

1	<p>1. $(1010.1110)_2 = (\text{_____})_{10}$ A) 4.2345 B) 9.456 C) 10.875 D) 11.657</p>
2	<p>1,2,3,4,5 is pushed in an empty stack in the given order, then four times pop operation is done. After each pop operation element is enqueued in an empty queue. Then two deque operations are done and after each dequeue elements are pushed in the stack. Topofstack() will return A) 1 B) 2 C) 3 D) 4</p>
3	<p>Which of the following is False A) Cost of Memory/unit is increasing B) As we move closer to the processor the size of memory units is decreasing C) As we move closer to the processor the speed of memory/unit is increasing D) Speed of the processors in terms of BIPS(billion instructions per second) is increasing</p>
4	<p>What will be the output of the following program main() { int count,i,x; for(count=1,x=0,i=0; count<=4;count++,i++) { x=i%2; if(x==0) continue; else { printf("%d\n",x); continue; } } printf("%d",count); }</p> <p>A. 2 4 5 B. 1 1 5 C. 2 5 5 D. 1 2 2</p>
5	<p>For calculating the sum of 100 numbers stored in the array we have following code for (int x = 0; x < 100; x++) { S=S+a[x]; } The above code can be replaced by one of the following code which will involve less operations A) x=0; axvalue=A[x]; for (int x = 0; x < 100; x++) { S=S+axvalue; } B) for (int x = 0; x < 100; x=x+2) { S=S+a[x]+(x+1); } C) for (int x = 0; x < 100; x=x+3) { S=S+a[x]+a[x+1]+a[x+2]; } D) for (int x = 0; x < 100; x=x+2) {</p>

	<pre>S=S+a[x]+a[x+1]; }</pre>
6	<p>If $f(n) = \Theta(g(n))$ and $g(n) = \Theta(h(n))$, then $h(n) = \Theta(f(n))$</p> <p>A) True B) false C) depends on the value of $f(n)$ D) depends on the value of $g(n)$</p>
7	<p>4. $n/100 = \Omega(n)$</p> <p>A) True B) false</p>
8	<p>Convert $(A-B)*(D/E)$ into a postfix expression</p> <p>A) AB-DE/* B) A*BD/E- C) ABDE-/* D) /-ABDE</p>
9	<p>If insertion sort runs in $8n^2$ steps and merge sort runs in $64n \lg n$ steps, for which values of n does insertion sort becomes slower than merge sort</p> <p>A) 8 B) 32 C) 64 D) 128</p>
10	<p>The terminal nodes of a binary tree occur in the same relative position in a) preorder b) inorder c) postorder</p> <p>A) a & b & c B) b & c only C) c & a only D) a & b only</p>