

# Day-2: 20-10-2020 MODULE -1 Assignment - 1

1. **Obtain the asymptotic bound for the recurrence relations given below using Master theorem.**

1. T (n) = 3T (n/2)+ n2

2. T (n) = 4T (n/2)+ n2

3. T (n) = T (n/2) + 2n

4. T (n) = 2nT (n/2) + nn

5. T (n) = 16T (n/4)+ n

6. T (n) = 2T (n/2)+ n log n 7. T (n) = 2T (n/2)+ n/ log n 8. T (n) = 2T (n/4)+ n0.51 9. T (n) = 0.5T (n/2)+ 1/n 10. T (n) = 16T (n/4)+ n!

11. T (n) = p2T (n/2) + log n 12. T (n) = 3T (n/2)+ n

13. T (n) = 3T (n/3)+ pn

14. T (n) = 4T (n/2)+ cn

15. T (n) = 3T (n/4)+ n log n 16. T (n) = 3T (n/3)+ n/2

17. T (n) = 6T (n/3)+ n2 log n

18. T (n) = 4T (n/2)+ n/ log n 19. T (n) = 64T (n/8)− n2 log n 20. T (n) = 7T (n/3)+ n2

21. T (n) = 4T (n/2)+ log n

# Solve the following recurrence relation using recursion tree method.

1. T(n) = T(n/5) + T(4n/5) + n

2. T(n) = 3T(n/4) + cn2

3. T(n) = cn+2T(n/2)

# Module 2: Combinatorial Optimization

1. Design a greedy algorithmic technique using binary min heap to encode the word ***‘mississippi’*** using variable length codeword. Calculate the number of bits may be required for encoding the message ‘mississippi’?
2. a=3 b=2 k=2 p=0 logba≃1.5 since logba<k (case 3)

Answer: θ(n2)

1. a=4 b=2 k=2 p=0 logba≃2 since logba=k (case 1)

Answer: θ(n2logn)

1. a=1 b=2 k=1 p=1 logba≃0 since logba<k (case 3)

Answer: θ (2)

1. Masters theorem does not work a not a constant
2. a=16 b=4 k=1 p=0 logba≃2 since logba>k (case 1)

Answer: θ(n2)

1. a=2 b=2 k=1 p=1 logba≃1 since logba=k (case 2)

Answer: θ(nlog2n)

1. a=2 b=2 k=1 p=-1 logba≃1since logba=k (case 2)

Answer: θ(nloglogn)

1. a=2, b=4, k=0.51 p=0 logba≃0.5 since logba<k(case3)

Answer: θ (n0.51)

1. a=0.5 cannot be solved using masters theorem a<1
2. a=16 b=4 k=c p=1 logba≃2 since logba<k (case 3)

Answer: θ(n!)

1. a=16 b=4 k=c p=1 logba≃2 since logba<k (case 3)

Answer: θ(n!)

1. a=3 b=2 k=1 p=0 logba≃1.58 since logba>k (case1)

Answer: θ(nlog23)

1. a=3 b=3 k=0.5 p=0 logba≃1 since logba>k (case 1)

Answer: θ(n)

1. a=4 b=2 k=1 p=0 logba≃2 since logba>k (case 1)

Answer: θ(n2)

1. a=3 b=4 k=1 p=1 logba≃0.79 since logba<k (case 3)

Answer: θ(nlogn)

1. a=3 b=3 k=1 p=0 logba≃1 since logba=k (case 2)

Answer: θ(nlogn)

1. a=6 b=3 k=2 p=1 logba≃1.63 since logba<k(case3)

Answer: θ(n2logn)

1. a=4 b=2 k=1 p=-1 logba≃2 since logba>k (case1)

Answer: θ(n2)

1. a=64 b=8 k=2 p=1 logba≃2 since logba=k (case2)

Answer: θ(n2log2n)

1. a=7 b=3 k=2 p=0 logba≃1.77 since logba<k (case3)

Answer: θ(n2)

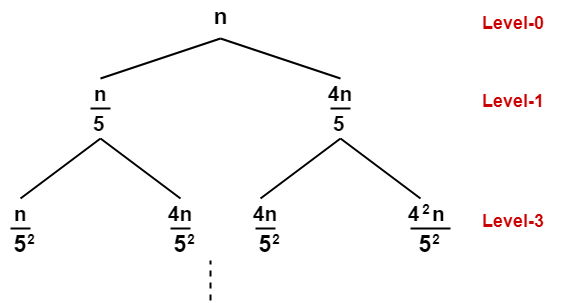
1. a=4 b=2 k=0 p=1 logba≃2 since logba>k (case1)

Answer:

.θ(n2)

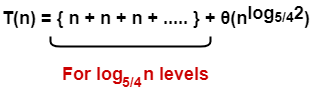
2. Solve the following recurrence relation using recursion tree method:

**1. T(n) = T(n/5) + T(4n/5) + n**



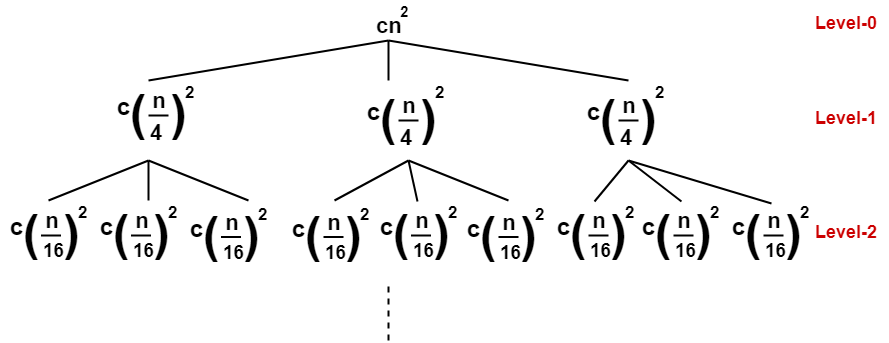
* Cost of level-0 = n
* Cost of level-1 = n/5 + 4n/5 = n
* Cost of level-2 = n/52 + 4n/52 + 4n/52 + 42n/52 = n

**Cost of last level = 2log5/4n x T (1) = θ(2log5/4n) = θ(nlog5/42)**



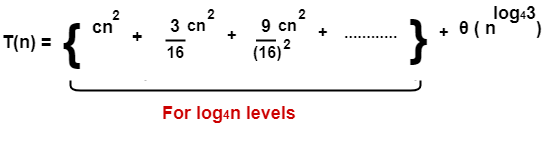
**Ans .  θ(nlog5/4n)**

**2. T(n) = 3T(n/4) + cn2**



* Cost of level-0 = cn2
* Cost of level-1 = c(n/4)2 + c(n/4)2+ c(n/4)2 = (3/16) cn2
* Cost of level-2 = c(n/16)2 x 9 = (9/162) cn2

**Cost of last level = nlog43 x T(1) = θ(nlog43)**



On solving, we get-

= (16/13) cn2 {1 – (3/16) log4n} + θ(nlog43)

= (16/13) cn2 – (16/13) cn2 (3/16) log4n + θ(nlog43)

**Ans. = O(n2)**

**3.T(n) = cn + 2t(n/2)**

T(n) T(n) T(n) = Cn

**T(n) = Cn**

**+ +**

**C(n/2) C(n/2)**

**+ + + +**

**T(n/4) T(n/4) T(n/4) T(n/4)**

**nk = θ(nlogn)+c**

assume that n/2^k = 1:

n = 2^k

**k = logn**

**3.**Design a greedy algorithmic technique using binary min heap to encode the word ***‘Mississippi’*** using variable length codeword. Calculate the number of bits may be required for encoding the message ‘Mississippi’?

Word: Mississippi

|  |  |  |
| --- | --- | --- |
| **Letter** | **Frequency** | **Binary** |
| **M** | **1** | **000** |
| **I** | **4** | **01** |
| **S** | **4** | **1** |
| **P** | **2** | **001** |

**Huffman tree:**

11

0

7

0

**3**

3

0 1 1 1

**M P I S**

**1 2 4 4**

**Character code = 4 \* 8 = 32 bits**

**Huffman code = (2\*3) +2+1 = 9 bits**

**Total = 41 bits**

**Total size of message = (1\*3) +(4\*2) +(4\*1) +(2\*3) = 21 bits**

∴ Total compressed size = 62 bits

****

* **M.Sathishkumar**
* **E0119052**
* **B. Tech CSE AI & ML**

