Computer Information Systems 23 - Homework 9

Cody Vig

Problem 1

(20 points) Consider the list: `5, 12, 25, 32, 38, 46, 58, 62, 85, 90, 97, 105, 110`.

1. How many comparisons are made with **sequential search** to determine if the number 120 is in the list?
2. How many comparisons are made with **binary search** to determine if the number 120 is in the list?
3. Repeat 1 and 2 for the number 25.

Solution

Problem 2

(20 points) Each of the following expressions represents the number of operations for certain algorithms. What is the order of each of these expressions? That is, put the following expressions in terms of Big-O notation.

1. $n^3 - 8n$
2. $5n^4 + 2n^2 + 8$
3. $(n^2 + n - 1)(8n + 5)$
4. $8n(6n + 1)$

Solution

Part a) Note $n^3 - 8n = n^3 \left(1 - \frac{8}{n^2}\right) \sim n^3$ as $n \to \infty$, so I claim $n^3 - 8n \in \mathcal{O}(n^3)$.

Proof. Define $f(n) := n^3 - 8n$ and $g(n) := n^3$ as integer-valued functions. Let $\varepsilon > 0$ and $N > 0$ satisfy $N^2 \geq \frac{8}{\varepsilon}$. Then for $n \geq N$ we have

$$\left| \frac{f(x)}{g(x)} - 1 \right| = \frac{8}{n^2} \leq \frac{8}{N^2} \leq \varepsilon.$$

Hence $\lim_{n \to \infty} \frac{f(x)}{g(x)} = 1$ and so $f \in \mathcal{O}(g)$.

Part b) Note $5n^4 + 2n^2 + 8 = n^4 \left(5 + \frac{2}{n^2} + \frac{8}{n^4}\right) \sim 5n^4$ as $n \to \infty$, so I claim $5n^4 + 2n^2 + 8 \in \mathcal{O}(n^4)$.

Proof. Define $f(n) := 5n^4 + 2n^2 + 8$ and $g(n) := n^4$ as integer-valued functions. Let $\varepsilon > 0$ and $N > 0$ satisfy $N^2 \geq \frac{8}{\varepsilon}$. Then for $n \geq N$ we have

$$\left| \frac{f(x)}{g(x)} - 5 \right| \leq \frac{2}{n^2} + \frac{8}{n^4} \leq \frac{2}{n^2} + \frac{8}{n^2} = \frac{10}{n^2} \leq \frac{8}{N^2} \leq \varepsilon.$$

Hence $\lim_{n \to \infty} \frac{f(x)}{g(x)} = 1$ and so $f \in \mathcal{O}(g)$.

Problem 3

(10 points) Suppose that L is a sorted list of 1024 elements. Using the binary search algorithm, what is the maximum number of comparisons made by binary search to determine if an item is in L? Explain how wthis is related to the time complexity of the binary search algorithm.

Solution

Problem 5

(10 points) With a sorted list of 1024 elements, what is the maximum number of recursive calls made to the binary search algorthm to find an element? Explain how this is related to the time complexity of the binary search algorithm.

Solution