Assignment 1

Grades : each assignment is 5% in the final score and there is a 1% bonus in each assignment for bonus problems. We use the following scale for the simplicity of grading: the total score is 50 (additional 10 for bonus problems) for each assignment.

Please use the basic definition of \( \Theta, O, \Omega \)-notations to prove things. In particular, please show your choice of constants \( c_1, c_2, n_0 \).

Problem 1 [Full Score: 15].

(1)[5 ] Prove for any real constants \( a \) and \( b \), where \( b > 0 \),
\[
(n + a)^b = \Theta(n^b).
\]

(2)[5 ] Is \( 2^{n+1} = O(2^n) \)? Is \( 2^{2n} = O(2^n) \)? Please provide a brief explanation.

(3)[5 ] Rank the following functions by order of growth: \( n, n!, \ln(n), 2^{\ln^2(n)}, 2^{2n}, (\ln(n))^{\ln^2(n)} \). No explanation is necessary.

Problem 2 [Full Score: 10]. Prove Theorem 3.1 (on page 48). Namely, for any two functions \( f(n) \) and \( g(n) \), we have \( f(n) = \Theta(g(n)) \) if and only if \( f(n) = O(g(n)) \) and \( f(n) = \Omega(g(n)) \).

Problem 3 [Full Score: 10]. Let \( f(n) \) and \( g(n) \) be asymptotically nonnegative functions. Using the basic definition of \( \Theta \)-notation, prove that
\[
\max(f(n), g(n)) = \Theta(f(n) + g(n)),
\]
and also
\[
f(n) + g(n) = \Theta(\max(f(n), g(n))).
\]

Problem 4 [Full Score: 15]. As all for loop algorithms, the following algorithm can be written as a while loop:

Algorithm: arrayMax(A,n):

Input : An array \( A \) storing \( n \geq 1 \) integers.

Output : The maximum elements in \( A \).

\[
\text{currentMax} \leftarrow A[0]
\]
for \( i \leftarrow 1 \) to \( n - 1 \) do
if currentMax < A[i] then
  currentMax ← A[i]
end if
end for
return currentMax

Write such a loop \( L \): while \( C \) do \( B \). Then design a loop invariant \( I \) to prove the correctness of the computation which converts the before-loop state \( \langle P \rangle \) into the after-loop state \( \langle Q \rangle \): \( P \) is \( (i = 0) \land (currentMax = A[0]) \) and \( Q \) is ”currentMax is the maximum value stored in \( A \)”. Prove the correctness of the while loop by proving the following three conditions: (follow the examples in the note)

- \( P \rightarrow I \).
- \( \langle I \land C \rangle \ B \langle I \rangle \).
- \( \langle I \land \neg C \rangle \rightarrow Q \).

**Problem 5 [Bonus Problems: 10].** Solve the following problems from the textbooks about \( O(\cdot) \) and \( \Omega(\cdot) \) notations.

1. Give an example of a positive function \( f(n) \) such that \( f(n) \) is neither \( O(n) \) nor \( \Omega(n) \).
2. Show that \( \sum_{i=1}^{n} i/2^i < 2 \).