CS 173, Spring 2016
Examlet 11, Part B

NETID:

FIRST:

LAST:

Discussion: Monday 9 10 11 12 1 2 3 4 5

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

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

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

Producing all parses for a sentence.

polynomial ☐ exponential ☐ in NP ☐

The running time of mergesort is recursively defined by $T(1) = d$ and $T(n) =$

$2T(n-1) + c$ ☐ $2T(n-1) + cn$ ☐
$2T(n/2) + c$ ☐ $2T(n/2) + cn$ ☐

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

$\Theta(n)$ ☐ $\Theta(n^2)$ ☐ $\Theta(n \log n)$ ☐ $\Theta(2^n)$ ☐

The running time of Karatsuba's algorithm is recursively defined by $T(1) = d$ and $T(n) =$

$2T(n/2) + cn$ ☐ $3T(n/2) + cn$ ☐
$4T(n/2) + cn$ ☐ $4T(n/2) + c$ ☐

CS 173, Spring 2016
Examlet 11, Part B

NETID:

FIRST:

LAST:

Discussion: Monday 9 10 11 12 1 2 3 4 5

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

The running time of mergesort

$\Theta(n)$ ☐ $\Theta(n \log n)$ ☐

$\Theta(n^2)$ ☐ $\Theta(2^n)$ ☐

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

$\Theta(\log n)$ ☐ $\Theta(n)$ ☐ $\Theta(n \log n)$ ☐ $\Theta(n^2)$ ☐

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

true ☐ false ☐

Algorithm A takes $\log_2 n$ time. On one input, A takes x time. How long will it take if I double the input size?

$x + 1$ ☐ $2x$ ☐ 2^x ☐ x^2 ☐

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

true ☐ false ☐ not known ☐

CS 173, Spring 2016
Examlet 11, Part B

NETID:

FIRST:

LAST:

Discussion: Monday 9 10 11 12 1 2 3 4 5

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

The running time of the Towers of Hanoi solver is recursively defined by $T(1) = d$ and $T(n) =$

$2T(n-1)+c$ ☐ $2T(n-1)+cn$ ☐

$2T(n/2)+c$ ☐ $2T(n/2)+cn$ ☐

For a problem to satisfy the definition of co-NP, a "yes" answer must have a succinct justification. true ☐ false ☐

The running time of the Towers of Hanoi solver

$\Theta(\log n)$ ☐ $\Theta(n \log n)$ ☐

$\Theta(n^2)$ ☐ $\Theta(2^n)$ ☐

$n^{log_2 3}$ grows faster than n ☐ slower than n ☐ at the same rate as n ☐

The Marker Making problem can be solved in polynomial time. true ☐ false ☐ not known ☐

CS 173, Spring 2016
Examlet 11, Part B

NETID:

FIRST: | **LAST:**

Discussion: Monday 9 10 11 12 1 2 3 4 5

(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

p = floor(n/2)

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

$T(1) = d$
$T(n) = 2T(n-1) + c$ $\Theta(n)$ ☐ $\Theta(n^2)$ ☐ $\Theta(n \log n)$ ☐ $\Theta(2^n)$ ☐

It takes exponential time to determine whether a propositional logic expression can be made true by picking the right true/false values for its propositional variables (e.g. p, q, r). true ☐ false ☐ not known ☐

Merging two sorted lists $\Theta(\log n)$ ☐ $\Theta(n)$ ☐ $\Theta(n \log n)$ ☐ $\Theta(n^2)$ ☐

CS 173, Spring 2016
Examlet 11, Part B

NETID:

FIRST: | **LAST:**

Discussion: Monday 9 10 11 12 1 2 3 4 5

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

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

Karatsuba's integer multiplication algorithm
$\Theta(n^2)$ ☐ $\Theta(n^3)$ ☐ $\Theta(n \log n)$ ☐
$\Theta(n^{log_2 3})$ ☐ $\Theta(n^{log_3 2})$ ☐ $\Theta(2^n)$ ☐

Marker Making polynomial ☐ exponential ☐ in NP ☐

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

$n^{log_2 4}$ grows faster than n^2 ☐ slower than n^2 ☐ at the same rate as n^2 ☐

CS 173, Spring 2016
Examlet 11, Part B

NETID:

FIRST: | **LAST:**

Discussion: Monday 9 10 11 12 1 2 3 4 5

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

The Travelling Salesman

polynomial ☐ exponential ☐ in NP ☐

Merging two sorted lists of numbers

$\Theta(\log n)$ ☐ $\Theta(n)$ ☐ $\Theta(n \log n)$ ☐ $\Theta(n^2)$ ☐

The running time of the Towers of Hanoi solver

$\Theta(\log n)$ ☐ $\Theta(n \log n)$ ☐

$\Theta(n^2)$ ☐ $\Theta(2^n)$ ☐

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

$\Theta(\sqrt{n})$ ☐ $\Theta(n)$ ☐ $\Theta(n \log n)$ ☐

$\Theta(n^2)$ ☐ $\Theta(2^n)$ ☐

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

$x + 2$ ☐ $2x$ ☐ 2^x ☐ x^2 ☐