COMP PhD Qualifier Sample Exam

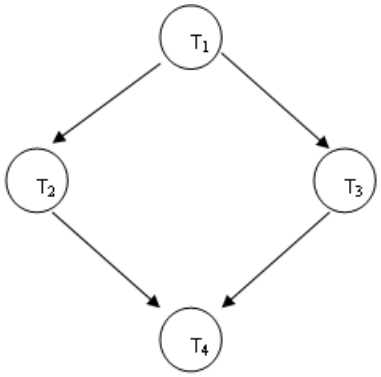
- Time: 09:30 - 12:30
- By submitting this exam, you agree to fully comply with Koç University Student Code of Conduct, and accept any punishment in case of failure to comply. If you do not submit this exam on time, you will directly fail the qualifier.
- In general, you may neither use other resources during the exam, nor ask any questions.
- Good luck!

1. [**agursoy 17 pts**] Two questions:
   (a) Let T be a binary tree with n nodes. Define the lowest common ancestor (LCA) between two nodes v and w as the lowest node in T that has both v and w as descendents (where we allow a node to be a descendent of itself). Given two nodes v and w, descrive a linear algorithm for finding LCA of v and w.
   (b) Describe, in pseudo-code, a link-hopping method for finding the middle node of a doubly linked list with header and trailer sentinels, and an odd number of real nodes between them. The method must use only hopping; it cannot use a counter. What is running time of your method?
2. [**dyuret 17 pts**] In a research study involving nine athletes, their heart rate was recorded during the competition. The sample mean and sample standard deviation were 188.0 and 7.2, respectively. Assuming that the heart rate distribution is approximately normal, construct a 98% confidence interval for the true mean heart rate of the athletes. (The CDF table for the standard normal distribution is given on the last page.)
3. [**yyemez 17 pts**] Suppose f(x), g(x) and h(x) are functions defined on the set of real numbers. By using the definition of big-oh notation, show that if f(x) is O(g(x)) and g(x) is O(h(x)), then f(x) is O(h(x)).
4. [**oozkasap 17 pts**] You are given the following graph that represents the relationship among four concurrently executing processes: T1, T2, T3, and T4. Each process is responsible for computing a sub-part of a solution. On the graph, a directed edge from Tx to Ty means that process Tx must finish its computation before Ty starts. It is also given that, processes T2 and T3 access a shared data item D during their computation.

Assume that the processes start execution concurrently. You are asked to describe how these processes can use semaphores efficiently to enforce the ordering and synchronization properties explained above.

(a) Write the pseudo-codes of these processes.

(b) Describe the task of the semaphores you use and their initial values.



5. [**akupcu 17 pts**] True/False. Answers must be explained.

   (a) For every two positive functions f and g, if g(n) is O(n), then f(g(n)) is O(f(n)).

   (b) A heap with n elements can be converted to a binary search tree in O(n) time.

   (c) No adversary can force randomized quicksort to run in quadratic time ($O(n^2)$).

6. [**stasiran 17 pts**] Suppose you have three types: TwoDimensionalShape, Rectangle, and Circle. You would like to write a method printAreasAndCircumferences() that takes an array containing references to Rectangle and Circle objects as input. The method prints their areas and circumferences one by one. How would you organize your types into a type hierarchy? Show code skeletons for all three types, using abstract classes and methods as necessary. How does your type hierarchy make the it possible for you to implement printAreasAndCircumferences()?

7. [**eyilmaz 17 pts**] Consider the relations

```
Branch (BranchName, Assets, City)
Customer(CustomerName, Address, City)
Account(AccountNumber, BranchName, CustomerName, Balance)
```

   The following query finds the assets and names of all banks that have depositors living in Blacksburg and have a balance of more than $100000 :

$$\pi_{\text{Asset, BranchName}}\left(\sigma_{\text{Customer.City='Blacksburg' AND Balance>100000}}\left(\text{Customer} \bowtie \text{Account} \bowtie \text{Branch}\right)\right)$$

   Use the equivalence rules from the previous question (or any others you can come up with) to rewrite this query into something that can be executed much faster.

Your ultimate goal is to ensure that each relation used in natural join contains the smallest possible number of rows and attributes. Convert the query in steps; in each step, apply one equivalence rule. Mention the rule, and write the new relational algebra expression that results from applying the rule.

8. [**msezgin 17 pts**]

   (a) Identify all variable declarations (circle, and label).

```
let x = 10

    in let f = proc(x) -(x,-(0,x))

       in let y = 1

          in let g = proc(x) begin set x=5; x; y end

             in -( (f begin set x=-(x,-(0,x)); y; x end), (g (g y)) )
```

   (b) Identify all variable references (circle, and label).

```
let x = 10

    in let f = proc(x) -(x,-(0,x))

       in let y = 1

          in let g = proc(x) begin set x=5; x; y end

             in -( (f begin set x=-(x,-(0,x)); y; x end), (g (g y)) )
```

   (c) Write down the lexical depth of each variable reference.

```
let x = 10

    in let f = proc(x) -(x,-(0,x))

       in let y = 1

          in let g = proc(x) begin set x=5; x; y end

             in -( (f begin set x=-(x,-(0,x)); y; x end), (g (g y)) )
```