Institute of Business & Information Technology

University of the Punjab

Quaid-e-Azam Campus, Lahore

Sheet No. _____

Assignment 02

Spring Term 2019

Code IT-466
Title Analysis of Algorithm
Due Date: Feb 26, 2019 before 0815hr

Degree BBIT
Batch F16-IT Specialization
Marks 100

Instructions:

1. Do not forget to pray before starting to attempt the paper. Trust me it helps.
   Remember! SOMEONE is always with you (Be Relaxed), and HE is also watching you (Be Honest)
2. **Question Paper is SELF EXPLANATORY. Understanding the Question Paper is part of Solution.**
3. Nothing Beyond the Finish Line will be Evaluated. Back Side of Pages is Beyond Finish Line.
4. For Calculations etc. Use the back side of the pages.
5. Error in Question will be advantageous to Student.
6. Read the Questions carefully before attempting.
7. Solve your paper using Black/Blue Pen only.
8. Attempt All Questions in a Precise Fashion.
9. Switch Off your Cellular Phones.
10. Manage Your Time.

GOOD LUCK

Reg. No: _____ **Name:** _____

Class: IT Specialization **Date:** Friday, Feb 22, 2019

Section: _____ **Signature:** _____

Q 01.	Q 02.	Q 03.	Q 04.	Q 05.	Q 06.	Q 07.	Q 08.	Q 09.	Q 10.	Total
15	15	15	15	15	15	15	15	15	15	150

Assistant's Signature

Examiner's Signature

Do Not Open until YOU are Advised by the Invigilator.

Question No 01. 15

Find the time complexity of the following iterative algorithms?

```
for(i=1; i<=n/2; i++)
    for(int j=1; j^3<=n; j++)
        for(int k=1; k<=n; k=k×2)
            Print("IBIT-PU");
```

4

$n/2$

$\sqrt[3]{n}$

$\log_2 n$

$$T(n) = \frac{n}{2} \log_2 n \cdot \sqrt[3]{n} = \frac{1}{2} n^{4/3} \log_2 n$$

$$= O\left(n^{4/3} \log_2 n\right)$$

```
for(j=1; j<=n; j++)
    for(i=1; i<=50; i++)
        for(c=1; c<=n/2; c+=2)
            print("IBIT");
```

3

n

50

$n/4$

$$T(n) = n \times 50 \times \frac{n}{4}$$

$$= \frac{25}{2} n^2 \Rightarrow O(n^2)$$

```
for(i=n/2; i>0; i --)
    for(j=i+1; j<=n; j ++)
        Print("IBIT");
```

4

$n/2$

$n/2$

$$T(n) = \frac{n}{2} \times \frac{n}{2} = \frac{n^2}{4}$$

$$\Rightarrow O(n^2)$$

```
for(int i=1;i <= n;i=i×2)
    for(int j=1; j <= 2×i; j++)
        Print("IBIT-PU")
    for(int k=1; k <=3×i; k++)
        Print("AoA")
```

4

$\log_2 n$

$\frac{2}{2} \log_2 n$

$\frac{1}{3} \log_2 n$

$$T(n) = \log_2 n \left[\frac{2}{2} \log_2 n + \frac{3}{3} \log_2 n \right]$$

$$= \log_2 n \; 5 \log_2 n = 5 (\log_2 n)^2$$

$$\Rightarrow O(\log_2 n)^2$$

Question No 02.

Find the time complexity of $T(n) = 5T\left(\frac{n}{5}\right) + cn^5$ using Recursion Tree?

Level	Subproblem		Workpower
0	$5^0 = 1$	cn^5	cn^5
1	5	$c\left(\frac{n}{5}\right)^5 \quad c\left(\frac{n}{5}\right)^5 \quad c\left(\frac{n}{5}\right)^5 \quad c\left(\frac{n}{5}\right)^5 \quad c\left(\frac{n}{5}\right)^5$	$5c\left(\frac{n}{5}\right)^5$
2	5^2	$c\left(\frac{n}{5^2}\right)^5 \quad c\left(\frac{n}{5^2}\right)^5 \quad c\left(\frac{n}{5^2}\right)^5 \quad c\left(\frac{n}{5^2}\right)^5 \quad c\left(\frac{n}{5^2}\right)^5$	$5^2 \cdot c\left(\frac{n}{5^2}\right)^5$
i	5^i	$c\left(\frac{n}{5^i}\right)^5$	$5^i c\left(\frac{n}{5^i}\right)^5$
$h-1$	5^{h-1}	$c\left(\frac{n}{5^{h-1}}\right)^5$	$5^{h-1} c\left(\frac{n}{5^{h-1}}\right)^5$
h	5^h	$T(1)$	$5^h T(1)$

$$\frac{n}{5^h} = 1 \Rightarrow h = \log_5 n$$

$$T(n) = 5^h \cdot T(1) + \sum_{i=0}^{h-1} 5^i c \cdot \left(\frac{n}{5^i}\right)^5$$

$$= n(1) + c\sum_{i=0}^{h-1} \frac{n^5}{5^{4i}} \qquad r = \frac{1}{5^4} < 1$$

$$+ cn^5 \sum_{i=0}^{h-1} \frac{1}{5^{4i}}$$

$$+ cn^5 \left[\frac{1\left(1 - 1/5^{4(h-1)}\right)}{1 - \frac{1}{5^4}}\right]$$

$$+ cn^5 \cdot \frac{625}{624}\left[1 - \frac{1}{5^{4(\log_5 n - 1)}}\right]$$

$$+ cn^5 \left[1 - \frac{1}{5^{4\log_5 n} \cdot 5^{-4}}\right]$$

$$+ cn^5 \left[1 - \frac{5^4}{n^4}\right]$$

$$+ cn^5 \left[\frac{n^4 - 625}{n^4}\right]$$

$$+ cn(n^4 - 625) = n + cn^5 - 625cn$$

$$= O(n^5)$$

Question No 03. 15

Find the Time Complexity of the $T(n) = 3T\left(\frac{n}{4}\right) + n^3$ using Recursion Tree?

Question No 04. 15

Find the Time Complexity of the $T(n) = T\left(\frac{n}{4}\right) + T\left(\frac{3n}{4}\right) + n$ using Recursion Tree?

For lower Bound,

$$T(n) = 2T\left(\frac{3n}{4}\right) + n$$

L	SP		WP
0	1	n	n
1	2	$\frac{3n}{4}$ $\quad$ $\frac{3n}{4}$	$2\left(\frac{3}{4}n\right)$
2	2^2	$\left(\frac{3^2}{4^2}n\right)$ $\left(\frac{3^2}{4^2}n\right)$ $\left(\frac{3^2}{4^2}n\right)$ $\left(\frac{3^2}{4^2}n\right)$	$2^2\left(\frac{3^2}{4^2}n\right)$
i	2^i	$\left(\frac{3^i}{4^i}n\right)$	$2^i\left(\frac{3^i}{4^i}n\right)$
$h-1$	2^{h-1}	$\left(\frac{3^{h-1}}{4^{h-1}}n\right)$	$2^{h-1}\left(\frac{3^{h-1}}{4^{h-1}}n\right)$
h	2^h	$T(1)$	

$$\frac{h}{2^h} = 1 \Rightarrow h = \log_2 n$$

$$T(n) = 2^h T(1) + \sum_{i=0}^{h-1} \frac{n\left(i - (3/4)^h\right)}{1 - \frac{3}{4}}$$

$$= 2^{\log_2 n} \cdot 1 + n\left[\frac{1 - \left(\frac{3}{4}\right)^{\log_2 n}}{1/4}\right]$$

$$= n + 4n\left(\frac{1 - 3^{\log_2 n}}{4^{\log_2 n}}\right)$$

$$= n + 4n\left(1 - \frac{3^{\log_2 n}}{n^2}\right)$$

$$= n + 4n\left(\frac{n^2 - 3^{\log_2 n}}{n^2}\right)$$

$$= n + 4n - 4\frac{3^{\log_2 n}}{n}$$

$$= O(n)$$

Question No 05. 15

Find the Time Complexity of the Algorithms using Reverse Substitution, Show Working.

$$T(n) = 2T(n-1) + cn^2$$

$T(n) = 2T(n-1) + cn^2$

$T(n) = 2^2 T(n-2) + 2c(n-1)^2 + cn^2$

$= 2^3 T(n-3) + 2c(n-2)^2 + 2c(n-1)^2 + cn^2$

$T(n-k) = 2^k T(n-k) + 2c(n-(k+1))^2 + \cdots + 2c(n-1)^2 + cn^2$

$n-k = 1 \Rightarrow k = n-1$

$T(n) = 2^{n-1} T(1) + 2c(0)^2 + 2c(1)^2 \cdots + 2c(n-1)^2 + cn^2$

$= 2^{n-1} \cdot 1 + 2c(1)^2 + \cdots + 2c(n-1)^2 + cn^2$

$= 2^{n-1} + 2c\left[1^2 + 2^2 + 3^2 \cdots n^2\right]$

$= 2^{n-1} + 2c \cdot \frac{n(n+1)(n+2)}{6}$

$T(n) = 2^{n-1} + \frac{1}{3}cn(n+1)(n+2)$

$= O(2^{n-1})$

Roll No.: _____

Name _____

Question No 06. **15**

Find the Time Complexity of the Algorithms using Reverse Substitution, Show Working.

$$T(n) = T(n-1) + cn^{-1}$$

$$T(n) = T(n-1) + cn^{-1}$$

$$= T(n-2) + c(n-1)^{-1} + cn^{-1}$$

$$= T(n-3) + c(n-2)^{-1} + c(n-1)^{-1} + cn^{-1}$$

$$T(n-K) = T(n-K) + c(n-(K+1))^{-1} + \cdots + cn^{-1}$$

$$n-K = 1 \Rightarrow K = n-1$$

$$T(n) = T(1) + c(1)^{-1} + c2^{-1} \cdots + cn^{-1}$$

$$= c\left[1 + \frac{1}{2} + \frac{1}{3} \cdots + \frac{1}{n}\right]$$

$$= c\log n$$

$$= O(\log n)$$

15

Question No 07.

Find the Time Complexity of the Algorithms using Reverse Substitution, Show Working.

$$T(n) = T(n-2) + T(n-3) + cn$$

15

Question No 08.

Find the Time Complexity of the algorithms with given time equations

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

Question No 09. 15

Find the Space Complexity $S(n)$ of the following Iterative Algorithms, Show Working.

4

```
Func()
    int X[m,n,p], long Y[m+n+p]
    int i=1, j=1, k=1, b=1
    while(i<=m)
        j=1
        while(j<=n)
            k=1
            while(k<=p)
                Read X[i,j,k]
                Y[b++]= X[i,j,k] × X[i,j,k]
```

mnp , $m+n+p$

3

$S(m,n,p) = mnp + m+n+p + 3$

$= O(mnp)$

3

```
Func()
    int X[n]
    int Y[n,n]
    for(int a=1; a<=n; a++)
        for (int b=1; b<=n; j++)
            Y[a,b]=X[a]+X[b]
```

n

n^2

1

1

$S(n) = n + n^2 + 2$

$= O(n^2)$

4

```
Func(X, n)
    int a=1, b=1
    for(a=1; a<=n; a=a*2)
        int Q[n×n]
        for(b=1; b<=n2; b++)
            Q[b]=X[a]
```

2

n^2

$S(n) = n^2 + 2$

$= O(n^2)$

4

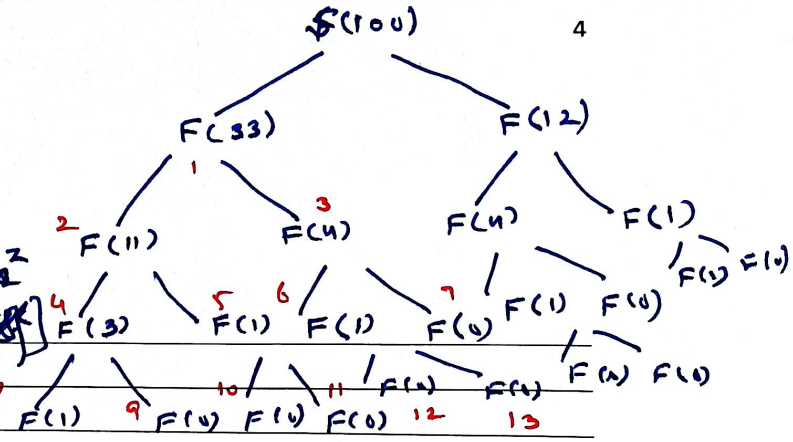
```
MyFunc(int Z)
    int M=1
    if (Z<1)
        return 1
    for (i=1; i<=Z; i*=2)
        M = M * i
    MyFunc(Z/3)
    for(j=1; j<=S; j++)
        M=Z-M
    MyFunc(Z/8)
```

1

1

$S(z) = 2 + k\log_2 z$

$\Rightarrow S(n) = O(\log_2 z)$



Question No 10. 15

Write down the Formal Statement of Master Theorem 6

$T(n) = aT\left(\frac{n}{b}\right) + n^k \log^p n$

Find the Time Complexity of the Following Recursive Algorithms using Master Theorem, Show Working.

$T(n) = 2T\left(\frac{n}{2}\right) + n^2 \log^{-1} n$ 3

$a >= 1$
$b > 1$
$K >= 0$

$a = 2, \quad b = 2, \quad K = 2, \quad p = -1$

$a < b^k, \quad 2 < 4$

$p = -1 \Rightarrow p < 0$

$2 = 2^2$
$2 = 4$
$2 < 4$

$T(n) = O(n^k)$
$= O(n^2)$

$T(n) = 7T\left(\frac{n}{2}\right) + \theta\left(\frac{1}{\log_2 n}\right)$ 3

$a = 7, \quad b = 2, \quad k = 2$

$a > b^k$

$T(n) = O(n^{\log_b a})$

$T(n) = O(n^{\log_2 7})$

$7 > 2^2$
$7 > 4$

$T(n) = 4T\left(\frac{n}{2}\right) + \frac{1}{n^4} \log_2 n$ 3

$a = 4 \quad b = 2 \quad K = -4 \quad \Rightarrow K >= 0$

Not applicable

Finish Line

Nothing beyond this line will be evaluated