 2- In search, a state can have a parent.

T 3- An algorithm with a complexity of  $3 \cdot n^2$  and another one with a complexity of  $5 \cdot n^2 + 3$  have the same Big-O complexity.  $\frac{5}{3}n^2$

$3n^2 + 3$

F 4- In a problem formulation, "Start  $\rightarrow$  Goal" can be an operator.

F 5- If you know that the correct answer to this question is "T", then answering "T" to it is optimal.

T 6- A complete search algorithm will always find a solution.

T 7- Even if all cars were self-driving, it would not make the world's roads a static environment.

F 8- The best chess programs in the world play chess optimally.

T 9- If all cars were self-driving, the world's roads would be a deterministic environment.

F 10- A first person shooter game (e.g. Halo, Quake, Call of Duty, Battlefield) played online between human players is not accessible to those human players.

76B03C49-BE07-4C28-B140-099B438BF3AB

561-F15-midterm1

#287

3 of 11



## 2. [30%] Search

Consider the search space on the following page, where  $S$  is the start state and  $D$ ,  $J$ , and  $K$  satisfy the goal test. Arcs are labeled with the cost of traversing them and the estimated cost to a goal is reported inside nodes.

For each of the following search strategies, indicate which goal state is reached (if any) and list, in order, all the states of the nodes popped off of the OPEN queue, and the cost of the path found by the strategy to reach from  $S$  to the goal state. When all else is equal, nodes should be removed from OPEN in alphabetical order.

You should **not expand nodes with states that have already been visited** (where "State  $X$  has been visited" here means: some node was previously created and enqueued with state  $X$ ). Note how the arcs in the figure are oriented, which means that you can only go from one state to another if the arrow points from the first to the second. For example, you can go from  $S$  to  $A$  (i.e.,  $A$  is a successor of  $S$ ) but not from  $A$  to  $S$  (i.e.,  $S$  is not a successor of  $A$ ).

a) [10%] Depth-first search

Goal state reached: J States popped off OPEN: S A C H J Path Cost 19

b) [10%] Uniform cost Search

Goal state reached: D States popped off OPEN: S A B F C E G I D Path Cost 10

c) [10%] A\* Search

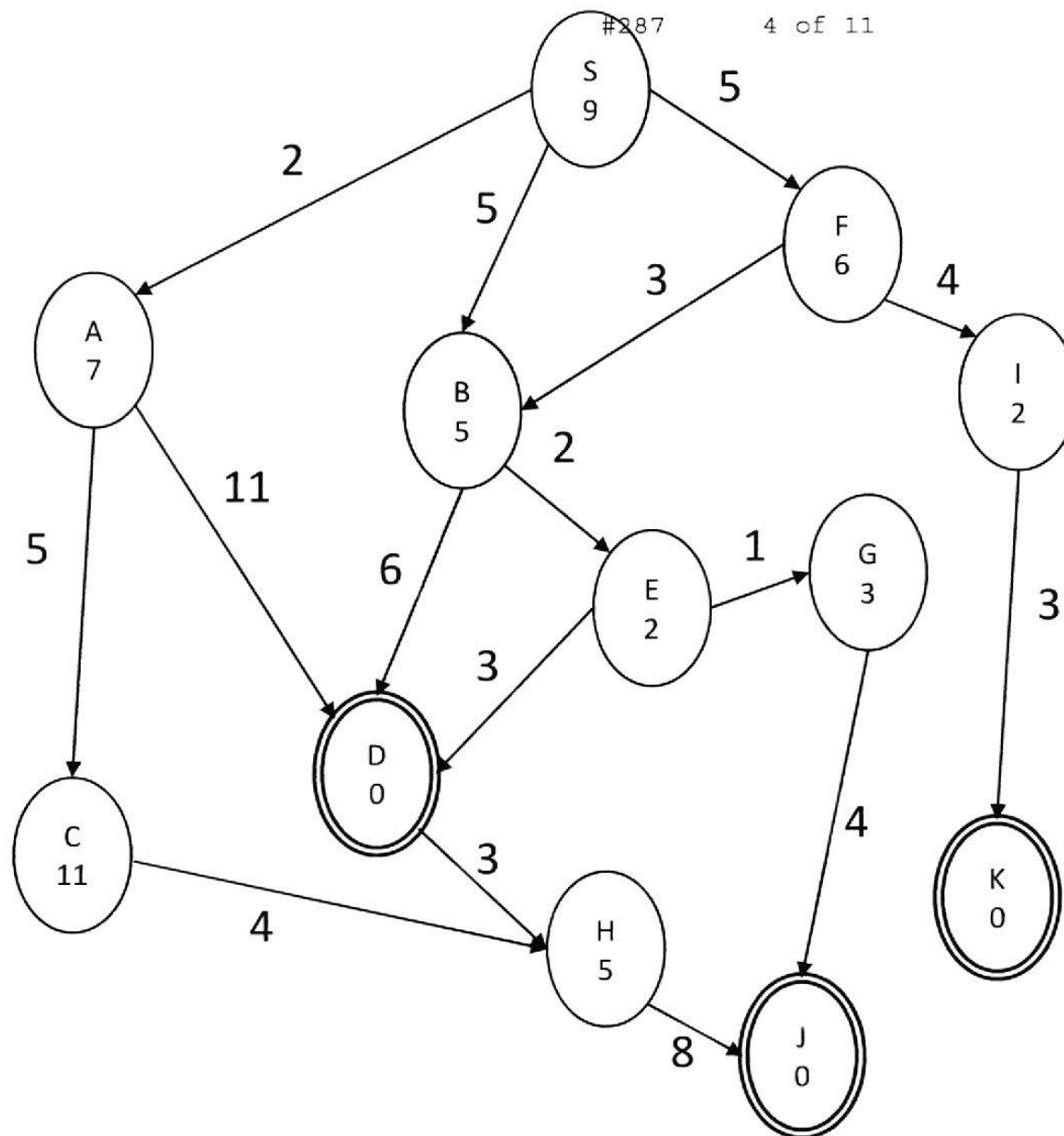
Goal state reached: D States popped off OPEN: S A B E D Path Cost 10

87E2031C-F588-4DCA-9802-4CBDD81DC52F

561-F15-midterm1

#287

4 of 11



DFS

 $\frac{11}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$ 

EX: S A C H J

UCS

 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$ 
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$ 

A\*

 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$

**3. [10%] Game Playing**

C8CE8FF2-DD2F-4119-95A1-A411D1BD1FAA

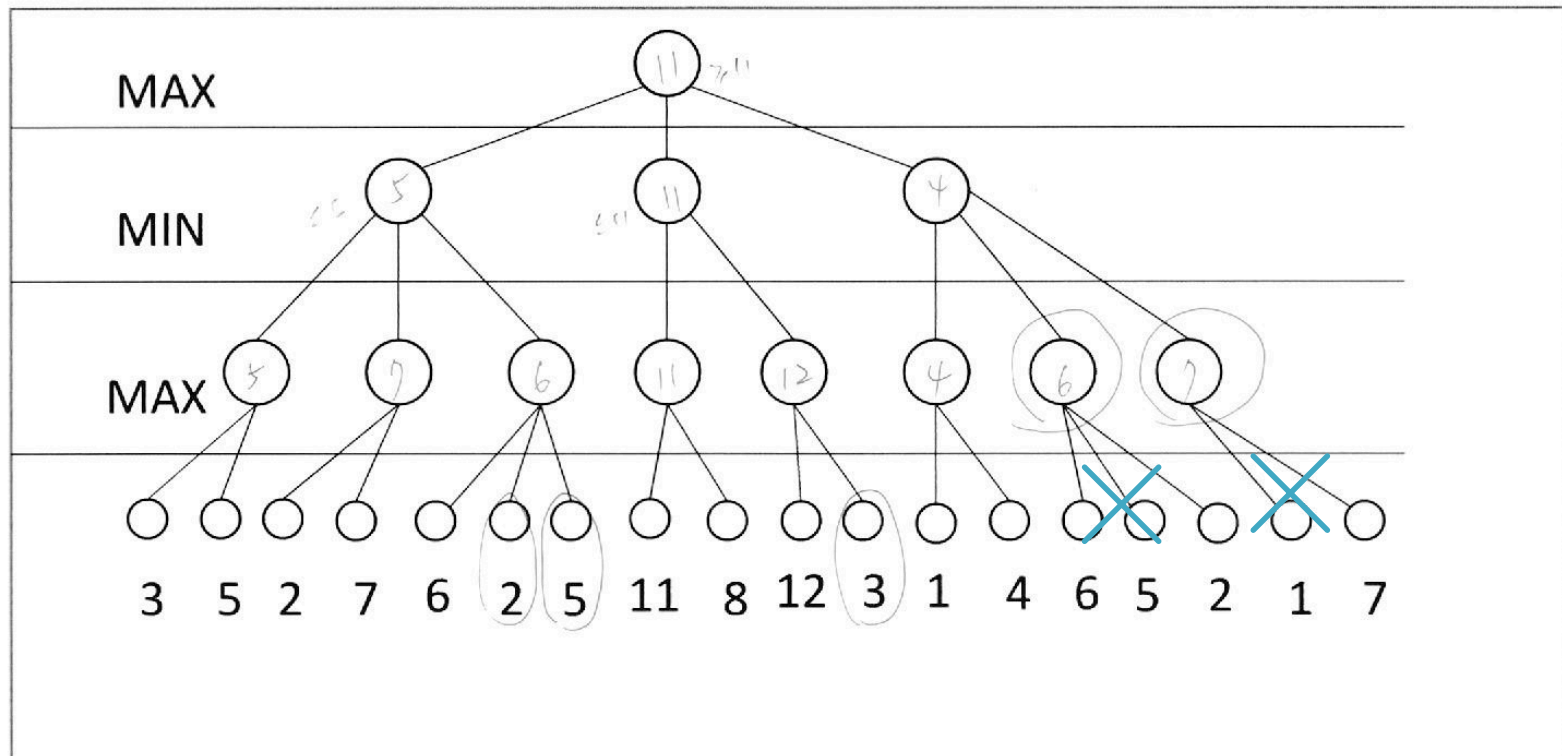
561-F15-midterm1

#287

5 of 11



Consider the following game tree in which the evaluation function values are shown below each leaf node. The root node is the maximizing player. Assume that the search always visits children left-to-right.



a) [4%] Compute the backed-up values according to the minimax algorithm. Show your answer by writing values next to the appropriate nodes in the above tree.

b) [6%] Which nodes will not be examined by the alpha-beta pruning algorithm? Circle them on the above tree.

**4. [20%] Constraint Satisfaction**

7D4597A0-655A-49A3-BAFE-4C7DAA6155E0

561-F15-midterm1

#287

6 of 11



You are in charge of a zoo.

**Five animals** need to be fed at specific times every **Monday, Wednesday and Friday**.

You need to assign **three zookeepers** on those days to feed the animals. The zookeepers have to wait until the feeding time is over. They have different skills, so some of them can only feed certain types of animals. They can also only feed one animal during a given time period.

Your animals need to be fed at the following times:

- Zebra: 6:00-6:30am
- Yabby: 6:30-7:00am
- X-Ray Fish: 6:45-7:15m
- Walrus: 6:15-6:45am
- Vervet: 6:15-6:30am

Your zookeepers have the following training:

- Ada can feed all animals except vervets.
- Bob can feed yabbies and walruses.
- Cecile can feed all animals.

a) Using the animal type as a variable, formulate this problem as a CSP with variables, domains and constraints. Constraints should be specified formally and precisely, but may be implicit rather than explicit.

Variables	Domains	Domain Values: Ada(A), Bob(B), Cecile(c)
Zebra (z)	AC	
Yabby (Y)	ABC	
X-Ray Fish (x)	AC	
Walrus (w)	ABC	
Vervet (v)	C	

Constraints:

Domain value cannot be equal for the following pairs

$(z, w) (z, v) (Y, x) (Y, w) (w, v)$



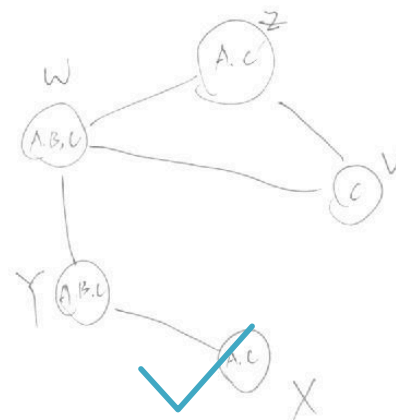
b) Draw the constraint graph.

661D9651-BD3C-45A7-9980-119A85F86E2E

561-F15-midterm1

#287

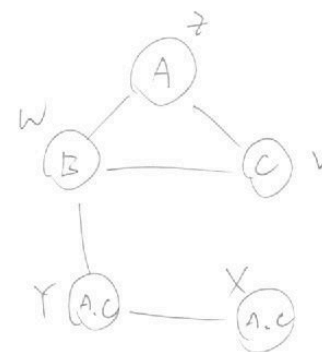
7 of 11



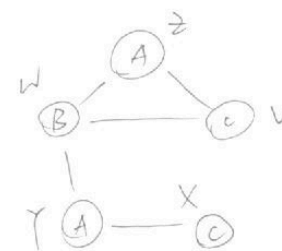
nodes represent variables  
 arcs represent constraint  
 (two nodes on the same arc  
 could not be assigned the  
 same value).

c) Show the domains of the variables after running arc-consistency on this initial graph (after having already enforced any unary constraints).

Variables	Domains
Z	A
Y	AC
X	AC
W	B
V	C



d) Give one solution to this CSP.

$$Z=A, Y=A, X=C, W=B, V=C$$


### 5. [20%] Heuristic design for the Flip-Side game

43116C34-5B57-4937-9D27-790E6D96BD68

561-F15-midterm1

#287

8 of 11



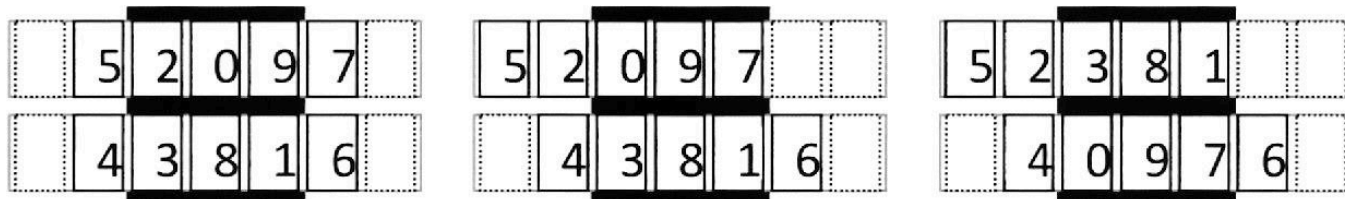
Flip-Side (source: <http://jaapsch.net/puzzles/flipside.htm>) is a game that consists of a frame containing 10 numbered blocks in **two rows** of five.

Each row contains additional space on either side of the blocks so that the blocks in the row can be slid either left or right. Note that all blocks slide together and no gaps can be formed within the blocks.

In the center of the frame is a flipper that holds 3 blocks from each row. When flipped, the three blocks in the top row swap with the three blocks directly beneath them.

The blocks are numbered from 0 to 9, and the aim is to arrange them in numerical order, i.e. 01234 on the top row, 56789 on the bottom row.

For example:



Initial Position

Shift Top Left

Flip

A. [2%] Describe a possible state representation for this game.

A state consists of two number arrays, each with 5 digits  
 (assigned from 0 to 9,  
 no duplicate)  
 and the position of two sub-frame in two rows  
 (position could be left, middle, or right)

9CFEEE10-30CE-4743-AE2A-2DDE181E4B96

561-F15-midterm1

#287

9 of 11



B. [3%] Describe the operators to be used during the search for a solution to this game.

Operators could be

- ① Shift the top sub-frame to left or right
- ② Shift the bottom sub-frame to left or right
- ③ Flip the blocks

C. [4%] What is the size of the state space, as a simple numerical formula?

Three positions for each sub-frame  $\Rightarrow 3 \times 3 = 9$   
Number of possible assignments of the digits would be  $10!$   
Therefore, the size of the state space would be  $9 \cdot 10!$   
#



C9BA9DE1-59AF-4E51-B767-C612D92E1951

561-F15-midterm1

#287

10 of 11



D. [1%] What kind of goal test will your search algorithm use?

To Test if the number on the top row is 01234,  
and the number on the bottom row is 56789  
(Need to satisfy both conditions)

E. [10%] Provide two admissible heuristics (other than  $h(n)=0$  for all  $n$ ) that can be used to guide an A\* search in the Flip-Side game. Briefly explain why each heuristic is admissible [each heuristic formulation: 2%; each explanation: 3%].

①  $h(n)=1$  for all possible state which doesn't pass the goal test  
 $h(n)=0$  for the goal state (which pass the goal test)

Explain: All of the non-goal state needs at least one move to achieve the goal state, so  $h(n)=1$  for them would not overestimate the distance from state to goal #

②  $h(n) = \frac{m}{6}$  where  $m$  represent the sum of misplaced indicator for every digit (indicate the digit is placed in the wrong row or not)  
Take digit 2 as example  
misplaced indicator =  $\begin{cases} 1, & 2 \text{ is placed in the bottom row} \\ 0, & 2 \text{ is placed in the top row} \end{cases}$

Explain: The goal state has  $h(n)=0$ , and  $h(n)$  is always positive for other states  
Each operation either doesn't change  $h(n)$  or reduce  $h(n)$  by at most 6  
(slide) (flip)

Even every step we could reduce  $h(n)$  by 6, it still needs  $\frac{m}{6}$  moves to achieve, so the  $h(n)$  heuristic is admissible #

**6. [10%] AI Applications**

15B90D92-D40D-4FDE-A9B8-2C6681959C27

561-F15-midterm1

#287 11 of 11

Circle the **best** choice for each question:**(a) [2%]** Eugene Goostman is famous for:

- a. Beating IBM Watson
- ☒ b. Passing the Turing Test
- c. Creating Google's self-driving car
- d. All of the above
- e. None of the above

**(b) [2%]** Genetic Algorithms are best suited to which problem domain?

- a. Travelling salesman problem
- b. N-queens problem
- c. Sudoku
- ☒ d. All of the above
- e. None of the above

**(c) [2%]** Creating a rational AI agent guarantees that the agent will:

- a. Always win
- b. Be omniscient
- ☒ c. Optimize its performance measure
- d. All of the above
- e. None of the above

**(d) [2%]** For suboptimal route planning, Google Maps could use this search technique:

- a. Greedy Search
- b. A\* with an inadmissible heuristic
- c. Depth-first search
- ☒ d. All of the above
- e. None of the above

**(e) [2%]** Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Which heuristic is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves?

- a. Straight-line distance
- b. Manhattan distance
- ☒ c.  $h(n)=1$
- d. All of the above
- e. None of the above

**1. [10%] General AI Knowledge**

C6A1285E-9863-4A55-A6E3-1A635A3D8669

561-F15-midterm1

#287

2 of 11



For each of the statements below, fill in the bubble T if the statement is always and unconditionally true, or fill in the bubble F if it is always false, sometimes false, or just does not make sense.

1	<input type="radio"/> T	<input checked="" type="radio"/> F
2	<input checked="" type="radio"/> T	<input type="radio"/> F
3	<input checked="" type="radio"/> T	<input type="radio"/> F
4	<input type="radio"/> T	<input checked="" type="radio"/> F
5	<input type="radio"/> T	<input checked="" type="radio"/> F
6	<input checked="" type="radio"/> T	<input type="radio"/> F
7	<input checked="" type="radio"/> T	<input type="radio"/> F
8	<input type="radio"/> T	<input checked="" type="radio"/> F
9	<input checked="" type="radio"/> T	<input type="radio"/> F
10	<input type="radio"/> T	<input checked="" type="radio"/> F

F 1- Backgammon is deterministic.

T 2- In search, a state can have a parent.

T 3- An algorithm with a complexity of  $3 \cdot n^2$  and another one with a complexity of  $5 \cdot n^2 + 3$  have the same Big-O complexity.  $\frac{5}{3}n^2$

$3n^2 + 3$

F 4- In a problem formulation, "Start  $\rightarrow$  Goal" can be an operator.

F 5- If you know that the correct answer to this question is "T", then answering "T" to it is optimal.

T 6- A complete search algorithm will always find a solution.

T 7- Even if all cars were self-driving, it would not make the world's roads a static environment.

F 8- The best chess programs in the world play chess optimally.

T 9- If all cars were self-driving, the world's roads would be a deterministic environment.

F 10- A first person shooter game (e.g. Halo, Quake, Call of Duty, Battlefield) played online between human players is not accessible to those human players.

76B03C49-BE07-4C28-B140-099B438BF3AB

561-F15-midterm1

#287 3 of 11



## 2. [30%] Search

Consider the search space on the following page, where  $S$  is the start state and  $D$ ,  $J$ , and  $K$  satisfy the goal test. Arcs are labeled with the cost of traversing them and the estimated cost to a goal is reported inside nodes.

For each of the following search strategies, indicate which goal state is reached (if any) and list, in order, all the states of the nodes popped off of the OPEN queue, and the cost of the path found by the strategy to reach from  $S$  to the goal state. When all else is equal, nodes should be removed from OPEN in alphabetical order.

You should **not expand nodes with states that have already been visited** (where “State  $X$  has been visited” here means: some node was previously created and enqueued with state  $X$ ). Note how the arcs in the figure are oriented, which means that you can only go from one state to another if the arrow points from the first to the second. For example, you can go from  $S$  to  $A$  (i.e.,  $A$  is a successor of  $S$ ) but not from  $A$  to  $S$  (i.e.,  $S$  is not a successor of  $A$ ).

a) [10%] Depth-first search

Goal state reached: J States popped off OPEN: S A C H J Path Cost 19

b) [10%] Uniform cost Search

Goal state reached: D States popped off OPEN: S A B F C E G I D Path Cost 10

c) [10%] A\* Search

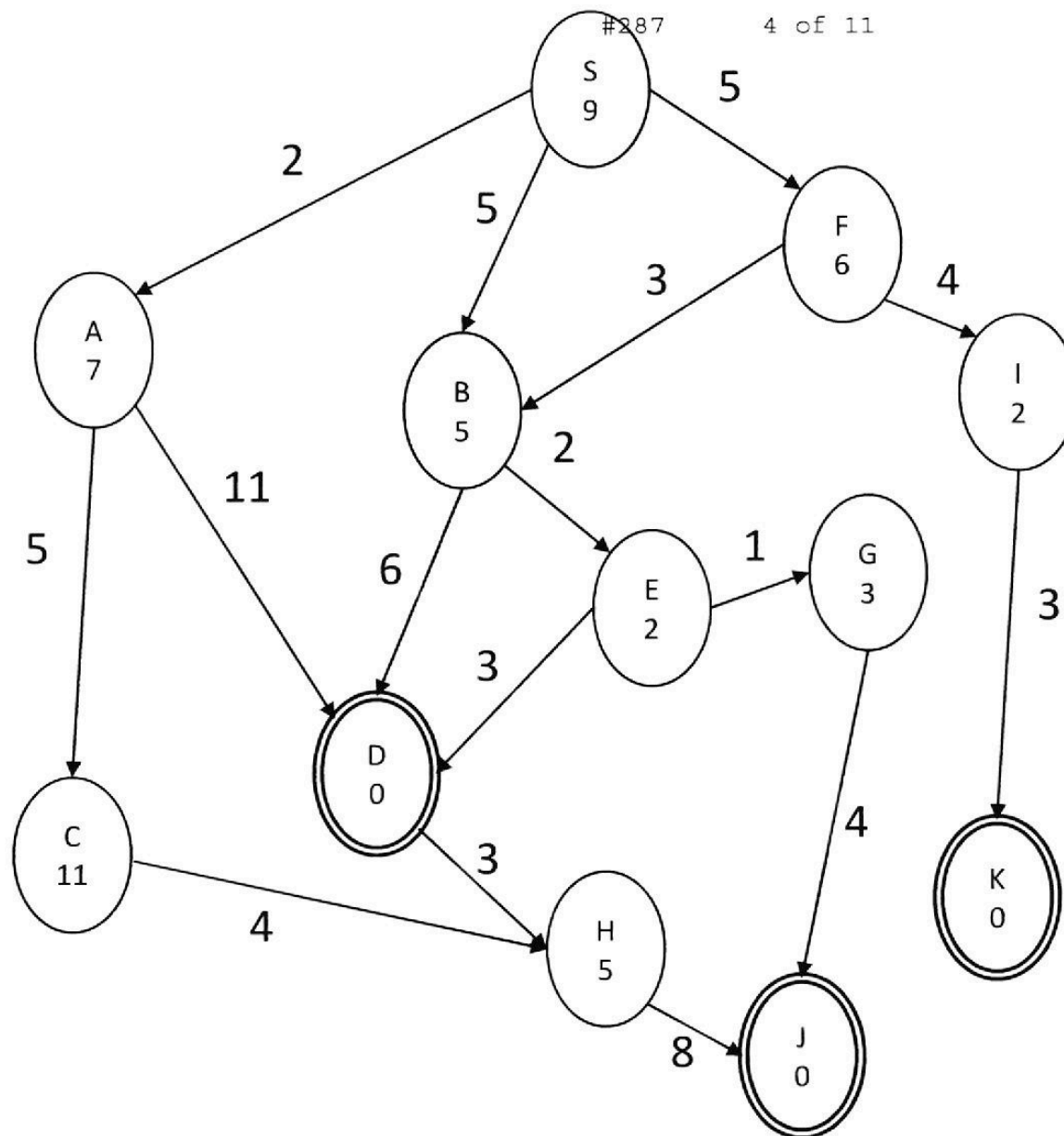
Goal state reached: D States popped off OPEN: S A B E D Path Cost 10

87E2031C-F588-4DCA-9802-4CBDD81DC52F

561-F15-midterm1

#287

4 of 11



DFS

 $\frac{11}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$ 

EX: S A C H J

UCS

 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$ 

A\*

 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$   
 $\frac{1}{8} \frac{1}{0}$

**3. [10%] Game Playing**

C8CE8FF2-DD2F-4119-95A1-A411D1BD1FAA

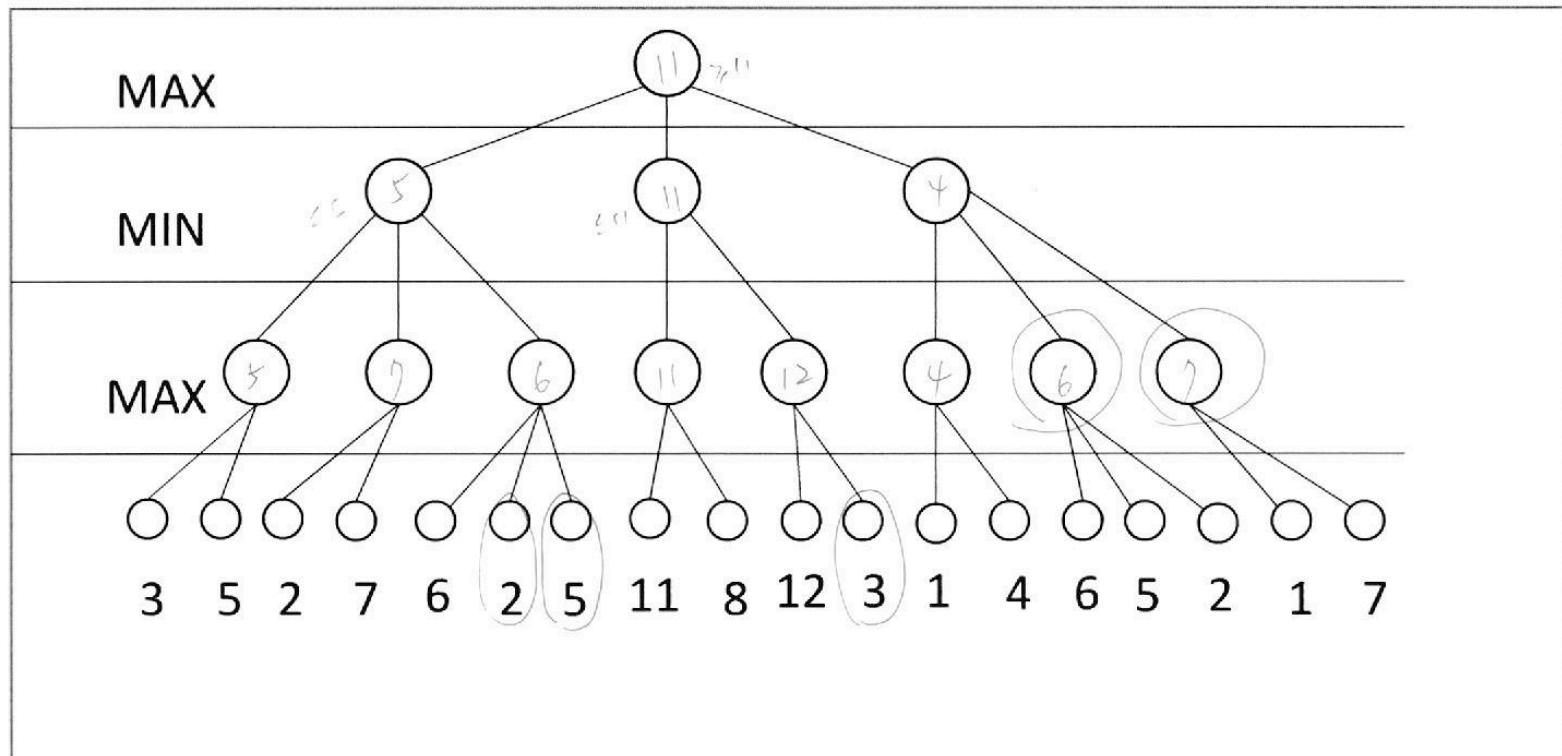
561-F15-midterm1

#287

5 of 11



Consider the following game tree in which the evaluation function values are shown below each leaf node. The root node is the maximizing player. Assume that the search always visits children left-to-right.



a) [4%] Compute the backed-up values according to the minimax algorithm. Show your answer by writing values next to the appropriate nodes in the above tree.

b) [6%] Which nodes will not be examined by the alpha-beta pruning algorithm? Circle them on the above tree.



**4. [20%] Constraint Satisfaction**

7D4597A0-655A-49A3-BAFE-4C7DAA6155E0

561-F15-midterm1

#287

6 of 11



You are in charge of a zoo.

**Five animals** need to be fed at specific times every **Monday, Wednesday and Friday**.

You need to assign **three zookeepers** on those days to feed the animals. The zookeepers have to wait until the feeding time is over. They have different skills, so some of them can only feed certain types of animals. They can also only feed one animal during a given time period.

Your animals need to be fed at the following times:

- Zebra: 6:00-6:30am
- Yabby: 6:30-7:00am
- X-Ray Fish: 6:45-7:15m
- Walrus: 6:15-6:45am
- Vervet: 6:15-6:30am

Your zookeepers have the following training:

- Ada can feed all animals except vervets.
- Bob can feed yabbies and walruses.
- Cecile can feed all animals.

a) Using the animal type as a variable, formulate this problem as a CSP with variables, domains and constraints. Constraints should be specified formally and precisely, but may be implicit rather than explicit.

Variables	Domains	Domain Values: Ada(A), Bob(B), Cecile(c)
Zebra (z)	AC	
Yabby (Y)	ABC	
X-Ray Fish (x)	AC	
Walrus (w)	ABC	
Vervet (v)	C	

Constraints:

Domain value cannot be equal for the following pairs

(z, w) (z, v) (Y, x) (Y, w) (w, v)

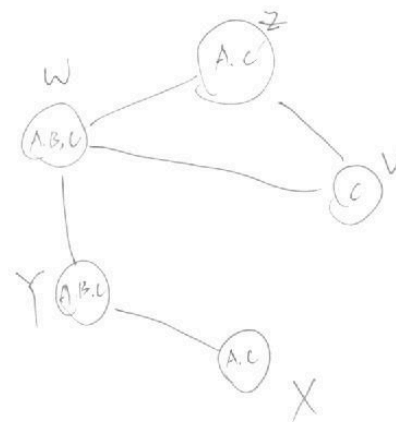
b) Draw the constraint graph.

661D9651-BD3C-45A7-9980-119A85F86E2E

561-F15-midterm1

#287

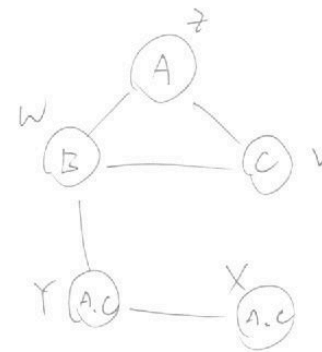
7 of 11



nodes represent variables  
 arcs represent constraint  
 (two nodes on the same arc  
 could not be assigned the  
 same value).

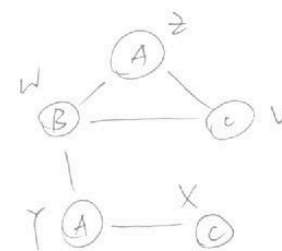
c) Show the domains of the variables after running arc-consistency on this initial graph (after having already enforced any unary constraints).

Variables	Domains
Z	A
Y	AC
X	AC
W	B
V	C



d) Give one solution to this CSP.

$$Z=A, Y=A, X=C, W=B, V=C$$



### 5. [20%] Heuristic design for the Flip-Side game

43116C34-5B57-4937-9D27-790E6D96BD68

561-F15-midterm1

#287

8 of 11



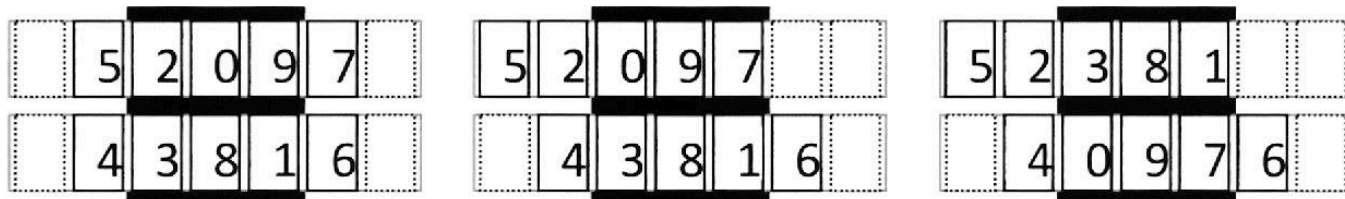
Flip-Side (source: <http://jaapsch.net/puzzles/flipside.htm>) is a game that consists of a frame containing 10 numbered blocks in **two rows** of five.

Each row contains additional space on either side of the blocks so that the blocks in the row can be slid either left or right. Note that all blocks slide together and no gaps can be formed within the blocks.

In the center of the frame is a flipper that holds 3 blocks from each row. When flipped, the three blocks in the top row swap with the three blocks directly beneath them.

The blocks are numbered from 0 to 9, and the aim is to arrange them in numerical order, i.e. 01234 on the top row, 56789 on the bottom row.

For example:



Initial Position

Shift Top Left

Flip

A. [2%] Describe a possible state representation for this game.

A state consists of two number arrays, each with 5 digits  
 (assigned from 0 to 9,  
 no duplicate)  
 and the position of two sub-frame in two rows  
 (position could be left, middle, or right)

9CFEEE10-30CE-4743-AE2A-2DDE181E4B96

561-F15-midterm1

#287

9 of 11



B. [3%] Describe the operators to be used during the search for a solution to this game.

Operators could be

- ① Shift the top sub-frame to left or right
- ② Shift the bottom sub-frame to left or right
- ③ Flip the blocks

C. [4%] What is the size of the state space, as a simple numerical formula?

Three positions for each sub-frame  $\Rightarrow 3 \times 3 = 9$

Number of possible assignments of the digits would be  $10!$

Therefore, the size of the state space would be  $9 \cdot 10!$

#

C9BA9DE1-59AF-4E51-B767-C612D92E1951

561-F15-midterm1

#287

10 of 11



D. [1%] What kind of goal test will your search algorithm use?

To Test if the number on the top row is 01234,  
and the number on the bottom row is 56789  
(Need to satisfy both conditions)

E. [10%] Provide two admissible heuristics (other than  $h(n)=0$  for all  $n$ ) that can be used to guide an A\* search in the Flip-Side game. Briefly explain why each heuristic is admissible [each heuristic formulation: 2%; each explanation: 3%].

①  $h(n)=1$  for all possible state which doesn't pass the goal test  
 $h(n)=0$  for the goal state (which pass the goal test)

Explain: All of the non-goal state needs at least one move to achieve the goal state, so  $h(n)=1$  for them would not overestimate the distance from state to goal #

②  $h(n) = \frac{m}{6}$  where  $m$  represent the sum of misplaced indicator for every digit (indicate the digit is placed in the wrong row or not)  
Take digit 2 as example  
misplaced indicator =  $\begin{cases} 1, & 2 \text{ is placed in the bottom row} \\ 0, & 2 \text{ is placed in the top row} \end{cases}$

Explain: The goal state has  $h(n)=0$ , and  $h(n)$  is always positive for other states  
Each operation either doesn't change  $h(n)$  or reduce  $h(n)$  by at most 6  
(slide) (flip)

Even every step we could reduce  $h(n)$  by 6, it still needs  $\frac{m}{6}$  moves to achieve, so the  $h(n)$  heuristic is admissible #

**6. [10%] AI Applications**

15B90D92-D40D-4FDE-A9B8-2C6681959C27

561-F15-midterm1

#287

11 of 11



Circle the **best** choice for each question:

(a) [2%] Eugene Goostman is famous for:

- a. Beating IBM Watson
- ☒ b. Passing the Turing Test
- c. Creating Google's self-driving car
- d. All of the above
- e. None of the above

(b) [2%] Genetic Algorithms are best suited to which problem domain?

- a. Travelling salesman problem
- b. N-queens problem
- c. Sudoku
- ☒ d. All of the above
- e. None of the above

(c) [2%] Creating a rational AI agent guarantees that the agent will:

- a. Always win
- b. Be omniscient
- ☒ c. Optimize its performance measure
- d. All of the above
- e. None of the above

(d) [2%] For suboptimal route planning, Google Maps could use this search technique:

- a. Greedy Search
- b. A\* with an inadmissible heuristic
- c. Depth-first search
- ☒ d. All of the above
- e. None of the above

(e) [2%] Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Which heuristic is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves?

- a. Straight-line distance
- b. Manhattan distance
- ☒ c.  $h(n)=1$
- d. All of the above
- e. None of the above