**CS 58000\_01/02I Design, Analysis and Implementation Algorithms (3 cr.)**

**Assignment As\_03**

**Student Name: ­­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

This assignment As 03 is due at 11:59 p.m., Wednesday, October 25, 2023. Please submit your assignment solution to Brightspace (purdue.brightspace.com). No late turn-in is accepted. Please write your name on the first page of your assignment. Your file name should be your last name such as NgP\_As03.docx. Please number your problem-answer clearly such as Problem I.a, I.b, I.c.i, I.c.ii, II, III.(a), III.(b), III.(c), III.(d), III.(e), III.(f). The problems’ answers must be arranged according to the order of the given problem. Please answer your questions using only a Word file (.docx file only). No pdf file will be accepted. Without using a Word file (.docx file) the submitted problems’ answers would not be graded.

The total number of points for this Assignment\_03 is 100 points.

1. [30 pts] Using definitions of Θ, O, and Ω to establish your claims:

a: Prove and disprove that for any real constants a and b, where b > 0,

(n + log2a)b = Θ(nb ).

b: Explain why the statement “The running time of algorithm A, T(A) 𝑖𝑠

𝑎𝑡 𝑙𝑒𝑎𝑠𝑡

O(n2 )” is meaningless.

c: Prove or disprove:

1. = O( )
2. = O( )

Note that:

II.[10 pts]

Is growing faster than the exponential function ?

Use to show your claim.

What is the order of growth if the closed form formula is ?

III.[60 points] Given the following recurrence relation

T(n) = T(n -1) + f(n),

find a bound for each of them.

(a) if f(n) = c, where c is a constant.

(b) if f(n) = log2 n.

(c) if f(n) = n.

(d) if f(n) = na .

(e) if f(n) = an.

(f) Solve T(n) = 2T(n-1) + c, where c is a constant