IITM-CS2800-2025 : Design and Analysis of Algorithms

Practice Problem Sheet #1 Released On : Jan 30, 1pm

PART A (to be done during the tutorial hours)

1. Recall the definition of the *iterated logarithm* function: $\log^* n$ is the number of times that you have to take the log of n for it to be at most 1.

   Give the tightest asymptotic relation between the following two functions.

$$f(n) = \log(\log^* n), \text{ and}$$
$$g(n) = \log^*(\log n).$$

2. For the following functions $f(n)$ and $g(n)$, answer what is the tightest asymptotic relation between $f(n)$ and $g(n)$.
   (a) $f(n) = n^{\log_2 c}$ and $g(n) = c^{\log_2 n}$ where $c > 2$ is a constant.
   (b) $f(n) = 2^{n^2}$ and $g(n) = n^2 2^{2n}$.
   (c) $f(n) = (\log n)!$ and $g(n) = n^2$.
   (d) $f(n) = n^{\frac{\log\log n}{\log n}}$ and $g(n) = n^{1/10}$.

3. Solve the following recurrences.
   (a) $T(n) = T(\sqrt{n}) + \log n$ and $T(2) = 100$.
   (b) $T(n) = 4\,T(\frac{n}{2}) + n$ and $T(1) = 1$.
   (c) $n\,T(n) = (n-2)\,T(n-1) + 2$ and $T(1) = 1$.
   (d) $T(n) = 4\,T(\frac{n}{3}) + n$ and $T(1) = 1$.

4. You have devised a new algorithm that you title *ternary search*. The algorithm is as follows:

```
TernarySearch(A, k)
Input: Sorted array A[1, 2, ..., n] and a key k
Output: Index of k, if present, else −1
if A is empty then return −1
if A[n/3] = k then
    return n/3
else
    if A[2n/3] = k then
        return 2n/3
    else
        if k < A[n/3] then
            Ternary Search(A[1, 2, ..., n/3 − 1])
        else
            if A[n/3] < k < A[2n/3] then
                Ternary Search(A[n/3 + 1, ..., 2n/3 − 1])
            else
                Ternary Search(A[2n/3 + 1, ..., n])
```

Is your algorithm better than binary search? Why/Why not. Support your reasons mathematically.

5. Smullyan[1] Island has three types of inhabitants: *knights*, who always speak the truth, *knaves* who always lies, and *normals* who speak truth sometimes, and lie sometimes. Everyone knows everyone else's names and type. Your goal is to know everyone's type. You can ask any inhabitant, of another inhabitant's type (this is the only question you are allowed to ask), but you cannot ask his/her own type. Asking the same question multiple times to a person will yield the same answer.

   (a) Suppose that a strict majority of inhabitants are knights, give an efficient algorithm to find the type of every inhabitant.

   (b) Prove that if the number of knights is at most half the total number of inhabitants, then it is impossible to find the type of every inhabitant correctly.

[1] Raymond Smullyan was a mathematician and philosopher famous for his many logic puzzles using self-reference involving knights and knaves. Check out What is the name of this book?

Part B (can be done outside the tutorial hours for practice)

1. Is $2^{O(n)} = O(2^n)$? If yes, give a proof. Otherwise, give a counter-example.

2. Does there exist an $\epsilon > 0$ such that $\log n = \Omega(n^\epsilon)$?

3. Consider the following statement: "In every instance of stable matching between courses and students, if the Gale-Shapley algorithm is run with the students applying to courses, at least one of the students will be alloted a course that is his/her first preference".

   If this statement is true, prove it. Otherwise, provide a counter-example.

4. Recall the definition of Big-Oh: We say that $f(n) = O(g(n))$ if $\exists c > 0, n_0 > 0$ such that for every $n \geq n_0$, $f(n) \leq cg(n)$. Show that the definition given above is equivalent to the following definition.

   $f(n) = O(g(n))$ if $\exists c > 0$ such that for every $n \geq 1$, $f(n) \leq cg(n)$.

5. Suppose we want to talk about the asymptotic relation between the two functions $T(n)$ and $f(n)$ (where $n \in \mathbb{N}$). Consider the limit:

   $$\ell = \lim_{n \to \infty} \frac{T(n)}{f(n)}$$

   If ℓ is a finite positive number, then $T(n) \in \Theta(f(n))$ (extra : prove this !).

   (a) Use this definition to compare $T(n) = 14n^3 + 5n^2 + 6n + 4$ and $f(n) = n^3$.

   (b) Attempt on applying this when $T(n) = n(2 + \sin n)$ and $f(n) = n$. Notice that the limit does not exist and hence the above method cannot be used to conclude and asymptotic relationship in this case. Go back to original definitions and argue that $T(n) \in \Theta(f(n))$ in this case.