For each question on the Mid-term Exam, “Zero” below gives the fraction of students who scored zero, “Partial” gives the fraction who got partial credit, and “Perfect” gives the fraction who scored 100%.

**Problem 1**
Zero: 11% (~20 students), Partial: 66% (~120 students), Perfect: 23% (~43 students)

**Problem 2**
Zero: 6% (~11 students), Partial: 68% (~124 students), Perfect: 26% (~47 students)

**Problem 3**
Zero: 19% (~34 students), Partial: 43% (~78 students), Perfect: 38% (~70 students)

**Problem 4**
Zero: 5% (~10 students), Partial: 75% (~136 students), Perfect: 20% (~36 students)

**Problem 5**
Zero: 4% (~8 students), Partial: 90% (~163 students), Perfect: 6% (~11 students)

**Problem 6**
Zero: 2% (~3 students), Partial: 1% (~2 students), Perfect: 97% (~177 students)

**Problem 7**
Zero: 5% (~9 students), Partial: 32% (~59 students), Perfect: 63% (~114 students)

**Problem 8**
Zero: 0% (~0 students), Partial: 45% (~82 students), Perfect: 55% (~100 students)

**Problem 9**
Zero: 13% (~25 students), Partial: 63% (~114 students), Perfect: 24% (~43 students)

**Problem 10**
Zero: 1% (~2 students), Partial: 76% (~138 students), Perfect: 23% (~41 students)
The Mid-term exam is now a pedagogical device.

You can recover 50% of your missed points by showing that you have debugged and repaired your knowledge base.

For each question on your exam where points were deducted:

Write about 4 sentences, and perhaps an equation or two.

Describe:

* What was the bug in your knowledge base that led to the error?

* How have you repaired your knowledge base so that the error will not happen again?

Please, do not write on your exam.

Please prepare a written document that describes the points above, print it out, and staple it to the first page of your exam. Please place the staple so that we can still read the header information on your exam.

**Turn it in at the beginning of class on Tuesday, 2 December.**
The exam will begin on the next page. Please, do not turn the page until told.

When you are told to begin the exam, please check first to make sure that you have all ten pages, as numbered 1-10 in the bottom-right corner of each page. We wish to avoid copy problems. We will supply a new exam for any copy problems.

The exam is closed-notes, closed-book. No calculators, cell phones, electronics.

Please clear your desk entirely, except for pen, pencil, eraser, a blank piece of paper (for scratch pad use), and an optional water bottle. Please write your name and ID# on the blank piece of paper and turn it in with your exam.

This page summarizes the points for each question, so you can plan your time.

1. (10 pts total, -1 pt each wrong answer, but not negative) MINIMAX WITH ALPHA-BETA PRUNING.
2. (10 pts total, 2 pts each) CSP WITH BACKTRACKING SEARCH.
3. (6 pts total, 2 pts each) CONVERSION TO CNF.
4. (6 pts total, 2 pts each) RESOLUTION OF CLAUSES.
5. (10 pts total, 1 pt each) LOGIC.
6. (8 pts total, 2 pts each) TASK ENVIRONMENT.
7. (10 pts total, 1/2 pt each) SEARCH PROPERTIES.
8. (15 points total, 3 pts each) CONSTRAINT SATISFACTION PROBLEMS.
9. (10 pts total) RESOLUTION THEOREM PROVING.
10. (15 pts total, 3 pts each) STATE-SPACE SEARCH.
1. (10 pts total, -1 pt each wrong answer, but not negative) MINIMAX WITH ALPHA-BETA PRUNING. While visiting Crete, you are challenged by a passing king to what he calls the “Labyrinth Challenge”. The rules are simple: you must make your way through a maze to find the largest prize for yourself. You are given the following map to plan your route:

You will start in the maze at the location labeled START and may travel North (N), South (S), East (E), or West (W). Your goal is to secure the largest, single prize for yourself, represented by the numbers spread across the maze. At four specific intersections (A, B, C, D), the king will be able to close off all but one pathway by closing gates around you, forcing you to take the path he gives you. Backtracking is not allowed. The king acts to minimize your payoff.

1.a. Fill in each blank triangle with its Mini-Max value. Process the game tree left-to-right.

1.b. Cross out each leaf node that will be pruned by Alpha-Beta pruning. Go left-to-right.

1.c. What is the best move for MAX? (write N, W, E, or S) _____N_____  
 **** TURN PAGE OVER AND CONTINUE ON THE OTHER SIDE ****
2. (10 pts total, 2 pts each) CSP WITH BACKTRACKING SEARCH. This question will use CSP algorithms on a 6-queens problem. Variable \( V_i \) represents the row of the queen placed in column \( i \). Each variable \( V_i \) has domain \( D_i = \{1,2,3,4,5,6\} \). The 6-queens problem requires that we place all 6 queens on the board such that none of them can attack each other; so we cannot have 2 queens in the same row, column, or diagonal. At this point we are part-way through the search, so one of the variables already has been assigned as \( V_1 = 3 \) (see figure below).

Show the search state after going all the way down the first branch in the search tree. Stop when you have either (1) found a solution, or (2) fail and need to backtrack.

Use (1) Forward Checking, (2) variable ordering by the Minimum Remaining Values (MRV) heuristic, (3) value ordering by the Least Constraining Value (LCV) heuristic. When there are ties on variable ordering from the MRV heuristic, break the tie by preferring the left-most tied variable. When there are ties on value ordering from the LCV heuristic, break ties by preferring the smallest value. At this point, you are part-way through the search, and you have just failed at \( V_1 = 1 \) and \( V_1 = 2 \), so \( V_1 = 3 \) is the only assignment now. The first one (\( V_1 = 3 \)) is done for you as an example of what the problem asks for and how to fill it in (“Q” means queen).

2.a. Write the next selected variable, or “Fail” if you need to backtrack now: \( V_1 \)

Write the value assigned to the selected variable: \( 3 \)

Write Q where you put your queen. Draw a line through values removed by Forward Checking. You will find it helpful to propagate your previous Qs and lines, but that is not required.

![Diagram showing queen placement and Forward Checking]

Before the first assignment, all variables have MRV=6, and so you choose the left-most, \( V_1 \), as the instructions state. All choices of values for \( V_1 \) remove 10 values from other domains, so you order them smallest-to-largest as the instructions state. You have just failed on \( V_1=1 \) and \( V_1=2 \). Now you have made the assignment \( V1=3 \), as shown.

2.b. (2 pts) Write the next selected variable, or “Fail” if you need to backtrack now: \( V_2 \)

Write the value assigned to the selected variable: \( 6 \)

Write Q where you put your queen. Draw a line through values removed by Forward Checking. You will find it helpful to propagate your previous Qs and lines, but that is not required.

![Diagram showing queen placement and Forward Checking]

Both \( V_2 \& V_3 \) have the same MRV=3, while \( V_4 \) has MRV=4 and \( V_5 \& V_6 \) have MRV=5. We break the tie between \( V_2 \& V_3 \) by preferring the left-most, which is \( V_2 \). At this point, \( D_2 = \{1,5,6\} \). Assigning \( V_2 = 1 \) removes 6 values, and \( V_2 = 5 \) removes 7 values, from unassigned variable domains. Assigning \( V_2 = 6 \) removes only 5, and so \( V_2 = 6 \) is preferred by the LCV heuristic. Remember that the LCV heuristic counts ONLY values that have not previously been removed.
2.c. (2 pts) Write the next selected variable, or “Fail” if you need to backtrack now: **V3**

Write the value assigned to the selected variable: _______ **2** _______

Write Q where you put your queen. Draw a line through values removed by Forward Checking. You will find it helpful to propagate your previous Qs and lines, but that is not required.

V3 has MRV=2, while V4 & V6 have MRV=3 and V5 has MRV=4. So MRV prefers V3. At this point, D3={2,4}. Assigning V3=2 or V3=4 both remove 5 values from unassigned variables domains, so neither is preferred by the LCV heuristic. We break the tie by assigning the least value, V3=2, as stated in the instructions.

2.d. (2 pts) Write the next selected variable, or “Fail” if you need to backtrack now: **V4**

Write the value assigned to the selected variable: _______ **5** _______

Write Q where you put your queen. Draw a line through values removed by Forward Checking. You will find it helpful to propagate your previous Qs and lines, but that is not required.

V4 has MRV=1 while V5 & V6 have MRV=2. So MRV prefers V4. At this point, D4={5}, so there is only one assignment available and it is made.

2.e. (2 pts) Write the next selected variable, or “Fail” if you need to backtrack now: **V5**

Write the value assigned to the selected variable: _______ **1** _______

Write Q where you put your queen. Draw a line through values removed by Forward Checking. You will find it helpful to propagate your previous Qs and lines, but that is not required.

V5 has MRV=1 while V6 has MRV=2. So MRV prefers V5. At this point, D5={1}, so there is only one assignment available and it is made.

**** TURN PAGE OVER AND CONTINUE ON THE OTHER SIDE ****
2.f. (2 pts) Write the next selected variable, or “Fail” if you need to backtrack now: V6

Write the value assigned to the selected variable: 4

Write Q where you put your queen. Draw a line through values removed by Forward Checking. You will find it helpful to propagate your previous Qs and lines, but that is not required.

Vi = 6
Vi = 5
Vi = 4
Vi = 3
Vi = 2
Vi = 1

V6 has MRV=1 while no other variables are available. So MRV prefers V6. At this point, D6={4}, so there is only one assignment available and it is made.

Success!!

In a “real” search, we actually would have succeeded at V1 = 2, as you can see easily by rotating this solution 90 degrees in either direction. But to do so requires backtracking, and that would have made the problem too long and complicated to work in an exam. So, you were asked to start at V1 = 3 instead.

3. (6 pts total, 2 pts each) CONVERSION TO CNF. Convert these expressions to CNF.

3.a. (2 pts) ( A ⇔ ( B ∨ C ) ) ( ( A → B ) ( A → C ) (¬A B C ) )

3.b. (2 pts) ( ( C ∧ D ) ⇒ ¬E ) (¬C ¬D ¬E )

3.c. (2 pts) ( ( A ⇒ B ) ⇒ C ) ( ( A C ) (¬B C ) )

See section 7.5.

It is OK if you put in “∧” and “∨” above. For example, it is OK if you answered 3.a. as ( ( A ∨ ¬B ) ∧ ( A ∨ ¬C ) ∧ (¬A ∨ B ∨ C ) ), and so on.

3.a is sentence S1, and 3.b is sentence S3, of problem 7.20, p. 283, in your textbook, after variable relabeling.

4. (6 pts total, 2 pts each) RESOLUTION OF CLAUSES. Use resolution to resolve the following pairs of clauses, simplify, and write the resulting clause. If no resolution is possible write “None”. If the resolvent simplifies to True write “True.” If more than one resolution is possible, pick any one that you like, and write its simplified result.

4.a. (2 pts) Resolve ( A B C ) with ( B → C → D ) to yield ( A B ¬D)

See section 7.5.

4.b. (2 pts) Resolve ( A B C ) with (¬B → C → D ) to yield True

4.c. (2 pts) Resolve ( A B C ) with ( B C → D ) to yield None
5. (10 pts total, 1 pt each) LOGIC. Which of the following are correct? (T = True, F = False)
Recall that the symbol ”|=“ means ”entails.”

5.a. (1 pt) _______ False |= True. (False ⇒ True) is valid.
5.b. (1 pt) _____ True |= False. (True ⇒ False) is unsatisfiable.
5.c. (1 pt) _______ (A ∧ B) |= (A ⇒ B). (A ∧ B) entails (A = True) and (B = True).
5.e. (1 pt) _______ (A ⇒ B) |= (¬A ∨ B). A & B are both true or both false, so (¬A ∨ B) is true.
5.f. (1 pt) _______ ((A ∧ B) ⇒ C) |= (A ⇒ C) ∨ (B ⇒ C).
5.g. (1 pt) _______ (A ∨ B) ∧ (¬C ∨ ¬D ∨ E) |= (A ∨ B).
5.h. (1 pt) F______ (A ∨ B) ∧ (¬C ∨ ¬D ∨ E) |= (A ∨ B) ∧ (¬D ∨ E).
5.i. (1 pt) T______ (A ∨ B) ∧ (¬A ⇒ B) is satisfiable.
5.j. (1 pt) _______ (A ⇒ B) ∧ (¬A ∨ B) is satisfiable.

6. (8 pts total, 2 pts each) TASK ENVIRONMENT. Your book defines a task environment as a set of four things, with the acronym PEAS. Fill in the blanks with the names of the PEAS components.

Performance (measure) Environment Actuators Sensors

7. (10 pts total, 1/2 pt each) SEARCH PROPERTIES.
Fill in the values of the four evaluation criteria for each search strategy shown. Assume a tree search where b is the finite branching factor; d is the depth to the shallowest goal node; m is the maximum depth of the search tree; C* is the cost of the optimal solution; step costs are identical and equal to some positive ε; and in Bidirectional search both directions using breadth-first search.

Criterion Complete? Time complexity Space complexity Optimal?
Breadth-First Yes O(b^d) O(b^d) Yes
Uniform-Cost Yes O(b^(1+floor(C*/ε))) O(b^(1+floor(C*/ε))) also OK Yes
Depth-First No O(b^m) O(bm) No
Iterative Deepening Yes O(b^d) O(bd) Yes
Bidirectional (if applicable) Yes O(b^(d/2)) O(b^d/2) Yes
8. (15 points total, 3 pts each) CONSTRAINT SATISFACTION PROBLEMS.

You are a map-coloring robot assigned to color this Pennsylvania region map. Adjacent regions must be colored a different color (R=Red, B=Blue, G=Green). The constraint graph is shown.

8.a. (3 pts total) FORWARD CHECKING. Cross out all values that would be eliminated by Forward Checking, after variable LH has just been assigned value G, as shown:

<table>
<thead>
<tr>
<th>AF</th>
<th>DC</th>
<th>LE</th>
<th>LH</th>
<th>LV</th>
<th>PI</th>
<th>PO</th>
<th>SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>B</td>
<td>R</td>
<td>G</td>
<td>B</td>
<td>R</td>
<td>G</td>
<td>B</td>
</tr>
</tbody>
</table>

8.b. (3 pts total) ARC CONSISTENCY. Cross out all values that would be eliminated by Arc Consistency (AC-3 in your book).

<table>
<thead>
<tr>
<th>AF</th>
<th>DC</th>
<th>LE</th>
<th>LH</th>
<th>LV</th>
<th>PI</th>
<th>PO</th>
<th>SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

8.c. (3 pts total) MINIMUM-REMAINING-VALUES HEURISTIC. Consider the assignment below. PO is assigned and constraint propagation has been done. List all unassigned variables that might be selected by the Minimum-Remaining-Values Heuristic: AF, DC, LV, SU.

<table>
<thead>
<tr>
<th>AF</th>
<th>DC</th>
<th>LE</th>
<th>LH</th>
<th>LV</th>
<th>PI</th>
<th>PO</th>
<th>SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>B</td>
<td>G</td>
<td>R</td>
<td>G</td>
<td>B</td>
<td>R</td>
<td>G</td>
</tr>
</tbody>
</table>

8.d. (3 pts total) DEGREE HEURISTIC. Consider the assignment below. (It is the same assignment as in problem 8.c. above.) PO is assigned and constraint propagation has been done. List all unassigned variables that might be selected by the Degree Heuristic: LH, AF.

<table>
<thead>
<tr>
<th>AF</th>
<th>DC</th>
<th>LE</th>
<th>LH</th>
<th>LV</th>
<th>PI</th>
<th>PO</th>
<th>SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>B</td>
<td>G</td>
<td>R</td>
<td>G</td>
<td>B</td>
<td>R</td>
<td>G</td>
</tr>
</tbody>
</table>

8.e. (3 pts total) MIN-CONFLICTS HEURISTIC. Consider the complete but inconsistent assignment below. AF has just been selected to be assigned a new value during local search for a complete and consistent assignment. What new value would be chosen below for AF by the Min-Conflicts Heuristic? R.

<table>
<thead>
<tr>
<th>AF</th>
<th>DC</th>
<th>LE</th>
<th>LH</th>
<th>LV</th>
<th>PI</th>
<th>PO</th>
<th>SU</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>G</td>
<td>G</td>
<td>R</td>
<td>G</td>
<td>B</td>
<td>B</td>
<td>G</td>
</tr>
</tbody>
</table>
9. (10 pts total) RESOLUTION THEOREM PROVING. Solve this puzzle as a logic agent.

“Three girls took part in a race. Alison (A) finished before Betty (B) but behind Clare (C). What was the finishing order?”

Obviously, the finishing order was C, A, B. You will use resolution to prove this fact.

You create 9 propositional variables, A/B/C1/2/3, to indicate that A, B, or C finished in place 1, 2, or 3. Your Knowledge Base is:

(A1 A3) (A2 B3) (C1 C2)

A bright and clever student has constructed a shorter proof than I was able to find:

You will use resolution to prove this fact.

You create 9 propositional variables, A/B/C1/2/3, to indicate that A, B, or C finished in place 1, 2, or 3. Your Knowledge Base is:

(A1 A3) (A2 B3) (C1 C2)

Your goal sentence is (C1 A2 B3) so your negated goal sentence is:

(C1 A2 B3)

Produce a resolution proof that the finishing order was (C1 A2 B3).

Repeatedly choose two clauses, write one clause in the first blank space on a line, and the other clause in the second. Apply resolution to them. Write the resulting clause in the third blank space, and insert it into the knowledge base. Continue until ( ). Use more lines if needed.

Resolve (A1 A2) and (¬A1 ¬A3) to give (A2 ¬A3).

Resolve (A2 ¬A3) and (A2 A3) to give (A2).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2 ¬A3) and (A2 A3) to give (A2).

Resolve (A1 A2) and (¬A1 ¬A3) to give (A2 ¬A3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Resolve (A2) and (¬C1 ¬A2 ¬B3) to give (¬C1 ¬B3).

Resolve (A2) and (¬A1 ¬A2) to give (¬A1).

Keep reducing the number of literals per clause, until you reach zero. Most resolvents have only one literal; two have two, and one (the last) has zero (goal).

Extra lines or steps are OK as long as your proof is correct. It is expected you will simplify expressions. E.g., if you resolved (¬B V ¬D) and (B V ¬D) to give (¬D V ¬D), of course you would simplify it to (¬D). You will simplify the expressions as you go, you don’t need a separate step.
10. (15 pts total, 3 pts each) STATE-SPACE SEARCH. Execute Tree Search through this graph (i.e., do not remember visited nodes, so repeated nodes are possible). It is not a tree, but pretend you don’t know that. The successors of each node are indicated by the arrows out of that node. Successors are returned in left-to-right order. The successors of S are A/B/C, and of C are F/C, in that order. C is a successor of itself.

For each search strategy below, indicate the order in which nodes are expanded (i.e., its children are generated), ending with the goal node that is found. The first one is done for you as an example.

10.a. DEPTH FIRST SEARCH.
S A D G1

10.b. (3 pts, -1 for each wrong answer, but not negative) UNIFORM COST SEARCH.
S C B A F C E D F C G1

10.c. (3 pts, -1 for each wrong answer, but not negative) GREEDY (BEST-FIRST) SEARCH.
S C C C C C C C C C C etc.

10.d. (3 pts, -1 for each wrong answer, but not negative)
S S A B C S A D G1

10.e. (3 pts, -1 for each wrong answer, but not negative) A* SEARCH.
S C B A F C E G2

10.f. (3 pts, -1 for each wrong answer, but not negative)
S C B A F C E G2

Question 10.f will be graded based on your answers to 11.b (for UCS) and 11.e (for A*).

f: find the optimal goal? Yes ___ Why or why not? Step costs are \( \geq \epsilon > 0 \)

If the answer is No, then why not? heuristic is not admissible (at D)

OK to say “Heuristic overestimated” or “Heuristic is too high.”

THIS IS THE END OF THE MID-TERM EXAM