CSC236 fall 2014, Assignment 2

Due November 3rd, 10 p.m.

The aim of this assignment is to work with recurrences, to notice the effect of changing the strength of a predicate being proven by Induction, and to get an early start on thinking about deterministic finite state automata (DFSAs) in a familiar concrete setting (as Python programs).

You may work in groups of no more than three students, and you should produce a single solution in a PDF file named `a2.pdf`, submitted to MarkUs.
You will receive 20% of the marks for any question (or part of a question) that you either leave blank or for which you write "I cannot answer this."

1. Define function f by

$$f(n) = \begin{cases} 10, & \text{if } n = 0 \\ 3f(\lfloor 2n/5 \rfloor) + 6n^4 & \text{if } n \geq 1 \end{cases}$$

   Prove there is a real number c, such that $f(n) \leq cn^4$ for most natural numbers n.
   DO NOT USE THE MASTER THEOREM.

2. Consider this recurrence T:

$$T(n) = \begin{cases} 3, & \text{if } n = 1 \\ 2 + T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) & \text{if } n \geq 2 \end{cases}$$

   For each natural number n, let $P(n)$ be: $T(n) \leq T(n+1)$.

   You will prove, by Complete Induction, that P is true for all positive natural numbers.
   DO NOT PROVE the (seemingly) more general result $1 \leq m \leq n \rightarrow T(m) \leq T(n)$, NOR ANY OTHER result by Induction.

   To practice for the proof, parts (a) and (b) ask you to show the Complete Inductive Step for two explicit numbers.

   (a) Write out $P(236)$ (simply substitute 236 for n).
   Determine an earlier value, or values, of k so that assuming $P(k)$ for those values allows you to prove $P(236)$.
   Write out that/those $P(k)$ explicitly, and use that to prove $P(236)$.

   (b) Repeat the previous part, but for $P(237)$.

   (c) Prove, by Complete Induction, that P is true for all positive natural numbers.

3. Read dfsa.py, and try running it. Once you understand it, create new DFSA instances:

   times_three_a: Accepts lists of Symbols such that: the number of **a**s in the list is a multiple of 3.
   Rejects all other lists.

   first_last_a: Accepts lists of Symbols such that: the first and last symbol is an **a**.
   Rejects all other lists.