CENG 115 Fall 2023 Homework 1

Due date: 23th of November, 9:45 AM

Note: You are expected to bring <u>hardcopies</u> of your answers to the class (your own section) on 23rd Nov (Recitation hour). Homeworks submitted afterwards will not be evaluated.

1. **(25 points)**
   A detective has interviewed three witnesses to a crime. These are the cook, the gardener and the handyman. From the stories of the witnesses the detective has concluded that
   - the cook and the gardener cannot both be telling the truth;
   - the gardener and the handyman are not both lying (i.e. if one of them is lying the other is not)
   - if the handyman is telling the truth then the cook is lying.
   Can the detective determine whether the cook is telling the truth or lying? Explain your reasoning.

   <u>Solution:</u>

Premise 2: the cook and the gardener cannot both be telling the truth	$\neg c \vee \neg g$
Premise 3: the gardener and the handyman are not both lying	$g \vee h$
Premise 4: if the handyman is telling the truth then the cook is lying	$h \rightarrow \neg c$

   <u>One way to solve:</u>
   $\neg c \vee \neg g$ (Premise 2)
   $g \vee h$ (Premise 3)
   $\neg h \vee \neg c$ (5) (obtained from Premise 4)
   Here, if the cook is telling the truth, then $\neg c$ is False.
   For (2) to be True, $\neg g$ becomes T.
   If $\neg g$ is True, then g is False. For (3) to be True, h becomes True.
   If h is True, $\neg h$ is False. For (5) to be True, $\neg c$ becomes True.
   However at the beginning $\neg c$ was assumed to be False. So there is a contradiction.

   If we start by $\neg c$ is True, then there is no contradiction, this shows $\neg c$ is True, **cook is lying**.

   <u>Another way to solve:</u>
   $\neg c \vee \neg g$ (Premise 2)
   $g \vee h$ (Premise 3)
   $\underline{\neg h \vee \neg c}$ (5) (obtained from Premise 4)
   Therefore, $\neg c \vee \neg c$ equivalent to $\neg c$, **cook is lying**.

2. **(25 points)** Determine whether 'the set of positive integers that are divisible by 5 but not by 6' is countable or uncountable. If this set is countable, prove it by proposing a bijection (one-to-one correspondence) from the set of positive integers to this set.

   Hint: In addition to the arithmetic operators, you are allowed to use *floor* and *ceiling* functions.

   <u>Answer:</u>
   Consider $f : \mathbf{Z}^+ \rightarrow \{x \mid x$ is an integer that is divisible by 5 but not by 6$\}$ where
   $f(n) = 5n + 5\left\lfloor \frac{n-1}{5} \right\rfloor$

$f(n)$ generates positive integers that are not divisible by 6 from $\mathbf{Z}^+$ as can be seen: (1,2,3,4,5,6,7, 8,..) $\rightarrow$ (5,10,15,20,25,35,40,45..).

This function is one-to-one because every element in the codomain has a one image in the domain. This function is also onto because it completely covers the set of positive integers that are that is divisible by 5 but not by 6.

3. (20 points)

a) Prove that $f(x) = 2x^2+5x \log_2 x$ is $O(x^2)$ by finding a pair of (c,k) for the inequality in the big-O definition.

Answer:

$g(x) = x^2 \quad f(x) = 2x^2+5x \log_2 x$

Let c=4 and k=1, is $f(x) \leq 4g(x)$ for $x \geq 5$? $f(1) \leq 4g(1)$? $2 \leq 4$? **Yes.**

Let c=4 and k=2, $f(2) \leq 4g(2)$? $8+10 \log_2 2 \leq 16$? **No.**

Let c=4 and k=4, $f(4) \leq 4g(4)$? $32+40 \leq 64$? **No.**

Let c=4 and k=8, $f(8) \leq 4g(8)$? $128+120 \leq 256$? **Yes.**

Therefore, (c,k)=(4,8) is a proper pair to show that $f(x)$ is $O(x^2)$.

Or, let c=4, $2x^2+5x \log_2 x \leq 4x^2$?

$5x \log_2 x \leq 2x^2$?

$\log_2 x \leq 0.4x$? **true when $8 \leq x$** □

b) $\sum_{k=1}^{n} (3k^2 - 1)$ is $\Theta(?)$.

Answer:

$$3\sum_{k=1}^{n} k^2 - \sum_{k=1}^{n} 1 = 3 \cdot \frac{n \cdot (n+1) \cdot (2n+1)}{\mathbf{6}} - n = n^3 + \ldots = \boldsymbol{\Theta}(n^3)$$

4. (30 points)

a) Suppose you are given a list of positive integers { $a_1, a_2, \ldots a_n$}. Describe an algorithm (write-down its pseudocode) that goes through the elements in the list one by one and finds the index of the first element that is greater than the sum of all previous elements in the list. If there is no such element, index should be returned as zero.
Attention: Your pseudo-code should not include functions/structures that are specific to C, C++, Java or Python

b) Define the worst-case time complexity of the algorithm you described in (a) with its exact order of growth, i.e. Θ. You can assume that all the lines in your code take a fixed amount of time, t. Explain your answer shortly.

Answer:

a)

```
procedure firstterm (a1,a2, … an: integers)
    location := 0
    sum := a1
    i := 2
while (i ≤ n and location = 0)
begin
     if ai>sum then location := i
     else
          begin
          sum := sum + ai
          i := i+1
     end
end
```

{location is the index of the term we are looking for, if location=0 then the term is not found}

b) For this pseudocode, complexity is $\Theta(n)$ for the **worst case** because the loop is **executed *n* times**. If pseudocode describes another algorithm complexity should be in accordance with that algorithm. For example, if the sum is calculated as another loop in the main loop, then $\Theta(n^2)$.