Assignment 1 - Written component

Graded

Student

Tyler NGUYEN

Total Points

7 / 8 pts

Question 1

Question 1

2 / 2 pts

✔ **– 0 pts** Correct

– 0.2 pts L' H for $\lceil b \rceil$ times not b-1

– 0.2 pts L'H for $\lceil b \rceil$ times not b times

– 1 pt Insufficient proof. Rough idea is correct, but proof is missing some key components

– 1 pt Partially correct. Statements are mostly correct but with some logical/technical errors.

– 2 pts Incorrect

Question 2

Question 2

1 / 1 pt

✔ **– 0 pts** Correct

– 0.6 pts incomplete answer. Rough idea is correct, but proof is missing some key components

– 1 pt incorrect

– 0.5 pts partially correct. Statements are mostly correct but with some logical/technical errors

– 1 pt No answer

Question 3

Question 3

1 / 1 pt

✔ **– 0 pts** Correct

– 1 pt Incorrect

– 0.6 pts Insufficent answer. Rough idea is correct, but proof is missing some key components

– 0.5 pts Partially correct. Statements are mostly correct but with some logical/technical errors

Question 4

Question 4

3 / 4 pts

- **– 0 pts** Correct.
- **– 0.5 pts** Missing runtime of the algorithm.
- **– 1 pt** "Nearest"/"Closest" in heuristic is ambiguous/wrong.
- **– 1 pt** Wrong Heuristic.
- ✔ **– 1 pt** Insufficient arguments in the proof.
- **– 1 pt** Why does showing the max/min of two cables is less than two others prove your heuristic will result in the minimum cable?
- **– 2 pts** No Proof.
- **– 2 pts** Some parts of the answer are unreadable.
- **– 4 pts** No submission.

Questions assigned to the following page: 1, 2, and 3

CPSC 413 Assignment 1
Author: Tyler Nguyen
UCID: 30158563

Question 1:

Want to show: if $\lim_{n\to\infty} \frac{n^b}{a^n}$ = 0 then $n^b \in o(a^n)$

Since the $\lim_{n\to\infty} \frac{n^b}{a^n}$ is of the form of $\frac{\infty}{\infty}$, we will need to use L'Hôpital's Rule such that we find the derivative of the numerator and denominator, then take the limit again with the derived numerator and denominator. So:

$\frac{d}{dn}(n^b) = b * n^{b-1}$

$\frac{d}{dn}(a^n) = a^n * \ln(a)$

From L'Hôpital's Rule we get $\lim_{n\to\infty} \frac{b*n^{b-1}}{a^n*\ln(a)}$ for which the limit will still be in the form of $\frac{\infty}{\infty}$, hence we will need to apply L'Hôpital's Rule b times, we apply it b times since each application of L'Hôpital's Rule will cut down the power of n in the numerator by 1, where after the second application it will be b - 2, the third application it will be b - 3 and so on, therefore after b applications the power of n will be b - b which is 0, hence then we will get b! for the numerator and $a^n * (\ln(a))^b$ for the denominator. So after applying L'Hôpital's Rule b times we will get:

$\lim_{n\to\infty} \frac{b!}{a^n*\ln(a)} = \frac{0}{\infty}$, since the numerator is always going to be a constant and denominator will keep on growing since a > 1 making ln(a) positive. From $\frac{0}{\infty} = 0$ therefore we have proven that for any given constants a > 1 and b > 0, $n^b \in 0(a^n)$ by limit theorem.

Question 2:

Want to show: if $\lim_{n\to\infty} \frac{\log_a(n)}{\log_b(n)} = c$ where c is some constant that is not 0 or ∞ , then $\log_a(n) \in \theta(\log_b(n))$ showing that the base of the logarithm does not affect the asymptotic complexity.

Now note that we can use the change of base formula for logarithms which is $\log_a(n) = \frac{\log_b(n)}{\log_b(a)}$, that was given on the formula sheet under the section "Handouts" on D2L.

From $\lim_{n\to\infty} \frac{\log_a(n)}{\log_b(n)}$ we can use the change of base formula so that we can deal with a problem that shares a common base as well as help simplify the problem. So we can then replace $\log_a(n)$ with $\frac{\log_b(n)}{\log_b(a)}$ which came from the change of base formula. So now we can conclude that:

$\lim_{n\to\infty} \frac{\log_a(n)}{\log_b(n)} = \lim_{n\to\infty} \frac{\frac{\log_b(n)}{\log_b(a)}}{\log_b(n)} = \lim_{n\to\infty} \frac{\log_b(n)}{\log_b(a)\log_b(n)} = \lim_{n\to\infty} \frac{1}{\log_b(a)} = \frac{1}{\log_b(a)}$, since the result of the limit is a positive constant where a,b are some fixed constants that are greater than 1 shows that $\log_a(n) \in \theta(\log_b(n))$ which establishes that the base of logarithms has no affect in the asymptotic complexity.

Question 3:

If f(n) = 2^{2n+3} and g(n) = 2^n then if $\lim_{n\to\infty} \frac{f(n)}{g(n)} =$ c where c is some constant that is not 0 or ∞ shows that f(n) ∈ θ(g(n)). So:

Questions assigned to the following page: 3 and 4

$\lim_{n\to\infty} \frac{2^{2n+3}}{2^n} = \lim_{n\to\infty} 2^{(2n+3)-n} = \lim_{n\to\infty} 2^{2n+3-n} = \lim_{n\to\infty} 2^{n+3} = \infty$, from the simplification where the formula $a^{b-c} = \frac{a^b}{a^c}$ was applied since the two functions share the same base of 2 we can see that the limit results in ∞ which is not a constant c, therefore f(n) does not exist in θ(g(n)).

Question 4:
Algorithm:
Pair each house with the closest internet hub to minimize the length of cable that is needed for each house, so we can initialize a total cable length to 0, then for each house that is located at index i in array A, locate the nearest internet hub in array B and compute the distance between them, then add the distances to m, then continue doing this for all the houses and lastly return the total cable length m.

Proof of correctness:

Greedy heuristic: always choose to link each house to the nearest available internet hub.
Now assume that there exists a more optimal solution than the one that is provided by the greedy algorithm therefore making the greedy algorithm suboptimal, which means there is a solution that can connect the houses to the internet hubs with a shorter total cable length than the result that is given by the algorithm.

I will be proving the correctness of this algorithm through a proof of contradiction.
Let r be the first point where the pairing (A[r], B[r]) in the greedy algorithm's solution differs from the more optimal solution. Now this is where the greedy algorithm's solution will link A[r] to B[r] while the assumed optimal solution will link A[r] to B[x] and another house A[y] to B[r] such that x and y are greater than r.

Now if we swap the links in the assumed optimal solution such that A[r] is now linked with B[r] and that A[y] is linked to B[x], by doing this swap it now aligns the link at position r with the greedy algorithms solution meaning that we just have aligned the pairing of the house at position r with the greedy algorithms choice. This swap does not increase the total length of the cable because |A[r] – B[r]| + |A[y] – B[x]| ≤ |A[r] – B[x]| + |A[y] – B[r]|since the houses and the internet hubs are sorted in increasing value as well as from the greedy heuristic of linking each house to its nearest hub which will minimize the distances that are involved in each link. This heuristic enforces that any deviation from these linkages, followed by a swap cannot result in a shorter path due to the arrangement of the house and internet hubs being linear due to the arrays being sorted from increasing value. However, there are three cases that need to be mentioned.

The first case is where if the house and internet hub are right on each other meaning that they share the same value (A[r] = B[r]). The link here will be optimal since the cable length will be 0.
Second case is if the nearest internet hub is already occupied where if A[r] was originally linked to an internet hub B[x] farther than B[r], suggesting that linking A[r] to its actual nearest hub of B[r] will reduce the total cable length which then contradicting the assumption of a more optimal solution.
Finally, the last case, where a more near internet hub becomes available, such that if A[y] where y > r was connected to B[r], which then means A[r] was linked to a further hub B[x], which then the swap will correct this by ensuring both houses are connected to their nearest internet hubs meaning that this result will not produce a bigger total cable length.

Question assigned to the following page: 4

Now with this new solution after aligning the first point of difference r with the greedy algorithms solution, we can see that it will have the same or a lower total cable length than the assumed optimal solution, which contradicts the assumption that the greedy algorithms solution was not optimal, which then proves the correctness of this algorithm.

Runtime:
Since it is stated in the question that both arrays A and B are sorted by increasing value, the runtime of the algorithm is O(n) where n will be the number of houses and hubs, because the algorithm is iterating over the houses once and for each house it is finding the nearest hub where the search for it is in constant time because the arrays are sorted.