Winter 2018 COMP424 Assignment 1 Solutions

Question 1 35 marks

a) 20 marks

i) *****BFS***** 5 marks

8 nodes visited in search tree, 4 nodes in solution path

1 4 2
5 3 0
->
1 4 2
5 0 3
->
1 0 2
5 4 3
->
0 1 2
5 4 3
->

ii) (same as BFS in this case) 5 marks

8 nodes visited in search tree, 4 nodes in solution path

iii) *****DFS***** 5 marks

189 nodes visited in search tree, 164 nodes in solution path

[1, 4, 2, 5, 3, 0] -> [1, 4, 0, 5, 3, 2] -> [1, 0, 4, 5, 3, 2] -> [0, 1, 4, 5, 3, 2] -> [5, 1, 4, 0, 3, 2] -> [5, 1, 4, 3, 0, 2] -> [5, 0, 4, 3, 1, 2] -> [5, 4, 0, 3, 1, 2] -> [5, 4, 2, 3, 1, 0] -> [5, 4, 2, 3, 0, 1] -> [5, 4, 2, 0, 3, 1] -> [0, 4, 2, 5, 3, 1] -> [4, 0, 2, 5, 3, 1] -> [4, 2, 0, 5, 3, 1] -> [4, 2, 1, 5, 3, 0] -> [4, 2, 1, 5, 0, 3] -> [4, 0, 1, 5, 2, 3] -> [4, 1, 0, 5, 2, 3] -> [4, 1, 3, 5, 2, 0] -> [4, 1, 3, 5, 0, 2] -> [4, 0, 3, 5, 1, 2] -> [4, 3, 0, 5, 1, 2] -> [4, 3, 2, 5, 1, 0] -> [4, 3, 2, 5, 0, 1] -> [4, 3, 2, 0, 5, 1] -> [0, 3, 2, 4, 5, 1] -> [3, 0, 2, 4, 5, 1] -> [3, 2, 0, 4, 5, 1] -> [3, 2, 1, 4, 5, 0] -> [3, 2, 1, 4, 0, 5] -> [3, 0, 1, 4, 2, 5] -> [3, 1, 0, 4, 2, 5] -> [3, 1, 5, 4, 2, 0] -> [3, 1, 5, 4, 0, 2] -> [3, 0, 5, 4, 1, 2] -> [0, 3, 5, 4, 1, 2] -> [4, 3, 5, 0, 1, 2] -> [4, 3, 5, 1, 0, 2] -> [4, 3, 5, 1, 2, 0] -> [4, 3, 0, 1, 2, 5] -> [4, 0, 3, 1, 2, 5] -> [4, 2, 3, 1, 0, 5] -> [4, 2, 3, 0, 1, 5] -> [0, 2, 3, 4, 1, 5] -> [2, 0, 3, 4, 1, 5] -> [2, 1, 3, 4, 0, 5] -> [2, 1, 3, 0, 4, 5] -> [0, 1, 3, 2, 4, 5] -> [1, 0, 3, 2, 4, 5] -> [1, 3, 0, 2, 4, 5] -> [1, 3, 5, 2, 4, 0] -> [1, 3, 5, 2, 0, 4] -> [1, 3, 5, 0, 2, 4] -> [0, 3, 5, 1, 2, 4] -> [3, 0, 5, 1, 2, 4] -> [3, 2, 5, 1, 0, 4] -> [3, 2, 5, 0, 1, 4] -> [0, 2, 5, 3, 1, 4] -> [2, 0, 5, 3, 1, 4] -> [2, 1, 5, 3, 0, 4] -> [2, 1, 5, 0, 3, 4] -> [0, 1, 5, 2, 3, 4] -> [1, 0, 5, 2, 3, 4] -> [1, 5, 0, 2, 3, 4] -> [1, 5, 4, 2, 3, 0] -> [1, 5, 4, 2, 0, 3] -> [1, 5, 4, 0, 2, 3] -> [0, 5, 4, 1, 2, 3] -> [5, 0, 4, 1, 2, 3] -> [5, 2, 4, 1, 0, 3] -> [5, 2, 4, 0, 1, 3] -> [0, 2, 4, 5, 1, 3] -> [2, 0, 4, 5, 1, 3] -> [2, 1, 4, 5, 0, 3] -> [2, 1, 4, 5, 3, 0] -> [2, 1, 0, 5, 3, 4] -> [2, 0, 1, 5, 3, 4] -> [0, 2, 1, 5, 3, 4] -> [5, 2, 1, 0, 3, 4] -> [5, 2, 1, 3, 0, 4] -> [5, 0, 1, 3, 2, 4] -> [0, 5, 1, 3, 2, 4] -> [3, 5, 1, 0, 2, 4] -> [3, 5, 1, 2, 0, 4] -> [3, 5, 1, 2, 4, 0] -> [3, 5, 0, 2, 4, 1] -> [3, 0, 5, 2, 4, 1] -> [0, 3, 5, 2, 4, 1] -> [2, 3, 5, 0, 4, 1] -> [2, 3, 5, 4, 0, 1] -> [2, 0, 5, 4, 3, 1] -> [0, 2, 5, 4, 3, 1] -> [4, 2, 5, 0, 3, 1] -> [4, 2, 5, 3, 0, 1] -> [4, 2, 5, 3, 1, 0] -> [4, 2, 0, 3, 1, 5] -> [4, 0, 2, 3, 1, 5] -> [4, 1, 2, 3, 0, 5] -> [4, 1, 2, 0, 3, 5] -> [0, 1, 2, 4, 3, 5] -> [1, 0, 2, 4, 3, 5] -> [1, 2, 0, 4, 3, 5] -> [1, 2, 5, 4, 3, 0] -> [1, 2, 5, 4, 0, 3] -> [1, 0, 5, 4, 2, 3] -> [0, 1, 5, 4, 2, 3] -> [4, 1, 5, 0, 2, 3] -> [4, 1, 5, 2, 0, 3] -> [4, 0, 5, 2, 1, 3] -> [0, 4, 5, 2, 1, 3] -> [2, 4, 5, 0, 1, 3] -> [2, 4, 5, 1, 0, 3] -> [2, 4, 5, 1, 3, 0] -> [2, 4, 0, 1, 3, 5] -> [2, 0, 4, 1, 3, 5] -> [0, 2, 4, 1, 3, 5] -> [1, 2, 4, 0, 3, 5] -> [1, 2, 4, 3, 0, 5] -> [1, 0, 4, 3, 2, 5] -> [0, 1, 4, 3, 2, 5] -> [3, 1, 4, 0, 2, 5] -> [3, 1, 4, 2, 0, 5] -> [3, 0, 4, 2, 1, 5] -> [0, 3, 4, 2, 1, 5] -> [2, 3, 4, 0, 1, 5] -> [2, 3, 4, 1, 0, 5] -> [2, 3, 4, 1, 5, 0] -> [2, 3, 0, 1, 5, 4] -> [2, 0, 3, 1, 5, 4] -> [0, 2, 3, 1, 5, 4] -> [1, 2, 3, 0, 5, 4] -> [1, 2, 3, 5, 0, 4] -> [1, 0, 3, 5, 2, 4] -> [0, 1, 3, 5, 2, 4] -> [5, 1, 3, 0, 2, 4] -> [5, 1, 3, 2, 0, 4] -> [5, 0, 3, 2, 1, 4] -> [5, 3, 0, 2, 1, 4] -> [5, 3, 4, 2, 1, 0] -> [5, 3, 4, 2, 0, 1] -> [5, 3, 4, 0, 2, 1] -> [0, 3, 4, 5, 2, 1] -> [3, 0, 4, 5, 2, 1] -> [3, 2, 4, 5, 0, 1] -> [3, 2, 4, 5, 1, 0] -> [3, 2, 0, 5, 1, 4] -> [3, 0, 2, 5, 1, 4] -> [3, 1, 2, 5, 0, 4] -> [3, 1, 2, 5, 4, 0] -> [3, 1, 0, 5, 4, 2] -> [3, 0, 1, 5, 4, 2] -> [0, 3, 1, 5, 4, 2] -> [5, 3, 1, 0, 4, 2] -> [5, 3, 1, 4, 0, 2] -> [5, 3, 1, 4, 2, 0] -> [5, 3, 0, 4, 2, 1] -> [5, 0, 3, 4, 2, 1] -> [5, 2, 3, 4, 0, 1] -> [5, 2, 3, 4, 1, 0] -> [5, 2, 0, 4, 1, 3] -> [5, 0, 2, 4, 1, 3] -> [5, 1, 2, 4, 0, 3] -> [5, 1, 2, 0, 4, 3] -> [0, 1, 2, 5, 4, 3] ->

iv) *****IDS***** (path is the same as what bfs found)
No solution found at depth 1
No solution found at depth 2 5 marks
Solution found at depth 3 !
7 nodes visited in search tree, 4 nodes in solution path
1 4 2
5 3 0
->
1 4 2
5 0 3
->
1 0 2
5 4 3
->
0 1 2
5 4 3
->

b) 5 marks

Yes, the Manhattan distance heuristic is still admissible. Assume the following notation:

$h_2(n)$: Manhattan distance heuristic from lecture 3, slide 13
$c_{unit}(n)$: Original cost function in unit-cost version of the puzzle
$c_{new}(n)$: New cost function

We have shown in class that $\forall n . h_2(n) \leq c_{unit}(n)$. Now note that an optimal solution to the new variant of the problem from any state is also a valid (though not necessarily optimal) solution to the original problem, and that the cost under the original problem is the same or lower, because all moves in the new variant cost at least 1. Thus, $\forall n . c_{unit}(n) \leq c_{new}(n)$.

This means that $\forall n . h_2(n) \leq c_{unit}(n) \leq c_{new}(n)$, thus h_2 is an admissible heuristic for this new problem.

c) 5 marks

Define a new heuristic, $h_3(n)$:

$$h_3(n) = \sum_{i=1}^{6} i \times M(i)$$

where $M(i)$ is the Manhattan distance for puzzle piece i . h_3 is admissible for the same reason that h_2 is: it is a relaxed version of the problem in which we ignore any intervening pieces, while respecting the cost of moving a puzzle piece. $h_2(n)=\sum_{i=1}^{6} M(i) \leq \sum_{i=1}^{6} i \times M(i)=h_3(n)$, so h_3 dominates h_2 .

d) 5 marks

No. Consider the following puzzle state:

5 | 1 | 2
  | 4 | 3

The cost to the goal state under the new cost function is 0.5, but the Manhattan distance heuristic gives an estimated cost of 1. Thus, the heuristic is no longer admissible.

Question 2 20 marks, 5 each, for (b), (c), (d), 1 mark if judge correctly, 4 additional mark if the proof is right.

a)

5

Consider a state space which is a chain. That is, each state only has one successor state. Then, DFS has a complexity of $O(n)$, where n is the number of nodes in the search tree, whereas IDS visits $1+2+3+4+\ldots+n=n(1+n)/2=O(n^2)$ nodes in the worst case.

5

b) Breadth-first search is a special case of uniform-cost search (UCS)

If, in UCS, we set the transition cost of every node to be a strictly positive c > 0, then UCS will expand nodes by breadth, exactly like breadth-first search.

UCS will only expand node n_i with cost c_i if all nodes with cost $c_j < c_i$ have been expanded. With a constant cost, all units at level k will have cost kc which is strictly larger than the units at level $k-1$ with cost $(k-1)c < kc$, as such nodes will be expanded by breadth, as in breadth-first search. As such we can say that breadth-first search is a special case of UCS.

5

c) Depth-first search (DFS) is a special case of best-first tree search (BFS)

BFS expands the most promising node at each step according to a heuristic h(n). DFS expands the deepest node at each step. If we use a heuristic which makes the deepest node the most promising, then BFS will act like DFS. For example we could use h(n) = -depth(n) or h(n) = 1/depth(n). As such we can say that DFS is a special case of BFS.

Note that because we are dealing with best-first tree search, each node will have a well-defined depth n.

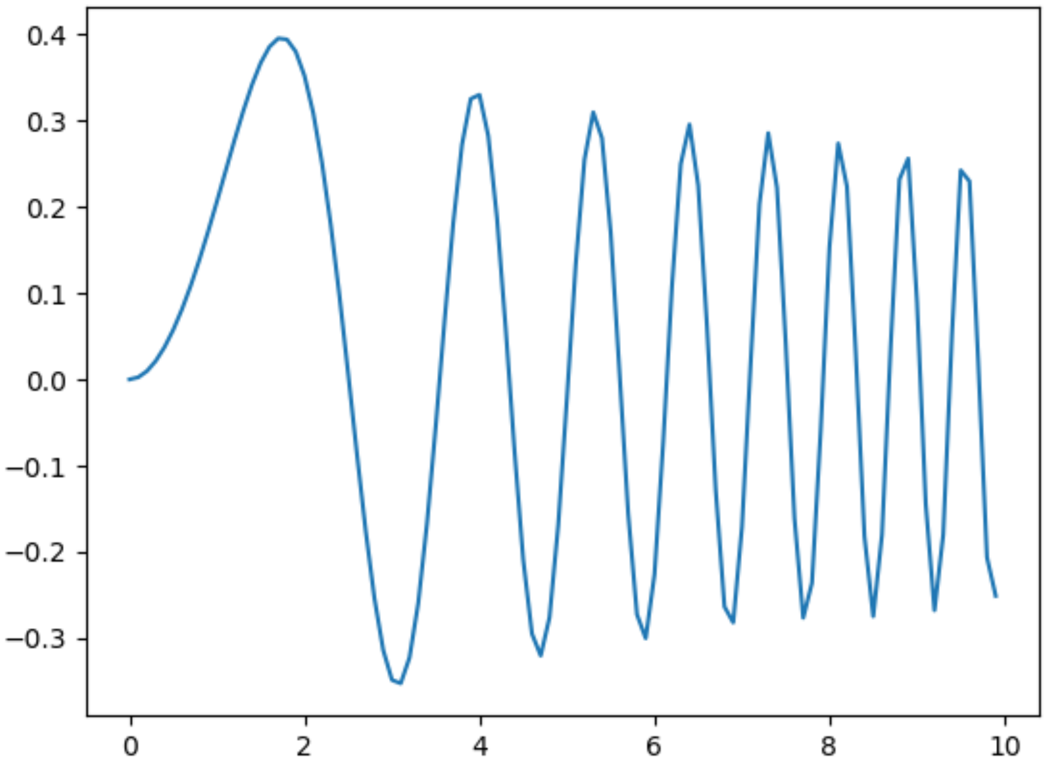
5

d) Uniform-cost search (UCS) is a special case of A* search.

Let g(n) be the cost-so-far. A* expands the most promising node according to its f(n) value which is f(n) =
g(n) + h(n). UCS expands the most promising node according to the cost-so-far, f(n) = g(n).
As such, UCS is a special case of A* search where h(n) = 0.

Question 3 25 marks
a)

Plot of function:



Solutions 15 marks, 5 marks each for finding correct X, Y, # of steps

```
(start, step size, my_x, my_y, nsteps)
```

0	0.01	1.74	0.3960	174
0	0.02	1.74	0.3960	87
0	0.03	1.74	0.3960	58
0	0.04	1.72	0.3958	43
0	0.05	1.75	0.3960	35
0	0.06	1.74	0.3960	29
0	0.07	1.75	0.3960	25
0	0.08	1.76	0.3958	22
0	0.09	1.71	0.3955	19
0	0.10	1.70	0.3951	17
1	0.01	1.74	0.3960	74
1	0.02	1.74	0.3960	37
1	0.03	1.75	0.3960	25
1	0.04	1.72	0.3958	18
1	0.05	1.75	0.3960	15
1	0.06	1.72	0.3958	12

1	0.07	1.77	0.3955	11
1	0.08	1.72	0.3958	9
1	0.09	1.72	0.3958	8
1	0.10	1.70	0.3951	7
2	0.01	1.74	0.3960	26
2	0.02	1.74	0.3960	13
2	0.03	1.73	0.3960	9
2	0.04	1.72	0.3958	7
2	0.05	1.75	0.3960	5
2	0.06	1.76	0.3958	4
2	0.07	1.72	0.3958	4
2	0.08	1.76	0.3958	3
2	0.09	1.73	0.3960	3
2	0.10	1.70	0.3951	3
3	0.01	1.74	0.3960	126
3	0.02	1.74	0.3960	63
3	0.03	1.74	0.3960	42
3	0.04	1.72	0.3958	32
3	0.05	1.75	0.3960	25
3	0.06	1.74	0.3960	21
3	0.07	1.74	0.3960	18
3	0.08	1.72	0.3958	16
3	0.09	1.74	0.3960	14
3	0.10	1.70	0.3951	13
4	0.01	3.96	0.3341	4
4	0.02	3.96	0.3341	2
4	0.03	3.97	0.3338	1
4	0.04	3.96	0.3341	1
4	0.05	3.95	0.3339	1
4	0.06	3.94	0.3331	1
4	0.07	3.93	0.3319	1
4	0.08	3.92	0.3301	1
4	0.09	4.00	0.3298	0
4	0.10	4.00	0.3298	0
5	0.01	5.32	0.3105	32
5	0.02	5.32	0.3105	16
5	0.03	5.33	0.3097	11
5	0.04	5.32	0.3105	8
5	0.05	5.30	0.3095	6
5	0.06	5.30	0.3095	5
5	0.07	5.35	0.3054	5
5	0.08	5.32	0.3105	4
5	0.09	5.36	0.3019	4
5	0.10	5.30	0.3095	3
6	0.01	6.39	0.2961	39
6	0.02	6.38	0.2955	19
6	0.03	6.39	0.2961	13
6	0.04	6.40	0.2955	10
6	0.05	6.40	0.2955	8

6	0.06	6.36	0.2908	6
6	0.07	6.42	0.2905	6
6	0.08	6.40	0.2955	5
6	0.09	6.36	0.2908	4
6	0.10	6.40	0.2955	4
7	0.01	7.31	0.2857	31
7	0.02	7.30	0.2854	15
7	0.03	7.30	0.2854	10
7	0.04	7.32	0.2846	8
7	0.05	7.30	0.2854	6
7	0.06	7.30	0.2854	5
7	0.07	7.28	0.2801	4
7	0.08	7.32	0.2846	4
7	0.09	7.27	0.2753	3
7	0.10	7.30	0.2854	3
8	0.01	8.12	0.2778	12
8	0.02	8.12	0.2778	6
8	0.03	8.12	0.2778	4
8	0.04	8.12	0.2778	3
8	0.05	8.10	0.2734	2
8	0.06	8.12	0.2778	2
8	0.07	8.14	0.2748	2
8	0.08	8.16	0.2646	2
8	0.09	8.09	0.2686	1
8	0.10	8.10	0.2734	1
9	0.01	8.86	0.2713	14
9	0.02	8.86	0.2713	7
9	0.03	8.85	0.2699	5
9	0.04	8.88	0.2679	3
9	0.05	8.85	0.2699	3
9	0.06	8.88	0.2679	2
9	0.07	8.86	0.2713	2
9	0.08	8.84	0.2663	2
9	0.09	8.82	0.2530	2
9	0.10	8.90	0.2560	1
10	0.01	10.00	-0.0689	0
10	0.02	10.00	-0.0689	0
10	0.03	10.00	-0.0689	0
10	0.04	10.00	-0.0689	0
10	0.05	10.00	-0.0689	0
10	0.06	10.00	-0.0689	0
10	0.07	10.00	-0.0689	0
10	0.08	10.00	-0.0689	0
10	0.09	10.00	-0.0689	0
10	0.10	10.00	-0.0689	0

b) The answer will depend on the choice of annealing schedule and step size. 10 marks

General observations: Correctly apply simulated annealing – 2 marks

different temperature T – 2 marks

different α – 2 marks

choice of step size – 2 marks
explanation – 2 marks

- As T increases, the algorithm finds the global maximum at X=1.74 more often. (when T is high, the algorithm explore more and when T is low, the algorithm exploit more)
- Small step sizes require a very large number of steps to converge, but don't always result in better performance. It is difficult to escape from local maxima this way.

Question 4
see next page.

20 marks

Q4. (a) Variables | $R_1=(x_1,y_1)$, $R_2=(x_2,y_2)$, $\cdots$ $R_k=(x_k,y_k)$

4

Domains | $R_i \in \{(1,1),(1,2),\cdots,(1,n),\cdots,(n,n)\}$.

or. $x_i \in \{1,\cdots,n\}$ $y_i \in \{1,\cdots,n\}$ for all $i \in \{1,\cdots,k\}$

Constraints: $x_i \neq x_j$ and $y_i \neq y_j$ for all $i,j \in \{1,..,k\}$ and $i \neq j$

2

(no two rooks can be in the same row or column).

(b)

8

$R_1=\{(1,1),(1,2),\cdots,(3,3)\}$
$R_2=\{(1,1),(1,2),\cdots,(3,3)\}$
$R_3=\{(1,1),(1,2),\cdots,(3,3)\}$

$R_1=(1,1)$
$R_2=\{(1,1),(1,2),\cdots,(3,3)\}$
$R_3=\{(1,1),(1,2),\cdots,(3,3)\}$

$R_2=(1,1)$ Violation $x_2=x_1$

$R_2=(1,2)$ Violation $x_2=x_1$

$R_2=(1,3)$ V $x_2=x_1$

$R_2=(2,1)$ V $y_2=y_1$

$R_2=(2,2)$ ✓
$R_3=\{(1,1),(1,2),\cdots,(3,3)\}$

$R_3=(1,1),(1,2),(1,3)$ V $x_3=x_1$

$R_3=(2,1),(2,2),(2,3)$ V $x_3=x_2$

$R_3=(3,1),(3,2)$ V $x_3=x_1$ V $y_3=y_2$

$R_3=(3,3)$ Solution

(C)

8

$R_1=\{(1,1),(1,2),\cdots,(3,3)\}$
$R_2=\{(1,1),\cdots\cdots(3,3)\}$
$R_3=\{(1,1),\cdots\cdots(3,3)\}$

$R_1=(1,1)$
$R_2=\{(2,2),(2,3),(3,2),(3,3)\}$
$R_3=\{(2,2),(2,3),(3,2),(3,3)\}$

$R_1=(1,1)$, $R_2=\{2,2\}$
$R_3=\{(3,3)\}$

$R_1=(1,1)$, $R_2=\{2,2\}$, $R_3=\{(3,3)$ solution.

4. Alternatively, it's fine if you assume $k \leq n$ and only looking at the row of the rooks.

(a) Variable: $X_1, X_2, \cdots X_k$ $(Y_1, \cdots Y_k)$

Domains: $X_i \in \{1, 2, \cdots, n\}$ for all $i \in \{1, \cdots, k\}$

$(Y_i \in \{1, 2, \cdots, n\})$

Constraints: $X_i \neq X_j$ (and $Y_i \neq Y_j$) for all i, j s.t $i \neq j$, $i, j \in \{1, \cdots, k\}$

(b)

$X_1 = \{1, 2, 3\}$
$X_2 = \{1, 2, 3\}$
$X_3 = \{1, 2, 3\}$

$X_1 = 1$
$X_2 = \{1, 2, 3\}$
$X_3 = \{1, 2, 3\}$

$X_2 = 1$
Violation: $X_2 = X_1$

$X_2 = 2$
$X_3 = \{1, 2, 3\}$

$X_3 = 1$
Violation: $X_3 = X_1$

$X_3 = 2$
Violation: $X_3 = X_2$

$X_3 = 3$
solution

(c)

$X_1 = \{1, 2, 3\}$
$X_2 = \{1, 2, 3\}$
$X_3 = \{1, 2, 3\}$

$X_1 = 1$
$X_2 = \{2, 3\}$
$X_3 = \{2, 3\}$

$X_2 = 2$
$X_3 = \{3\}$

$X_3 = 3$
Solution