Question 1 — 1 pts

What is the time complexity of fun()?

```
int fun(int n)
{
  int count = 0;
  for (int i = n; i > 0; i /= 2)
     for (int j = 0; j < i; j++)
        count += 1;
  return count;
}
```

(A) O(n^2)
(B) O(nLogn)
(C) O(n)
(D) O(nLognLogn)

- d
- b
- c
- a

Question 2

1 pts

Let W(n) and A(n) denote respectively, the worst case and the average case running time of an algorithm executed on an input of size n. which of the following is ALWAYS TRUE?

a) $A(n) = \Omega(W(n))$

b) $A(n) = \Theta(W(n))$

c) $A(n) = O(W(n))$

d) $A(n) = o(W(n))$

- ○ a
- ● c
- ○ d
- ○ b

Question 4

1 pts

In a competition, four different functions are observed. All the functions use a single for loop and within the for loop, same set of statements are executed. Consider the following for loops:

A) for(i = 0; i < n; i++)

B) for(i = 0; i < n; i += 2)

C) for(i = 1; i < n; i *= 2)

D) for(i = n; i > -1; i /= 2)

If n is the size of input (positive), which function is most efficient(if the task to be performed is not an issue)?

- ○ D
- ● C
- ○ B
- ○ A

Question 3

1 pts

Which of the given options provides the increasing order of asymptotic complexity of functions f1, f2, f3, and f4?

f1(n) = 2^n

f2(n) = n^(3/2)

f3(n) = nLogn

f4(n) = n^(Logn)

(A) f3, f2, f4, f1
(B) f3, f2, f1, f4
(C) f2, f3, f1, f4
(D) f2, f3, f4, f1

- ○ D
- ○ C
- ○ B
- ○ A

Question 5

1 pts

Consider the recurrence equation $T(n) = 2T(n-1)$, if $n>0$

$= 1$, otherwise

Then T(n) is (in big O order)

A) O(n)

B) $O(2^n)$

C) O(1)

D) O(nlogn)

- ○ C
- ○ D
- ● B
- ○ A

Time R

Attempt

59 Min

Question 6

1 pts

Consider the program

```
void function(int n) {
int i, j, count=0;
for (i=n/2; i <= n; i++)
for (j = 1; j <= n; j = j*2)
count++;}
```

The complexity of the program is

a) O(log n)

b) $O(n^2)$

c) $O(n^2 \log n)$

d) O(n log n)

- b
- d
- a

Question 7

1 pts

The average case and worst case complexities for the Merge sort algorithm are

a) $O(n^2), O(n^2)$

b) $O(n^2), O(n \log_2 n)$

c) $O(n \log_2 n), O(n^2)$

d) $O(n \log_2 n), O(n \log_2 n)$

- b
- a
- d
- c

Question 8

1 pts

What is the value of following recurrence. $T(n) = 5T(n/5) + 1$, $T(1) = 1$, $T(0) = 0$

a) Theta (n)

b) Theta (n^2)

c) Theta (nlogn)

d) Theta (n^3)

- ○ b
- ● d
- ● a
- ● c

Question 9

1 pts

The running time of an algorithm is represented by the following recurrence relation:

if n <= 3 then T(n) = n

else T(n) = T(n/3) + cn

Which one of the following represents the time complexity of the algorithm?

(A) Θ (n)

(B) Θ (n log n)

(C) Θ (n^2)

(D) Θ (n^2log n)

- ○ d
- ○ c
- ○ a
- ○ b

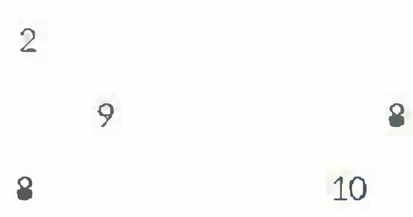
Question 10

1 pts

Delete the minimum element in the Min Binary Heap Tree given below (The links are not drawn and assume the obvious links exist):

Answers:

I



II

III

IV

a) I

b) II

c) III

d) IV

- b
- a
- c

Time Running:

Attempt due: Nov 4 a

43 Minutes, 48 S

Question 11

1 pts

Insert 16 to the Binary Max Heap tree below (The links are not drawn and assume the obvious links exist):

Answers: I II III IV

a) I

b) II

c) III

d) IV

- ○ b
- ○ d
- ○ c

Question 12

1 pts

Which of the following is a correct ordering, from best to worst of algorithm speeds?

a) Linear, Exponential, Polynomial, Logarithmic

b) Logarithmic, Linear, Polynomial, Exponential

c) Exponential, Polynomial, Linear, Logarithmic

d) Logarithmic, Exponential, Linear, Polynomial

e) Logarithmic, Linear, Exponential, Polynomial

- ○ b
- ○ a
- ○ e
- ○ c
- ○ d

Question 13 — 1 pts

Suppose you have an algorithm that operates on a set of data with n elements. If the recurrence formula that computes the time required for the algorithm is given by

$$T(n) = \begin{cases} 2T\left(\frac{n}{2}\right) + Dn & if\ n > 1 \\ C & if\ n = 1 \end{cases}$$

where D and C are constants, which of the following gives the order of complexity of the algorithm?

a) logn

b) nlogn

c) n

d) n^2

e) None of the other answers

- ○ c
- ○ e
- ○ d
- ○ b
- ○ a

Tim
Atte
28 M

Question 14

1 pts

The recurrence relation for the Merge Sort algorithm is defined by:

1. $T(n)=T(n-1)+1$
2. $T(n) = 2T(n/2)+1$
3. $T(n)=T(n-1)+\text{Theta}(n)$
4. $T(n)=2T(n/2)+\text{Theta}(n)$
5. None of the above

- ○ 1
- ○ 3
- ○ 5
- ○ 2
- ● 4

Time Runr

Attempt due:

25 Minute

Question 15

1 pts

Which of the following list(s) is (are) not max-heap based on the MaxHeapify algorithm?

a) A = {23, 17, 14, 16, 12, 10, 13, 11, 15}
b) A = {56, 45, 20, 44, 30, 10, 15, 39, 40}
c) A = {56, 23, 50, 20, 18, 45, 49, 19, 17}
d) A = {58, 30, 48, 25, 18, 45, 49, 19, 17}
e) None of the other answers

- ○ b
- ○ d
- ○ e
- ○ c
- ○ a

Question 16

The solution for the recurrence relations

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

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

is respectively

a) O(1), O(n)

b) O(n), O(logn)

c) O(n^2), O(logn)

d) O(logn), O(logn)

The runtime complexity, T(n), of the three following recurrence relation solved by Master's Theorem) are

$T(n) = 6T(n/3) + n^2 \log n$

$T(n) = 64T(n/8) - n^2 \log n$

$T(n) = T(n/2) + n/\log n$

(The solution for the three relations are respectively given by)

A: $\Theta(n \log n)$, $\Theta(n^2)$, $\Theta(n^2 \log n)$

B: $\Theta(n^2)$, $\Theta(n^2 \log n)$, $\Theta(n \log n)$

C: $\Theta(n^2 \log n)$, Master's Theorem does not apply, $\Theta(n^2)$

D: $\Theta(n^2)$, Master's Theorem does not apply, $\Theta(n^2 \log n)$

E: $\Theta(n^2 \log n)$, $\Theta(n^2)$, Master's Theorem does not apply

F: $\Theta(n^2)$, $\Theta(n^2 \log n)$, Master's Theorem does not apply

Question 18

1 pts

In Hire-Assistant problem (hiring n candidates):

1. What is the probability that you hire exactly one time?
2. What is the probability that you hire exactly n time?

a) 1/n and n/n

b) 1 and 1/n

c) 1/n and (n-1)/n

d) 1/n and 1/n!

e) 1/n and (n-1)/n!

- ○ c
- ○ e
- ○ b
- ○ a
- ◉ d

Next ▸

Tim
Atte
13

Question 19

1 pts

The solution to the recurrence $T(n) = T(n/6) + T(7n/9) + O(n)$ is $O(n)$. (Assume $T(n)=1$ for n smaller than some constant c).

a) True

b) False

- (●) a
- ( ) b

Time Ru
Attempt
9 Minu

Question 20

1 pts

How many comparisons are required in insertion sort to sort a file if the file is sorted in reverse order?

1. N^2
2. N
3. N-1
4. N/2

- ○ 2
- ○ 1
- ○ 3
- ○ 4