Question 1 (1 point)

The Fibonacci numbers are defined by $F(0) = 0, F(1) = 1,$ and $F(n) = F(n-1) + F(n-2)$, for $n \geq 2$. Consider the following recursive algorithm that takes an integer $n \geq 0$ as input:

Algorithm $fib(n)$:

If $n \leq 1$: return $n$

Else: return $fib(n-1) + fib(n-2)$

End of Algorithm

Let $P(n)$ be the number of times the plus operation (i.e., + in the Else-case) is executed when running Algorithm $fib(n)$.

Which of the following is true for all $n \geq 0$?

- $P(n) = F(n) - 1$
- $P(n) = F(n+1) - 1$
- $P(n) = F(n+1)$
- $P(n) = F(n)$
- None of the other answers.
Question 2 (1 point)

Professor Mul Tiplier claims that she can multiply two \( n \)-bit integers in the following way:

Five times, recursively multiply two \( \left(\frac{n}{7}\right) \)-bit integers. Additionally, perform \( O(\sqrt{n}) \) extra amount of bit-operations.

Is Professor Tiplier's claim true or false?

- True
- False
Question 3 (1 point)

Consider the following algorithm that takes as input an integer $n \geq 1$:

Algorithm $\text{Random}(n)$:

If $n = 1$: print "Hello World"

Else: Let $k$ be a uniformly random element in the set \{1, 2, . . . , n − 1\}; $\text{Random}(k)$

End of algorithm

Let $n$ be a large integer. What is the expected running time of algorithm $\text{Random}(n)$?

- $\Theta(\sqrt{n})$
- $O(1)$
- $\Theta(n)$
- $\Theta(\log \log n)$
- $\Theta(\log n)$
Question 4 (1 point)

You are given an extremely clever data structure $DS$ that stores numbers, and supports the operations Insert and Delete_Min.

Consider the following algorithm that takes as input an arbitrary sequence $x_1, x_2, \ldots, x_n$ of numbers.

Step 1: Initialize an empty data structure $DS$.

Step 2: For $i = 1, 2, \ldots, n$: Insert $x_i$ into $DS$.

Step 3: For $i = 1, 2, \ldots, n$: Delete the smallest element from $DS$ and print it.

Is the following True or False: The running time of this algorithm is $\Omega(n \log n)$.

- True
- False
Question 5 (1 point)

Let $A[1 \ldots n]$ be a Min_Heap storing $n$ numbers. How much time does it take to sort the elements in this array?

- $\Theta(n^2)$
- $\Theta(1)$
- $\Theta(n)$
- $\Theta(n \log n)$
Midterm - Preview
Time Limit: 1:30:00  Time Left: 1:23:52  Michiel Smid: Attempt 1  Exit Preview

Page 6 of 17

Question 6 (1 point)
What is $\log 1 + \log 2 + \log 3 + \log 4 + \cdots + \log n$?

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

Submit Quiz  0 of 17 questions saved
Question 7 (1 point)

Professor He Ap claims that she has developed a data structure $DS$ that stores numbers and supports the following operations:

1. Insertion of a number in $o(\log n)$ time, where $n$ is the number of elements stored in $DS$.
2. Extract_Min in $o(\log n)$ time, where $n$ is the number of elements stored in $DS$.

Note that in 1. and 2. above, $o()$ is the little-Oh notation.

Is Professor Ap's claim True or False?

- True
- False

Submit Quiz 0 of 17 questions saved
In class, we have seen that a Heap can be visualized as a binary tree. Assume that, instead, we visualize a heap as a ternary tree, i.e., a tree in which each node has at most three children, all levels are full, except possibly the last level, but at the last level, all leaves are as far to the left as possible.

Let \( n \) be the number of nodes of this ternary heap. What is the height of this heap?

- None of the other answers.
- \( \lceil \log_3(2n) \rceil \)
- \( \lceil \log_3(3n/2) \rceil \)
- \( \lceil \log_3(2n/3) \rceil \)
- \( \lceil \log_3(3n) \rceil \)
Professor Justin Bieber has designed an algorithm that multiplies two $n$-bit integers in the following way:

Seven times, recursively multiply two ($n/5$)-bit integers. Additionally, perform $O(n)$ extra amount of bit-operations.

What is the running time of Professor Bieber's algorithm?

- None of the other answers.
- $\Theta(n \log_5 5)$
- $\Theta(n)$
- $\Theta(n^2)$
- $\Theta(n \log_7 7)$
Midterm - Preview

Time Limit: 1:30:00
Time Left: 1:21:47
Michiel Smid: Attempt 1

Question 10 (1 point)

Is the following True or False?

There exists an undirected graph with 85 vertices; 21 of these vertices have degree 7, whereas each of the other 64 vertices has degree 8.

- True
- False

Submit Quiz

0 of 17 questions saved
Question 11 (1 point)

Let $G = (V, E)$ be an undirected connected graph, and let $v$ be a vertex of $V$. After running algorithm $\text{explore}(v)$, it turns out that there is no back edge.

Is the following True or False: The graph $G$ must be a tree.

- True
- False
Question 12 (1 point)

Let $G = (V, E)$ be an undirected connected graph, and let $v$ be a vertex of $V$. After running algorithm $\text{explore}(v)$, it turns out that there is at least one back edge.

Is the following True or False: The graph $G$ cannot be bipartite.

- True
- False
Let $G = (V, E)$ be a directed graph, and let $v$ be a vertex of $V$. We run algorithm $\text{explore}(v)$. As we have seen in class, at termination, each vertex $u$ of $V$ has values $\text{pre}(u)$ and $\text{post}(u)$.

Is the following True or False: There can be an edge $(v, u)$ such that

$$\text{pre}(u) < \text{pre}(v) < \text{post}(u) < \text{post}(v).$$

- True
- False
Let \( G = (V, E) \) be a directed graph. We run depth-first search on \( G \), i.e., algorithm \( DFS(G) \).

Is the following True or False: If there is a directed cycle in \( G \) that contains a cross edge, then there is also a directed cycle in \( G \) that contains a back edge.

- True
- False

Submit Quiz  

0 of 17 questions saved
Question 15 (1 point)

You are given a sequence of \( n \) numbers, each of which is an integer in the set \( \{1, 2, \ldots, k\} \). Which of the following is true?

- The sequence can be sorted in \( \Theta(k) \) time.
- None of the other answers.
- Every algorithm that sorts the sequence takes \( \Omega(n \log n) \) time.
- The sequence cannot be sorted in \( \Theta(n + k) \) time.
- The sequence can be sorted in \( \Theta(n + k) \) time.
Let \( G = (V, E) \) be a directed acyclic graph, in which each edge \((u, v)\) has a real weight \( wt(u, v) > 0 \). Let \( s \) be a fixed source vertex of \( V \). Recall that \( \delta(s, v) \) denotes the length of a shortest directed path from \( s \) to \( v \). In class, we have seen an algorithm that computes \( \delta(s, v) \) for all vertices \( v \), in \( O(|V| + |E|) \) total time.

Is the following True or False: Both the algorithm and its running time analysis are still correct if edge weights can be negative.

- [ ] True
- [ ] False
Question 17 (1 point)

Michiel's Taxi Company is located in a city whose map is a directed acyclic graph \( G = (V, E) \). Each edge \((u, v)\) in this graph has a positive weight \( wt(u, v) \). When a customer wants to take a taxi from vertex \( s \) to vertex \( v \), Michiel takes the longest path from \( s \) to \( v \); we denote the length of this path by \( \delta_{\text{max}}(s, v) \).

What is the best time complexity to compute \( \delta_{\text{max}}(s, v) \) for all vertices \( v \)?

- \( \Theta(|V| + |E| \log |E|) \)
- \( \Theta(|V| \log |V| + |E| \log |E|) \)
- None of the other answers.
- \( \Theta(|V| + |E|) \)
- \( \Theta(|V| \log |V| + |E|) \)