Jason Jiang

CS540

HW4

Question 1

1.

If h(B) <= 1/2 it will be considered admissible, because the distance between B and G is 1/2 and an admissible heuristic never over-estimates.

2.

a. Enqueue the start node A.

Dequeue A from OPEN and add to CLOSE

Get successors of A: B and C1

Add B and C1 to OPEN

End states:

OPEN: B. C1

h(B) = 100m g(B) = 1/2 f(B)=100+1/2

h(C1) = 0, g(C1) = 1, f(C1) = 1

Back Pointer B -> A and C1->A

b.   Dequeue C1 from OPEN

Add C1 to CLOSE

get successors of C1: C2

add C2 to OPEN

end states:

OPEN: C2 B

CLOSE: A, C1

h(C2) = 0, g(C2)=1+1/2, f(C2) = 1+1/2

Back pointer: C2->C1

c. Dequeue C2 from OPEN

Add C2 to CLOSE

get successors of C2: C3

add C3 to OPEN

end states:

OPEN: C3 B

CLOSE: A, C1, C2

h(C3) = 0, g(C3)=1+3/4, f(C2) = 1+3/4

Back Pointer: C3->C2

d. Dequeue C3 from OPEN

Add C3 to CLOSE

get successors of C3: C4

add C4 to OPEN

end states:

OPEN: C4 B

CLOSE: A, C1, C2, C3

h(C4) = 0, g(C4)=1+7/8, f(C4) = 1+7/8

Back Pointer: C4->C3

e. Dequeue C4 from OPEN

Add C4 to CLOSE

get successors of C4: C5

add C5 to OPEN

end states:

OPEN: C5 B

CLOSE: A, C1, C2, C3, C4

h(C5) = 0, g(C5)=1+15/16, f(C6) = 1+15/16

Back Pointer: C5->C4

3.

limit of f(Ci) when i approaches infinity equals 1+1/2+1/4+1/8......= $(1/2)/(1-1/2)+1=2$

4.

In a heuristic search, the search will only go through the point with the lowest h value in the OPEN list. Because 100 always larger than 1/2, 1/4, 1/8, etc, the search will never search any succcessors of B, where contains goal G, but only go through the branch contains C1, C2, C3.... So it will never reach the goal G.

5.

From the question 3 we've got f(Ci) will be 2 if i approaches infinity. So if we wouldl like to search the successors of B, we need f(B) < 2. So h(B) < 2-g(B), h(B) < 3/2

Also to make it inadmissible, we need to have h(B) < 1/2, so 1/2 < h(B) < 3/2

6.

From 5, we observe we can still get to G though h(B) is not admissible. So it is sufficient but not necessary.

Question 2

a. Current Point: x=2, f(x) = 2

   Tempurature: 2(0.9)^1 = 1.8

   Random number is 3, f(3) = 1;

   p = e^(-|2-1|/1.8) = 0.574

   Random successor is not better than current state x=2.

   p>z, so the x=3 still accepted

b. CP: x = 3, f(x) = 1

   T: 2(0.9)^2=1.62

   y = 1, f(y) = 3

   //p = e^(-2/1.62) = 0.291

   Probability of moving to successor is actually 1, because f(y) > f(x)

c. CP: x = 1, f(x) = 3

   T: 2(0.9)^3=1.458

   y = 1, f(1)=3

   It is not better than current (equal)

p = e^(0) = 1

1 > z, we will not choose it as successor. But actually this is the same state as current state.

d. CP: x = 1, f(x)=3

   T: 2(0.9)^4=1.312

   y=4, f(4) = 0

   p=e(-3/1.312)=0.102

   p < z, so it will not be accepted.

e. CP: x=1, f(x) = 3

   T: 2(0.9)^5=1.181

   y=2, f(y) = 2;

   p = e^(-1/1.181) = 0.429

   p > z, successor will be accepted

f. CP: x = 2, f(x) = 2

   T: 2(0.9)^6 = 1.068

   y=3, f(y) = 1

   p = e^(-1/1.068) = 0.392

   p<z, it will not be accpeted

g. CP: x=2, f(x) = 2

   T: 2(0.9)^7=0.957

   y=4, f(y) = 0

   p = e^(-2/0.957) = 0.124

   p < z it will not be accepted

h. CP: x= 2, f(x) =2

   T: 2(0.9)^8=0.861

   y=3, f(y)=1

   p=e^(-1/0.861) = 0.313

   p<z, it will not be accepted

Question 3

1.

There are n! states for n trees

2.

We will have n-1 states for one neighborhood. According to 1, one neighborhood will cover $(n-1)/n!$

3.

Using lines counter I found there are 112511 trees.

So there are 112511! States

4.

Starting from the office, going to every trees, and backing to office, the inspector has to travel n+1 times of the distance of diameter. 112512*10 = 3 LD

5.

Similiar to 4, inspector will travel 112512 times. 112512*0.01 = 1125.12km

6.

25 mph = 40.23 kmph. Assume the inspector will work 24 hrs a day, 24*40.23 = 965.52 km.

965.52 km < 1125.12 km. So the inspector cannot finish the task even in the best situation. It is impossible to do that in one day.