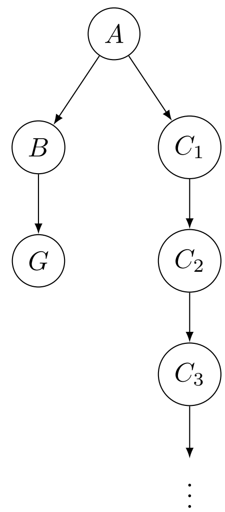
Question 1: Inadmissible Heuristic Affects Completeness [50 points]

We saw in the class that if the heuristic function h is inadmissible, A* search may find a suboptimal goal. We now show that A* may not even be complete, namely it may not terminate even if a goal with small cost exists. Consider the following graph where the right branch goes on forever: where A is the initial state



and G is the only goal state. Let

$$cost(A, B) = 1/2, \;\; cost(B, G) = 1/2$$

so the solution path from A to G has a cost of 1. Furthermore, let

$$cost(A, C_1) = 1 \;\; cost(C_1, C_2) = 1/2 \;\; cost(C_2, C_3) = 1/4 \;\; cost(C_3, C_4) = 1/8 \;\; \dots$$

so that the costs decrease by half toward the right.

Suppose for all states s the heuristic function $h(s) = 0$ *except* that $h(B) = 100$. This makes h inadmissible. Nonetheless, let us run the A* algorithm with this h.

1. (5 points) By definition, what range of $h(B)$ is considered admissible?

$$0 \leq h(B) \leq h^\star(B) = cost(B, G) = 1/2$$

5 points awarded for correctness. No deduction for not mentioning non-negativity.

2. (20 points) Run the A* algorithm (on slide 9 of part 2 of informed search) by hand for five iterations: that is, you should execute step 5 five times. At the end of *each* iteration, show the following:

   - states in OPEN
   - states in CLOSED

OPEN and CLOSED

Iteration	OPEN	CLOSED
1	B, C_1	A
2	B, C_2	A, C_1
3	B, C_3	A, C_1, C_2
4	B, C_4	A, C_1, C_2, C_3
5	B, C_5	A, C_1, C_2, C_3, C_4

Calculation

n	$g(n)$	$h(n)$	$f(n)$	back pointer
A	0	0	0	-
B	1/2	100	201/2	A
C_1	1	0	1	A
C_2	3/2	0	3/2	C_1
C_3	7/4	0	7/4	C_2
C_4	15/8	0	15/8	C_3
C_5	31/16	0	31/16	C_4

- for each state, show its f, g, h values
- for each state, show its back pointer

OPEN and CLOSED: 10 points. 1 point each.
$h(n)$: 1 point.
$g(n)$ and $f(n)$: 7 points. 0.5 each.
Back pointers: 2 points. -1 point for each incorrect pointer.
Note: some students did not expand A in the first iteration and hence stopped at the 4th iteration in the table. We still give full credits in this case.

3. (5 points) What is $\lim_{i\to\infty} f(C_i)$? Show your derivation.

$$\begin{aligned}
\lim_{i\to\infty} f(C_i) &= \lim_{i\to\infty} (g(C_i) + h(C_i)) \quad \text{(2 points for } f(C_i) \text{ definition)} \\
&= \lim_{i\to\infty} (f(C_{i-1}) + cost(C_{i-1}, C_i) + 0) \\
&= \lim_{i\to\infty} 1 + \sum_{k=1}^{i-1} 2^{-k} \quad \text{(2 points for intermediate derivation)} \\
&= \sum_{i=0}^{\infty} 2^{-i} \\
&= 2 \quad \text{(1 point for the correct result)}
\end{aligned}$$

Full credits will be awarded if the answer is correct and sufficient derivation is given.

4. (5 points) Explain in English why the search will not find G.

   The algorithm will pick the state with the smallest $f(s)$ in each iteration. In this case, C_i will always be picked because the upper bound of $f(C_i)$ is $2 < f(B) = 100.5$. Therefore, B will never be expanded and the algorithm will never reach G.
   5 points awarded for correctness.
   2 points awarded if reason mentioned includes h is inadmissible

5. (10 points) Is there a range of $h(B)$ that is inadmissible but still allows A* search to find G? If yes, give the range; if no, explain why.

   We just need to make sure B will be selected at some time. Thus we have

$$f(B) \leq f(C_i) < 2$$

   which means

$$h(B) = f(B) - g(B) < 2 - 1/2 = 3/2$$

   Therefore

$$1/2 < h(B) < 3/2$$

   Partial Marks: 2 Points for choosing yes as an answer
   Deduction: -5 points if derivation is missing
   -2 points for incorrect or no lower bound
   -5 points for incorrect upper bound

6. (5 points) Is admissible h a sufficient condition, a necessary condition, both, or neither for A* search to find the optimal goal?

   By Question 5, the algorithm will still expand B and thus find the goal state even if $h(B)$ is not admissible. On the other hand, if an admissible heuristic is adapted, A* will eventually find an optimal solution because it will at least underestimate the cost, searching as many paths as possible. Therefore, admissible h is a sufficient but not necessary condition for A* search to find the optimal goal.
   5 points awarded for correctness. If student claims it's neither sufficient nor necessary, and gives a *correct* example (e.g. an infinite graph), we also award full credit. However, no point if student claims it is necessary, or neither sufficient nor necessary and does not give an example, or gives a wrong example.

Question 2: Simulated Annealing [25 points]

Consider a state space consisting of integers. Suppose that the successors of a state x include all integers in $[x-10, x+10]$, including x itself. Consider the score function $f(x) = \max\{4 - |x|, 2 - |x-6|, 2 - |x+6|\}$. It has three peaks with one forming a unique global maximum.

Recall that a random successor y is generated on each iteration of simulated annealing. If this successor is better than the current state x, it is always accepted. If it is not better, it will still be accepted with probability

$$p = \exp\left\{-\frac{|f(x) - f(y)|}{T}\right\}.$$

Recall that to do something with probability p, you generate a random number $z \in [0, 1]$ uniformly and if $z \leq p$ you do it.

Use the temperature cooling scheme

$$T(i) = 2(0.9)^i,$$

where i is the iteration number. E.g., for the first iteration, the temperature will be $T(1) = 2(0.9)^1 = 1.8$.

Perform 8 iterations of simulated annealing using $x = 2$ as your starting point. We already "randomly" picked a successor y at each iteration for you, and also generated a "random number" z at that iteration for you to decide whether to go to an inferior successor, should you need it. These are listed in Table 1. Use these numbers. For each iteration, write down (1) the current point, (2) the temperature (rounded to the nearest thousandth), and (3) the probability of moving to the successor given the successor and the temperature.

Iteration Number	Random Successor y	Random Number z
1	3	0.102
2	1	0.223
3	1	0.504
4	4	0.493
5	2	0.312
6	3	0.508
7	4	0.982
8	3	0.887

Table 1: **Successor States and Random Numbers**

Iteration Number	current x	T	p	new x
1	2	1.800	0.574	3
2	3	1.620	N/A (or 1)	1
3	1	1.458	1 (or N/A)	1
4	1	1.312	0.102	1
5	1	1.181	0.429	2
6	2	1.063	0.390	2
7	2	0.957	0.124	2
8	2	0.861	0.313	2

Table 2: **Solution to Q2**

1 point for format (rounding T to the 3rd decimal place; no deduction on the wrong format of p)
Temperature: 8 points, 1 for each iteration
Probability: 8 points, 1 for each iteration ($p = 1$ can be omitted)
New state: 8 points, 1 for each iteration
NOTE: in the second iteration, deduct one point for calculating $p = 0.291$; no deduction if stating $p = 1$

Question 3: City of Madison Open Data: Street Trees [25 points]

Visit the website `http://data-cityofmadison.opendata.arcgis.com`. Go City Datasets / SUSTAINABILITY to find the "Street Trees" dataset:

```
Street Trees  thumbnail
Shared by CityOfMadison
The different classifications of the tree data are as follows: ID:...
Custom License 9/11/2017 Spatial Dataset 112,511 Rows
```

Once you get into this dataset's webpage, you can zoom in to the map on top of the webpage to see where the trees are. Try find Computer Science Department and see the few trees around the building! There is a "Download" button on the upper right. Download the spreadsheet and take a look. You will see many trees with their attributes.

Suppose the city wants to inspect the trees for disease. An inspector starts from their office, drives a truck to visit each tree once, and goes back to the office at the end. The inspector wants to find the shortest route to do so. Because of traffic and one-way streets, the order the trees are visited affects the total distance the inspector needs to travel. It is therefore reasonable to represent a state by a permutation of the trees. Let there be n trees. For a particular permutation $t_1, \ldots, t_n$, the total distance of that state is defined as

$$f(t_1, \ldots, t_n) = d(O, t_1) + \sum_{i=1}^{n-1} d(t_i, t_{i+1}) + d(t_n, O)$$

where $d(a, b)$ is the driving distance from a to b, and O represents the office. One can view f as the score of the state, and the inspector wants to find a state with the minimum score.

1. (2 points) How many states are there for n trees?

   There are $n!$ states for n trees.

   2 points for $n!$.

2. (3 points) The inspector can perform hill climbing to approximately minimize the score. To do so, a successor (neighborhood) function needs to be defined for each state. It turns out that one valid way to generate successors for the state $t_1, \ldots, t_n$ is to pick a position $j \in [1, n-1]$ and swap t_j, t_{j+1}. (If you are interested: any permutation can be expressed as a product of adjacent transpositions.) What fraction of the state space does one neighborhood cover? Note for this problem, the neighborhood of a state does not include the state itself.

   There are $(n-1)$ possible successors for each state. So the fraction is $\frac{n-1}{n!}$.

   2 marks for getting to $(n-1)$, 1 mark for the correct answer.

   For the next 4 questions, if the value of n doesn't have to be 112511. Full marks are credited for reasonable explanations of the value chosen by the students.

3. (5 points) Find n from your Madison dataset and compute the number of states in scientific notation, round to one significant digit (i.e. $c \times 10^d$ where c is a single digit). This is why the inspector cannot enumerate all states!

   $$n = 112511, n! \approx 1 \times 10^{519455}$$

   3 marks for finding the correct n, 2 for the correct expression of $n!$.

4. (5 points)One way to estimate the worst case total distance is to assume that to travel between any two trees (or a tree and the office), the inspector has to travel the diameter of the city. Assume Madison has a diameter of 10 km. What is this worst case estimate of the total distance? Use the actual n for Madison and express your answer in LD, namely the average earth moon distance. Keep one significant digit.

$$f(t_1, \ldots, t_n) \leq 10 \text{ km} \times (n+1) = 1125120 \text{ km} \approx 3 \text{ LD}$$

3 marks for the correct expression. 2 for the correct answer in LD. -1 for incorrect significnat digit.

5. (5 points)One way to estimate the best case total distance is to assume that to travel between any two trees (or a tree and the office), the inspector has to travel the minimum distance between any two trees. Assume that distance is 10 m. What is this best case estimate of the total distance? Express your answer in km.

$$f(t_1, \ldots, t_n) \geq 10 \text{ m} \times (n+1) = 1125120 \text{ m} \approx 1125.12 \text{ km}$$

3 marks for the correct expression. 2 for the correct answer in km. 1125 is also correct. If uses $n * d$ instead of $(n+1) * d$, deduct 1 point in total for 3.4 and 3.5

6. (5 points)Suppose the inspector drives at 25 miles per hour (note the Imperial unit). Can the inspector finish the job in one day?

The inspector could drive $25 \times 1.6 \times 24 = 960$ km in one day, less than the best case estimate. Therefore, the inspector cannot finish the job in one day.
3 marks for "no". 2 for explanation.