50.021 Artificial Intelligence

Quiz 5

Student Name: **Student ID:**

[Q1].

a A* algorithm is based on

A. Breadth-First-Search

B. Depth-First Search

C. Best-First-Search

D. Hill climbing

b Which of the following statements about the search problem are NOT true?

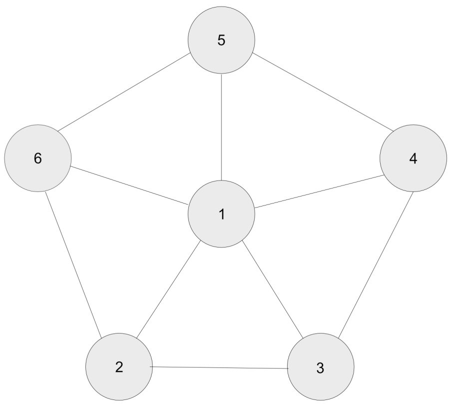
A. The name best-first search is a venerable but inaccurate one. After all, if we could really expand the best node first, it would not be a search at all; it would be a straight march to the goal. All we can do is choose the node that appears to be best according to the evaluation function.

B. Heuristic function h(n) is cheapest path from root to goal node

C. Hill climbing achieves optimal solutions in convex problems otherwise it will find only local optima.

D. Best-First search is a type of informed search, which uses evaluation function returning lowest evaluation to choose the best next node for expansion.

c Consider the map shown below.The regions can be colored using at most four colors so that no two adjacent regions have the same color. How many solutions are there for the



map-coloring problem?

A. 0

B. 120

C. 160

D. 140

d Consider the problem of finding the shortest route through several cities,such that each city is visited only once and in the end return to the starting city (the Travelling Salesman problem). Suppose that in order to solve this problem we use a genetic algorithm, in which genes represent links between pairs of cities. For example, a link between London and Paris is represented by a single gene LP. Let also assume that the direction in which we travel is not important, so that LP = PL.) How many genes will be in the alphabet of the algorithm if the number of cities is 10?

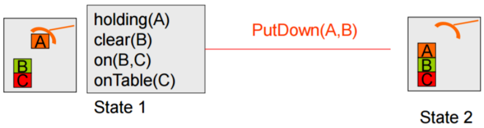
A. 45

B. 90

C. 100

D. 50

e Which are correct encodings of the STRIPS Blocksworld PutDown(A,B) action schema?



A. $(\{holding(A), clear(B)\}, \{on(A,B), handEmpty\}, \{clear(B)\})$

B. $(\{holding(A), clear(B)\}, \{on(A,B), handEmpty, clear(A)\}, \{holding(A), clear(B)\})$

C. $(\{holding(A), clear(B)\}, \{clear(A)\}, \{holding(A), clear(B)\})$

D. $(\{holding(A), clear(B)\}, \{on(A,B), handEmpty, clear(A)\}, \{clear(B), holding(A), on(B,C)\})$

Solution: 1.C 2.B 3.B 4.A 5.B

[Q2]. Refer to the directed graph in fig. 1.

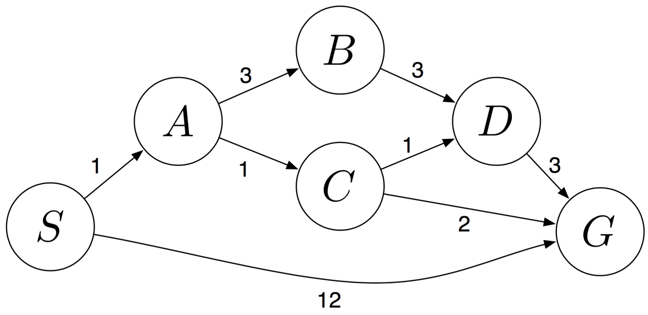


Figure 1: a graph

Answer the following questions,

1. What path would A* graph search, using any *consistent and admissible* heuristic, return for this search problem?

   Solution The optimum path, which is S-A-C-G

2. Consider the heuristics below,

State	$h1$	$h2$
S	5	4
A	3	2
B	6	6
C	2	1
D	3	3
G	0	1

   For each of the question below, answer yes/no and if you answer no, state all the occurrences where the heuristics is not consistent/admissible.

   (a) Is $h1$ admissible?
   Solution No, $h1$ at S is $5 > 4$

   (b) Is $h1$ consistent?
   Solution No, because consistency implies admissability. Since $h1$ is non-admissible, it will not be consistent (but one can be admissible and not consistent). An admissible heuristic says that we cannot overestimate when getting to a particular goal node. But, a consistent heuristic says that you can't overestimate when getting to *any* node. Since goal node is the same node, a consistent heuristic is admissible. However admissibility only guarantees this property for one node, admissible does not imply consistency.

   (c) Is $h2$ admissible?
   Solution No, $h2$ at G is 1 which is obviously wrong.

   (d) Is $h2$ consistent? **Solution** No, since $h_2(G) > 0$.

3. Using $h2$ as the heuristic, perform Greedy Best First search from S. What path would it return?

   Solution S - G

4. If $h2$ is inadmissible, modify **at most one** of it such that it is admissible. Otherwise, use $h2$ as is. Perform A* search with $h2$. Show your steps. Is it the optimum path?

   Change $h_2(G) = 0$ to make h_2 admissible.

   The steps: $[SA(3), SG(12)]$

$[SAC(3), SAB(10), SG(12)]$
$[SACG(4), SACD(6), SAB(10), SG(12)]$

Final path is $S - A - C - G$.

[Q3]. This morning, two teaching assistants: Yuqi and Natalie are set to schedule five tasks in four one-hour slots: 8am-9am, 9am-10am, 10am-11am, 11am-12pm. The tasks are:

(F) Pick up *food* for some research seminar (takes 1 hour)
(H) Prepare *homework* questions for the beloved AI class (takes 2 hours)
(Q) Mark the long overdue *quiz* from last week (takes 1 hour)
(S) Lead a research *seminar* (takes 1 hour)
(O) Standby for *office hour* consultations (takes 2 hours)

There are 10 constraints for the schedule:

- (a) In any given time slot, each TA can do at most 1 task (F, H, Q, A, O)
- (b) The marking of quizzes (Q) should be before the office hour (O) in case some students asked about it.
- (c) The food (F) should be picked up before the seminar (S)
- (d) The seminar (S) should be finished before 10am
- (e) Yuqi is going to deal with the food pickup (F) since he has 10$ grab-taxi discount
- (f) The TA not leading the seminar (S) should still attend the seminar, and hence cannot perform other tasks (F, H, Q, O)
- (g) The seminar leader (S) does not do the office hour consultation (O)
- (h) The TA who standby for office hour (O) must also mark the quiz (Q)
- (i) Preparing homework questions (H) takes 2 consecutive hours, hence should start latest at 10am
- (j) Office hour (O) takes 2 consecutive hours, hence should start latest at 10am

To formulate this problem as a CSP, we use the variables F, H, Q, S, O. Note that the domain for each variable in this case is two-dimensional: $V = (V_n, V_t)$, where V_n = TA's name initial and V_t = starting time of the task, and $V = \{F, H, Q, S, O\}$. We use the TA's initials, Y (Yuqi) and N (Natalie). So for example, if we want to assign Yuqi to mark quizzes at 9am, we write it as $Q_n = Y, Q_t = 9$ or shorter as: $Q = (Y, 9)$. We will use below shorthand notation $Q = Y9$ for $Q = (Y, 9)$. Similarly, $Q \in \{Y9, N9\}$ means that the variable Q can take one of the two values from that set (e.g. as the result of forward checking or constrain propagation.). There are eight possible domains for each of the five variables V:$Y8, Y9, Y10, Y11, N8, N9, N10, N11$. Answer the following questions,

1. What is the size of the state-space for this CSP?
   Solution 8^5
2. Which of the statements above *include unary* constraints?
   (d), (e), (i), (j). Constraints (i) and (j) are both unary and binary in a single sentence.
3. In the table below, enforce all unary constraints by crossing out domains that are not possible.

F	Y8	Y9	Y10	Y11	N8	N9	N10	N11
H	Y8	Y9	Y10	Y11	N8	N9	N10	N11
Q	Y8	Y9	Y10	Y11	N8	N9	N10	N11
S	Y8	Y9	Y10	Y11	N8	N9	N10	N11
O	Y8	Y9	Y10	Y11	N8	N9	N10	N11

Solution

F	Y8	Y9	Y10	Y11	~~N8~~	~~N9~~	~~N10~~	~~N11~~
H	Y8	Y9	Y10	~~Y11~~	N8	N9	N10	~~N11~~
Q	Y8	Y9	Y10	Y11	N8	N9	N10	N11
S	Y8	Y9	~~Y10~~	~~Y11~~	N8	N9	~~N10~~	~~N11~~
O	Y8	Y9	Y10	~~Y11~~	N8	N9	N10	~~N11~~

4. After enforcing the unary constraints, assume we select the variable S and assign the value Y9 to it. Perform forward checking by crossing out the appropriate domains from each variables.

F	Y8	Y9	Y10	Y11	N8	N9	N10	N11
H	Y8	Y9	Y10	Y11	N8	N9	N10	N11
Q	Y8	Y9	Y10	Y11	N8	N9	N10	N11
S	Y8	Y9	Y10	Y11	N8	N9	N10	N11
O	Y8	Y9	Y10	Y11	N8	N9	N10	N11

Solution

F	Y8	~~Y9~~	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
H	~~Y8~~	~~Y9~~	Y10	~~Y11~~	~~N8~~	~~N9~~	N10	~~N11~~
Q	Y8	~~Y9~~	Y10	Y11	N8	~~N9~~	N10	N11
S	~~Y8~~	Y9	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
O	~~Y8~~	~~Y9~~	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	N10	~~N11~~

5. Based on the results in the previous part (4), what variable will we choose to assigned next based on the MRV heuristics (breaking ties alphabetically by choosing lower **variable** letter first)? Assign the first possible value to this variable, and perform forward checking, give the results by crossing out values in the table below. Have we arrived at a dead end? (i.e. any of the variable domains become empty?)

F	Y8	Y9	Y10	Y11	N8	N9	N10	N11
H	Y8	Y9	Y10	Y11	N8	N9	N10	N11
Q	Y8	Y9	Y10	Y11	N8	N9	N10	N11
S	Y8	Y9	Y10	Y11	N8	N9	N10	N11
O	Y8	Y9	Y10	Y11	N8	N9	N10	N11

Solution Variable F gets selected, and gets assigned value Y8.

F	Y8	~~Y9~~	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
H	~~Y8~~	~~Y9~~	Y10	~~Y11~~	~~N8~~	~~N9~~	N10	~~N11~~
Q	~~Y8~~	~~Y9~~	Y10	Y11	N8	~~N9~~	N10	N11
S	~~Y8~~	Y9	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
O	~~Y8~~	~~Y9~~	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	N10	~~N11~~

No, there's no empty domain.

6. We return to the result from enforcing the unary constraints in part (3). From your table in part (3), select the variable S and assign the value Y9. Enforce arc consistency and cross out invalid domain values in the table below.

F	Y8	Y9	Y10	Y11	N8	N9	N10	N11
H	Y8	Y9	Y10	Y11	N8	N9	N10	N11
Q	Y8	Y9	Y10	Y11	N8	N9	N10	N11
S	Y8	Y9	Y10	Y11	N8	N9	N10	N11
O	Y8	Y9	Y10	Y11	N8	N9	N10	N11

Solution

F	Y8	~~Y9~~	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
H	~~Y8~~	~~Y9~~	Y10	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
Q	~~Y8~~	~~Y9~~	~~Y10~~	~~Y11~~	N8	~~N9~~	~~N10~~	~~N11~~
S	~~Y8~~	Y9	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	~~N10~~	~~N11~~
O	~~Y8~~	~~Y9~~	~~Y10~~	~~Y11~~	~~N8~~	~~N9~~	N10	~~N11~~

7. Check your answer in part (6). Does any solution exist? Provide the solution(s) if any.

Solution Arc consistency along this path gives one solution, F:Y8, H:Y10, Q:N8, S: Y9, O:N10

8. Express the following additional constraints in terms of variables F,H,Q,S,O and its domains:
   - The homework preparation (H) cannot have the same starting time as variable S.
   - The homework preparation takes 2 hours. Express this as a binary constraint. Show this binary constraint only for interaction with variable S (as for all other variables it would look very similar).

   For this question you can express the domain of each variable V in two dimensions: $V = (V_n, V_t)$, where V_n is the TA initials and V_t is the time that a task starts. So for example, if we want to assign Yuqi to mark quizzes at 9am, we write it as $Q_n = Y, Q_t = 9$ or $Q = (Y, 9)$.

   Solution:

   - $H_t \neq S_t$
   - $H_t + 1 \neq S_t$