CS 173, Spring 18 Examlet 11, Part B

Name: _____

NetID: _____ **Lecture: A B**

Discussion: Thursday Friday 9 10 11 12 1 2 3 4 5 6

(15 points) Check the (single) box that best characterizes each item.

$T(1) = d$
$T(n) = T(n-1) + n$

$\Theta(n)$	☐	$\Theta(n \log n)$	☐	$\Theta(n^2)$	√	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

$T(1) = d$
$T(n) = 2T(n/4) + n$

$\Theta(\log n)$	☐	$\Theta(\sqrt{n})$	☐	$\Theta(n)$	√	$\Theta(n \log n)$	☐
$\Theta(n^2)$	☐	$\Theta(n^3)$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

The running time of mergesort is $O(n^3)$. true √ false ☐

For a problem to satisfy the definition of NP, a "no" answer must have a succinct justification. true ☐ false √

The Travelling Salesman Problem polynomial ☐ exponential ☐ in NP √

CS 173, Spring 18 Examlet 11, Part B

Name:_____

NetID:_____ **Lecture: A B**

Discussion: Thursday Friday 9 10 11 12 1 2 3 4 5 6

(15 points) Check the (single) box that best characterizes each item.

$T(1) = d$
$T(n) = 2T(n/2) + n$

$\Theta(n)$	☐	$\Theta(n \log n)$	☑	$\Theta(n^2)$	☐	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

$T(1) = d$
$T(n) = 3T(n-1) + c$

$\Theta(n)$	☐	$\Theta(n \log n)$	☐	$\Theta(n^2)$	☐	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☑

Algorithm A takes n^5 time. On one input, A takes x time. How long will it take if I double the input size?

$2x$ ☐ $5x$ ☐ $32x$ ☑ x^5 ☐

The Marker Making problem can be solved in polynomial time.

true ☐ false ☐ not known ☑

Problems in class P (as in P vs. NP) require exponential time

true ☐ false ☑ not known ☐

CS 173, Spring 18 Examlet 11, Part B

Name:_____

NetID:_____ **Lecture: A B**

Discussion: Thursday Friday 9 10 11 12 1 2 3 4 5 6

(15 points) Check the (single) box that best characterizes each item.

$T(1) = d$
$T(n) = T(n/2) + d$

$\Theta(\log n)$	☑	$\Theta(\sqrt{n})$	☐	$\Theta(n)$	☐	$\Theta(n \log n)$	☐
$\Theta(n^2)$	☐	$\Theta(n^3)$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

$T(1) = d$
$T(n) = 3T(n/3) + n$

$\Theta(n)$	☐	$\Theta(n \log n)$	☑	$\Theta(n^2)$	☐	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

All ways to assign True/False values to n input variables

$\Theta(\log n)$	☐	$\Theta(n)$	☐	$\Theta(n \log n)$	☐	$\Theta(n^2)$	☐
$\Theta(n^3)$	☐	$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☐	$\Theta(2^n)$	☑

Producing all parses for a sentence.

polynomial ☐ exponential ☑ in NP ☐

The chromatic number of a graph with n nodes can be found in polynomial time.

true ☐ false ☐ not known ☑

Name:_____

NetID:_____ **Lecture: A B**

Discussion: Thursday Friday 9 10 11 12 1 2 3 4 5 6

(15 points) Check the (single) box that best characterizes each item.

$T(1) = d$
$T(n) = 3T(n/2) + n$

$\Theta(n)$	☐	$\Theta(n \log n)$	☐	$\Theta(n^2)$	☐	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☑	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

$T(1) = d$
$T(n) = T(n-1) + c$

$\Theta(\log n)$	☐	$\Theta(\sqrt{n})$	☐	$\Theta(n)$	☑	$\Theta(n \log n)$	☐
$\Theta(n^2)$	☐	$\Theta(n^3)$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

The running time of the Towers of Hanoi solver is $\Theta(n!)$ true ☐ false ☑

Circuit satisfiability can be solved in exponential time. true ☑ false ☐ not known ☐

Producing all parses for a sentence requires exponential time. true ☑ false ☐ not known ☐

Name:_____

NetID:_____ **Lecture: A B**

Discussion: Thursday Friday 9 10 11 12 1 2 3 4 5 6

(15 points) Check the (single) box that best characterizes each item.

$T(1) = d$
$T(n) = T(n/2) + n$

$\Theta(\log n)$	☐	$\Theta(\sqrt{n})$	☐	$\Theta(n)$	☑	$\Theta(n \log n)$	☐
$\Theta(n^2)$	☐	$\Theta(n^3)$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

$T(1) = d$
$T(n) = 2T(n/3) + d$

$\Theta(n)$	☐	$\Theta(n \log n)$	☐	$\Theta(n^2)$	☐	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☑	$\Theta(n^{\log_2 3})$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

Algorithm A takes 2^n time. On one input, A takes x time. How long will it take if I add one to the input size?

$x + 2$	☐	$2x$	☑	2^x	☐	x^2	☐

Problems in class P (as in P vs. NP) can be solved in exponential time

true	☑	false	☐	not known	☐

Deciding if a graph is 2-colorable

polynomial	☑	exponential	☐	in NP	☐

Name:_____

NetID:_____ **Lecture: A B**

Discussion: Thursday Friday 9 10 11 12 1 2 3 4 5 6

(6 points) Your partner has implemented the function Merge(A,B), which merges two sorted linked lists of integers. Using Merge, fill in the missing parts of this implementation of Mergesort.

Mergesort($L = (a_1, a_2, \ldots, a_n)$) \\ input is a linked list L containing n integers

Solution: if (n=1) return L

p = floor(n/2)

Solution:

$L_a = (a_1, \ldots, a_p)$

$L_b = (a_{p+1}, \ldots, a_n)$

return Merge(Mergesort(L_a), Mergesort(L_b))

(9 points) Check the (single) box that best characterizes each item.

$T(1) = d$
$T(n) = 3T(n/2) + d$

$\Theta(n)$	☐	$\Theta(n \log n)$	☐	$\Theta(n^2)$	☐	$\Theta(n^3)$	☐
$\Theta(n^{\log_3 2})$	☐	$\Theta(n^{\log_2 3})$	☑	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

Merging two sorted lists

$\Theta(1)$	☐	$\Theta(\log n)$	☐	$\Theta(n)$	☑	$\Theta(n \log n)$	☐
$\Theta(n^2)$	☐	$\Theta(n^3)$	☐	$\Theta(2^n)$	☐	$\Theta(3^n)$	☐

Circuit satisfiability can be solved in polynomial time.

true ☐ false ☐ not known ☑