Written by (anonymous ID):

Corrected by (anonymous ID):

Written examination: Solution suggestions
TIN175/DIT411, Introduction to Artificial Intelligence

Question 1 had completely wrong alternatives, and cannot be answered!
Therefore, the grade limits was lowered by 1 point!

Tuesday 13 February 2018, 8:30–12:30

Examiner: Peter Ljunglöf

This examination consists of 12 questions, each giving 1 point. In total you can get 12 points. Questions 11–12 are slightly more advanced, and they are necessary for getting a higher grade.

Grades:

To get grade 3 / G / pass you need
at least ~~7~~ 6 points.

To get grade 4 you need
at least ~~9~~ 8 points, including
at least 1 point from questions 11–12.

To get grade 5 / VG / distinction you need
at least ~~11~~ 10 points, including
at least 2 points from questions 11–12.

Note: if you are a student from TIN174 or DIT410
you can only get grade 3 / G / pass.

Tools: Paper and pencil.

No extra books, papers or calculators.
No red pen!

Notes: Answer every question directly on the question paper, and write your ID number at the top of every paper.

If you have any extra papers with associated calculations, you should hand in them too.

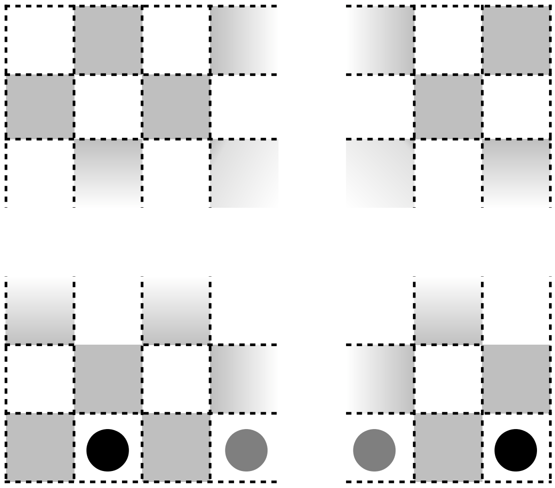
Remember to write legibly!

https://www.smbc-comics.com/comic/ai-2

Written by (anonymous ID):

Corrected by (anonymous ID):

Checkers board problem



n black checkers pieces occupy the white squares at the bottom row of an $2n \times 2n$ checkers board. The pieces must be moved to the top row but in reverse order; so that piece i that starts in square $(2i, 1)$ must end up in square $(2n-2i+1, 2n)$. At each time step, all n pieces move simultaneously. A piece can move one square diagonally (up/down, left/right), or stay where it is. Two pieces cannot occupy the same square.

1. State space and branching factor

Note: This question has the wrong answer suggestions, so the pass limit of the exam was reduced

What is the size of the state space (approximately)?

correct?

☐ n ☐ n^2 ☐ n^3 ☐ n^4 ☐ 2^n ☐ $n + n^2$

Correct answer is approximately n^{2n}

What is the branching factor (approximately)?

☐ 5 ☐ n ☐ $5n$ ☐ $5 + n$ ☐ n^2 ☐ $5n^2$

Correct answer is approximately 5^n

2. Dominating and admissible heuristics

correct?

If we assume that there are no other pieces on the board, then the following is a nontrivial admissible heuristic for moving piece i to its goal location **$(2n-2i+1, 2n)$**:

$h_i = 2n - y_i =$ the number of vertical rows between the piece's current row and the top row

Describe an admissible heuristic h for the complete problem with n pieces, which dominates every heuristics h_i $(1 \le i \le n)$.

The max of all h_i: $\max_i(h_i) = 2n - \min_i(y_i)$

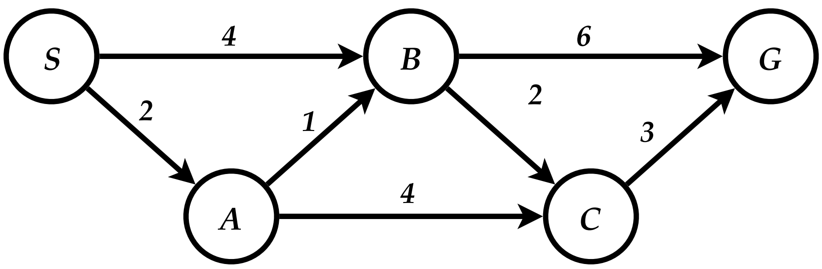
Note that the sum of all h_i is not admissible, since all bricks move simultaneously and this is considered one move.

Written by (anonymous ID):

Corrected by (anonymous ID):

Heuristic search

The following search problem has only three states, and three directed edges. *S* is the start node and G is the goal node. To the right, three different heuristic functions are defined, h_1, h_2 and h_3.



	h_1	h_2	h_3
S	8	7	3
A	3	4	0
B	7	5	4
C	2	2	0
G	0	0	0

3. Which heuristics is admissible?

Which (if any) of the heuristics are admissible? Check all that apply.

correct?

☐ h_1 ✅ h_2 ✅ h_3 ☐ none

Note that none of the heuristics are consistent
(e.g., $|h_i(S)–h_i(A)| > \text{cost}(S, A)$ for all h_i)

4. What path is returned?

correct?

What solutions will be returned by (a) **breadth-first tree search**, and (b) **A* tree search** using the h_3 heuristics?

	Breadth-first tree search	A* tree search using using h_3
Final solotion path found	*S B G*	*S A B C G*
Cost of final path	**10**	**8**

A* search will return the optimal path, since h_3 is admissible

Heuristic search (cont.)

5–6. What would a search algorithm do?

The generic search algorithm uses a frontier of nodes that are waiting to be expanded. At each iteration, one node is removed from the frontier, and its neighbors are added to the frontier.

Fill the table below, for (a) **A* tree search** using the h_1 heuristics, and (b) for **uniform-cost tree search**.

Important: Write the frontier in order with the head to the left, and write the f-values together with the nodes in the frontier, like this: "**X(3) Y(3) Z(7)**". Break all ties in the frontier alphabetically (this is why **X** came before **Y** in the example before).

	Question 5. A* tree search using h_1	*correct?*	Question 6. Uniform-cost tree search	*correct?*
Initial frontier	*S*(8)		*S*(0)	
Node removed from frontier	*S*		*S*	
Nodes added to frontier	*A*, *B*		*A*, *B*	
Frontier after one iteration	*A*(5 = 2+3) *B*(11 = 4+7)		*A*(2) *B*(4)	
Node removed from frontier	*A*		*A*	
Nodes added to frontier	*B*, *C*		*B*, *C*	
Frontier after two iterations	*C*(8 = 6+2) *B*(10 = 3+7) *B*(11 = 4+7)		*B*(3) *B*(4) *C*(6)	
Node removed from frontier	*C*		*B*	
Nodes added to frontier	*G*		*C*, *G*	
Frontier after three iterations	*G*(9 = 9+0) *B*(10 = 3+7) *B*(11 = 4+7)		*B*(4) *C*(5) *C*(6) *G*(10)	
Node removed from frontier	*G*		*B*	

I split the A* *f*-values into *cost+heuristics*, to make it clearer.

If you used tree search where duplicate frontier nodes are updated instead of added again, that was ok too. If you showed graph search, you got 1/2 point.

If you used the cost of the latest edge only, you got 0 points.
If you used breadth-first search in question 6, you got 0 points.

Written by (anonymous ID):

Corrected by (anonymous ID):

Map colouring

The map to the right shows the Baltic countries with their neighbours. We want to paint them using only the colours red, green, blue and yellow, so that no bordering countries have the same colour.

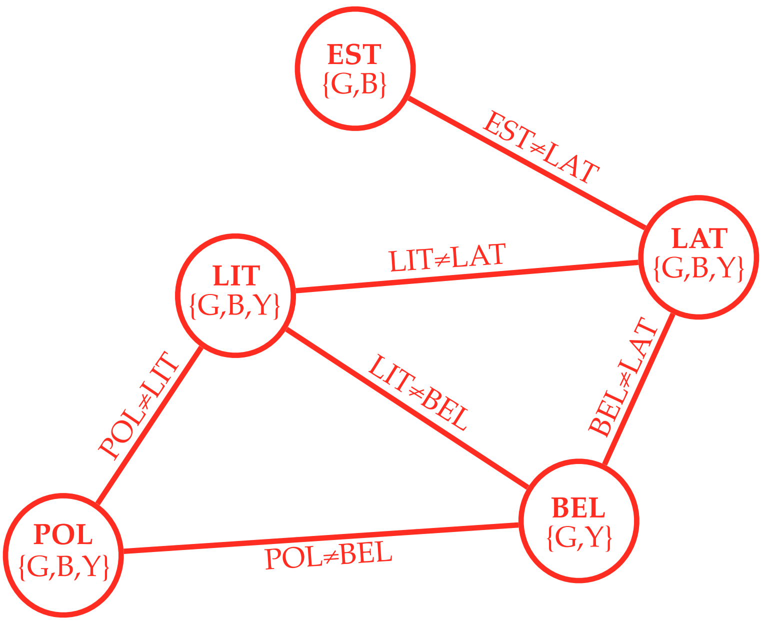
Russia has already decided that it wants to be red, and since is is bordering every other country, noone else can be red. This leaves us with the possible colours (G)reen, (B)lue and (Y)ellow for the variables EST, LAT, LIT, BEL and POL.

Furthermore, ***Estonia does not want to be yellow***, and ***Belarus does not want to be blue***!
(Don't ask why, but it might be because of some people's fondness of constrait satisfaction problems).

7. Formulate this problem as a CSP

Draw the constraint graph below, and write the constraints at the graph edges.
Enforce all unary constraints by reducing the initial domains.
Circle every graph node and write the resulting domain inside the circle.

correct?



If you addded a node for Russia (with the correct constraints, and only Red as the domain), that was also considered correct.

Written by (anonymous ID):

Corrected by (anonymous ID):

Map colouring (cont.)

8. Which variable should we choose next?

correct?

Now we want to colour the rest of the countries using the backtracking algorithm.
We need to select a variable which we can assign a value. Which one should we choose?

a) Which variable(s) does the *Minimum Remaining Values** heuristic suggest that you try next? (*i.e, *choose the variable with the fewest legal values*). Check all that apply:

☑ BEL ☑ EST ☐ LAT ☐ LIT ☐ POL

b) Which variable(s) does the *Degree Heuristic** suggest that you try next? (*i.e, *choose the variable with most constraints on remaining variables*). Check all that apply:

☑ BEL ☐ EST ☑ LAT ☑ LIT ☐ POL

It is also ok to only answer BEL, because then we assume that we only choose between MRV variables

9. Make the graph arc consistent

correct?

Assume that we colour Poland yellow.
What are the resulting domains after enforcing arc consistency on the graph?
(If you want you can show your calculations below).

Here's one possibility:

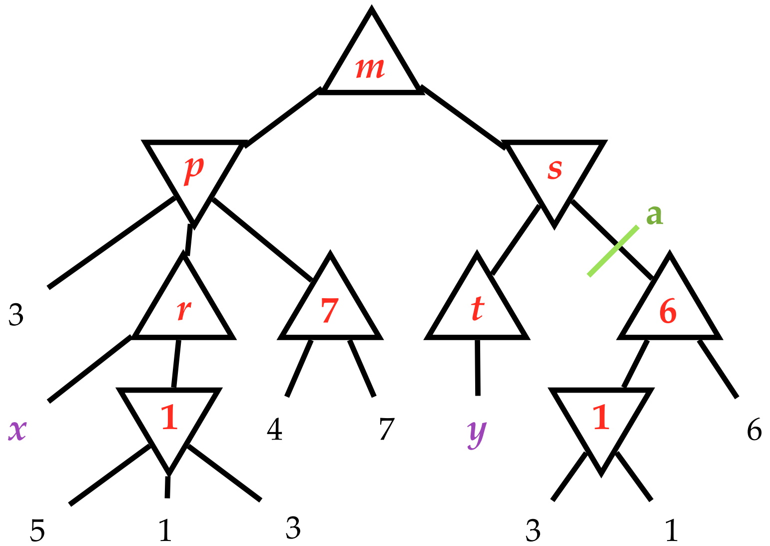
1. POL=Y implies BEL=G
2. POL=Y and BEL=G implies LIT=B
3. BEL=G and LIT=B implies LAT=Y

Note: This is a simplification of how the AC-3 algorithm would make the graph arc consistent.

Variable	*Domain after arc consistency*
EST	B, G
LAT	Y
LIT	B
BEL	G
POL	Y

Minimax search

In the following minimax game tree, △ are maximising nodes, and ▽ are minimising nodes.
Note that two leaves are unspecified and marked x and y.



4. When will the minimax value be x or y?

a) For which values of x will the minimax value of the topmost node be x?
(Assuming that $y = 0$)

correct?

$1 \leq x \leq 3$, because: $r = \max(x, 1)$ implies $x \geq 1$
$p = \min(r, 3)$ implies $x \leq 3$
$m = \max(p, s)$ implies $x \geq 0$ $(s = t = y = 0)$

b) For which values of y will the minimax value of the topmost node be y?
(Assuming that $x = 0$)

$1 \leq y \leq 6$, because: $s = \min(y, 6)$ implies $y \leq 6$
$m = \max(p, s)$ implies $y \geq 1$ $(p = \min(3, 1, 7) = 1)$

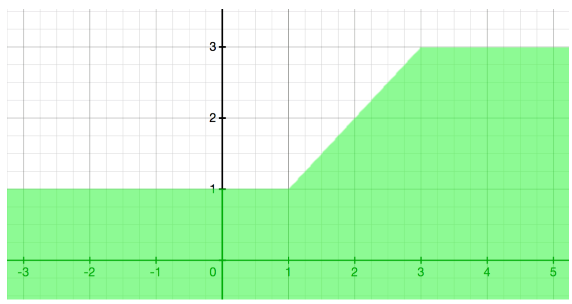
11.* When will alpha-beta prune a branch?

For which values of x and y will minimax with α-β pruning *not* consider branch **a**?
(In other words: For which values of x and y will branch **a** be pruned away?)

correct?

$x < 1 \;\&\; y \leq 1 \;\vee\; 3 < x \;\&\; y \leq 3 \;\vee\; 1 \leq x \leq 3 \;\&\; y \leq x$

Alternatively:
$y \leq \min(\max(x, 1), 3) = \max(\min(x, 3), 1)$



Written by (anonymous ID): ....

Corrected by (anonymous ID): ....

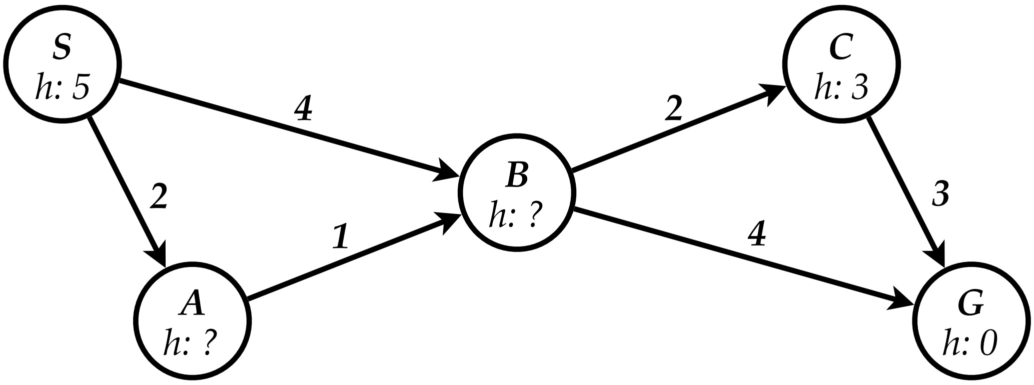
Non-consistent heuristics

12.* Create an admissible but non-consistent heuristics

correct?

Below is a search graph with starting state S and goal state G.
It also has a heuristics h, but unfortunately it is not specified in all places.

For which values of $h(A)$ and $h(B)$ will the heuristics be admissible but not consistent?



Admissibility requires:

$0 \leq h_B \leq 4 \quad \& \quad 0 \leq h_A \leq 5$

Consistency requires:

$h_S-2 \leq h_A \leq h_S+2 \quad \& \quad h_S-4 \leq h_B \leq h_S+4 \quad \& \quad h_C-2 \leq h_B \leq h_C+2 \quad \& \quad h_A-1 \leq h_B \leq h_A+1$

$\Leftrightarrow \quad 3 \leq h_A \leq 7 \quad \& \quad 1 \leq h_B \leq 9 \quad \& \quad 1 \leq h_B \leq 5 \quad \& \quad h_A-1 \leq h_B \leq h_A+1$

So, admissibility but non-consistency means:

$0 \leq h_A \leq 5 \quad \& \quad 0 \leq h_B \leq 4 \quad \& \quad \neg(3 \leq h_A \leq 7 \;\&\; 1 \leq h_B \leq 9 \;\&\; 1 \leq h_B \leq 5 \;\&\; h_A-1 \leq h_B \leq h_A+1)$

which can be simplified to:

$0 \leq h_A \leq 5 \quad \& \quad 0 \leq h_B \leq 4$

$\& \quad (h_A < 3 \;\vee\; h_A > h_B + 1)$