SSN College of Engineering, Kalavakkam – 603 110
(An Autonomous Institution, Affiliated to Anna University, Chennai)
Department of Computer Science and Engineering
Continuous Assessment Test – I

Degree & Branch:	**B.E. Computer Science and Engineering**		Semester:	**4**
Subject Code & Name:	**UCS1403 DESIGN AND ANALYSIS OF ALGORITHMS**			
Academic Year:	**2020-2021**	Batch: 2019-2023	Date:	**18-03-2021**
Time: 10:30 am-12:00 pm	Answer All Questions		Maximum: 50 Marks	

Part A (30 min, $10 \times 2 = 20$ marks)

1. Identify basic operations. $\boxed{\text{CO2, K1}}$

   A. multiplication

   B. assignment

   C. exponentiation

   D. factorial

2. Choose the strongest statement about $10n^2 + 6n$. $\boxed{\text{CO2, K1}}$

   A. $10n^2 + 6n \in O(n^2)$.

   B. $10n^2 + 6n \in \Omega(n^2)$.

   C. $10n^2 + 6n \in \omega(n^2)$.

   D. $10n^2 + 6n \in \Theta(n^2)$.

3. $f(n) \in \Theta(g(n))$ if $\boxed{\text{CO2, K2}}$

   A. $\lim\limits_{n\to\infty} \dfrac{f(n)}{g(n)} = \infty$

   B. $\lim\limits_{n\to\infty} \dfrac{f(n)}{g(n)} = 0$

   C. $\lim\limits_{n\to\infty} \dfrac{f(n)}{g(n)} = c \in \mathbb{R}^+$

   D. $\lim\limits_{n\to\infty} \dfrac{f(n)}{g(n)} = 1$

4. Compare the growth rates of $\log 2n$ and $\sqrt{n}$. $\boxed{\text{CO2, K3}}$

   A. $\log 2n = O(\sqrt{n})$

   B. $\log 2n = \Omega(\sqrt{n})$

   C. $\log 2n = \Theta(\sqrt{n})$

D. $\log 2n$ and $\sqrt{n}$ cannot be compared.

5. Which of the following values are suitable for constants c and n_0 to prove that $10n^2+4n+2 = O(n^2)$? (Hint: $f(n) = O(g(n)) \equiv f(n) \leq cg(n)$ for $n \geq n_0$) $\boxed{\text{CO2, K3}}$

   A. $c = 16, n_0 = 1$

   B. $c = 14, n_0 = 2$

   C. $c = 11, n_0 = 5$

   D. $c = 10, n_0 = 6$

6. What is the asymptotic running time of the algorithm? $\boxed{\text{CO2, K2}}$

   for $i \leftarrow 1$ **to** m
       for $j \leftarrow 1$ **to** n
           $c[i,j] \leftarrow a[i,j] + b[i,j]$
       end
   end

   A. $\Theta(mn)$

   B. mn

   C. $O(n^2)$

   D. n^2

7. What is the running time of the algorithm? $\boxed{\text{CO2, K2}}$

   $c \leftarrow 0$
   for $i \leftarrow 1$ **to** 10 **do**
       for $j \leftarrow 1$ **to** i **do**
           $c \leftarrow c + 1$
       end
   end

   A. $O(n^2)$

   B. $O(n)$

   C. $O(1)$

   D. $O(10)$

8. What is the value of c when the loop terminates? $\boxed{\text{CO2, K3}}$

$n \leftarrow 1000$
$c \leftarrow 0$
while $n > 0$ **do**
  $c \leftarrow c + 1$
  $n \leftarrow \lfloor n/2 \rfloor$
end

A. 10

B. 1000

C. 500

D. 20

9. How many times $x \leftarrow x + 1$ is executed? CO2, K3

$x \leftarrow 0$
for $i \leftarrow 1$ **to** n **do**
  for $j \leftarrow i$ **to** $2n$ **do**
    $x \leftarrow x + 1$
  end
end

A. $\dfrac{3n(n-1)}{2}$

B. $2n(n-1)$

C. $2n^2$

D. n^2

10. If $T(n) = \frac{n^2}{2}$, then what is $T(2n)$? CO2, K2

A. $2n^4$

B. $2n^2$

C. $\dfrac{n^4}{4}$

D. $4n^4$

11. Find the time complexity of sum(a). CO2, K3

Algorithm: Sum a

Input: A *list* a
Output: $\sum$ a

if a = [] **then return** 0
return a[0] + Sum (a[1:])

A. $O(n)$

B. $O(n^2)$

C. $O(1)$

D. $O(\log n)$

12. The asymptotic growth rate of $T(n) = 9T\left(\frac{n}{3}\right) + n^2$ is CO2, K3

A. $\Theta(n^2 \log n)$

B. $\Theta(n^2)$

C. $\Theta(n^3)$

D. $\Theta(n \log n)$

13. What is the asymptotic solution of CO2, K3

$$T(n) = \begin{cases} 1 & \text{if } n = 0 \\ T(n-1) + n & \text{if } n > 0 \end{cases}$$

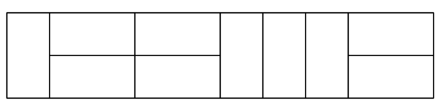
A. $O(n^2)$

B. $O(n)$

C. $O(2^n)$

D. $O(\log n)$

14. A domino is a 2×1 or 1×2 rectangle. How many different ways $T(n)$ are there to completely fill a $2 \times n$ rectangle with n dominos? CO2, K3



A 2×10 rectangle filled with 10 dominos

A. $T(n) = T(n-1) + T(n-2), T(2) = 2, T(1) = 1$

B. $T(n) = 1 + T(n-1) + T(n-2), T(2) = 2, T(1) = 1$

C. $T(n) = 1 + T(n-1), T(2) = 2, T(1) = 1$

D. $T(n) = 2 + T(n-2), T(2) = 2, T(2) = 1$

15. What is the running time of a bruteforce algorithm to solve the following problem: Given a set of n integers, does it contain a pair of elements a, b such that $a + b = 0$? CO2, K2

A. $O(n)$

B. $O(n^2)$

C. $O(n^3)$

D. $O(n \log n)$

16. What is the running time of a bruteforce algorithm to solve the following problem: Given a set of n integers, does it contain three elements a, b, c such that $a + b = c$? CO2, K3

A. $O(n)$

B. $O(n^2)$

C. $O(n^3)$

D. $O(n \log n)$

17. For each of the following functions, indicate how much the function's value will change if its argument is increased fourfold. CO2, K3
a. $\log_2 n$ b. $\sqrt{n}$ c. n^3 d. 2^n

A. $\log_2 n$

B. $\sqrt{n}$

C. n^3

D. 2^n

18. Arrange these functions in increasing order of asymptotic growth: $c^n, n^2 \log n, n^{2.1}, \log^4 n$ CO2, K3

A. $\log^4 n, n^2 \log n, n^{2.1}, c^n$

B. $\log^4 n, n^{2.1}, n^2 \log n, c^n$

C. $n^2 \log n, n^{2.1}, \log^4 n, c^n$

D. $n^{2.1}, n^2 \log n, \log^4 n, c^n$

19. What is the worst-case and best-case running time of the algorithm? CO2, K3

$i \leftarrow n$
until $i = 0$ `or` $a[i-1] \leq a[i]$ **do**
  swap $a[i-1], a[i]$
  $i \leftarrow i - 1$
end

A. $n, 1$

B. n, n

C. $n, \frac{n}{2}$

D. $n^2, 1$

20. What is the running time of insertion sort if all the elements are equal? CO2, K2

A. $O(n)$

B. $O(n^2)$

C. $O(\log n)$

D. $O(1)$

21. Two complex numbers $x = a + bi$ and $y = c + di$ can be multiplied as CO2, K3

$$xy = ac + (ad + bc)i + bd$$

Which of the following sets will realize xy using minimum number of multiplications?

A. ac, bd, ad, bc

B. $ac, ad + bc, bd$

C. $ac, bd, ac + bd - (a - b)(c - d)$

D. $ac, bd, (a + b)(c + d) - ac - bd$

Part B (40 min, 20 marks)

22. We are given an array of n numbers and a target number K. We want to determine if there are two numbers in the array whose sum equals the target K. For instance, if the input is `[8, 4, 1, 6]` and K is 10, then the answer is yes (4 and 6 sum to 10). A number may be used twice. Construct a brute-force algorithm to solve this problem. Analyse the running time of the algorithm. (10) CO1, K2

1.3.1, 1.4.1, 2.1.3

23. Suppose you have an array of n elements in which each element contains one of the three distinct keys: `true`, `false`, or `maybe`. Design an $O(n)$ algorithm to rearrange the array so that all `false` elements precede `maybe` elements, which in turn precede `true` elements. That is, the postcondition of the algorithm is CO1, K3

1.4.1, 2.1.3

$$a[1\,..\,i] = \texttt{false}, a[i+1\,..\,j] = \texttt{maybe}, a[j+1\,..\,n] = \texttt{true}$$

You may use only constant extra space. Prove the running time. (10)

Part C (20 min, 10 marks)

24. Computing a^n using any of the following recursive definitions will give a faster algorithm. CO1, K3

1.4.1, 2.1.3

$$a^n = \begin{cases} 1 & n = 0 \\ (a^{n/2})^2 & n \neq 0 \text{ and even} \\ a \times (a^{\lfloor n/2 \rfloor})^2 & n \text{ is odd} \end{cases}$$

or

$$a^n = \begin{cases} 1 & n = 0 \\ (a^2)^{n/2} & n \neq 0 \text{ and even} \\ a \times (a^2)^{\lfloor n/2 \rfloor} & n \text{ is odd} \end{cases}$$

a) Construct a recursive algorithm from each of these definitions.
b) Trace the execution of each algorithm on 2^{11}.
c) Which of the two algorithms is more suitable for constructing a tail recursive (iterative) algorithm? Why?
d) Construct an iterative algorithm.
e) Show that the running time of the algorithm is $O(\log n)$.