STS Notes

- float - > 7 decimals
  double - > 15 decimals
- byte - > 1
  short - > 2
  int - > 4
  long - > 8
  float - > 4
  double - > 8
  char - > 2
  boolean - > 1 in bytes [1 byte = 8 bits]
- String Constant Pool (SCP)
  String s1=”Andhra”; // this points to value in SCP
  String s2=new String(“Andhra”) // this makes a new object each time in heap

  s1.equal(s2) // True
  s1 == s2 // False
- System.out.println()   System.in
  class | object | function   class | object
- Scanner sc=new Scanner(System.in)
  char ch=sc.next().charAt(0);
  int n=sc.nextInt();
  String s=sc.next();
  String str=sc.nextLine();
- Note : (n & -1) = n
  C1   C2
  1 . & - > T or F | Check next condition
  2. && - > only if T | Check next condition

  a=5 b=5 c=5
  if((a>b) && (++a>c))
  ++a;
  Sopln(a); // 6
  if(a ^ ((a^b) & -(a<b?1:0)))
  ++c;
  Sopln(c);

  3. << (left shift) Eg – 5 is 0000 0101 then 5<<2 is 0001 0100 that is 20 in decimal
  >> (right shift) Eg - -8 is 1111 1000 then -8 >> 2 is 0011 1110 that is -2 in decimal
  4 . ? :
- 1 | n| log n | n log n | n2
- Big 0 tells worst case and upper bound
  Big omega tells best case and lower bound

- PPT 1 ,4 & 5 to be read once :)

TC | SC

Simple Sieve $N \log\log N$ | N
Segmented Sieve ,, | $\sqrt{N}$
Incremental Sieve ,, | N

Euler Phi $\sqrt{n} \log n$ | 1

$$\phi(n) = \begin{cases} n-1, & \text{if } n \text{ is prime} \\ n\left(1-\frac{1}{p_1}\right)\left(1-\frac{1}{p_2}\right)\cdots, & \text{otherwise} \end{cases}$$

$p_1, p_2, \ldots$ are prime factors

CRT $n \log m$ | 1

Strobogrammatic number N | 1

0 → 0, 1 → 1, 6 → 9, 8 → 8, 9 → 6

$0, 1, 8$ → Strobo + Palindrome

Binary Palindrome $\log n$ | 1

Booth's Algorithm

Block Swap / Rotating array n | 1

Euclid's Algorithm $\log(\min(a,b))$ | 1

• b = 0 then a
• $\gcd(a,b) = \gcd(b, a\%b)$

Valid for only INTEGERS. $ax + by = \gcd(a,b)$

$x = y_1$
$y = x_1 - (a/b)y_1$

Karatsuba Algorithm $n^{1.585}$ or $n^{\log_2 3}$ | 1

result = $ac \times 10^m + (abcd - ac - bd)10^{m/2} + bd$

Longest sequence of 1s after flip n | 1

Max Product Subarray n | 1

Swap 2 nibbles 1 | 1

Leaders in Array n | n

Lexicographically First Palindrome n | 1

Majority Element n | 1

Boyre-Moore-Solution Majority Vote

Max Equilibrium Sum n | 1

Max Sum - Hour Glass mn | 1 $(R-2) \times (C-2)$ = no. of hour glasses

Selection Sort n^2 | 1

Quick Sort $n^2, n\log n$ | 1

Weighted Substring n^n | 1

Move Hypens to Beginning n | 1

★ Manacher's / Longest Palindromic Substring n | n

Using DP → n^2 | n^2

Maneuvering Problem n | 1

Josephus Problem n | 1

josephus(n, k) = (josephus(n-1, k) + k - 1) % n + 1
and josephus(1, k) = 1

Activity Selection n | 1

★ Permutations $n \times n!$ | n

★ Combinations ^nC_r | 1

Maze $2^{n \times n}$ | $n \times n$

N-queen

Huffman Coding

1

→ Message: ABBCDB CCDAA BBEEE BEAB

Length = 20 letters

Total = 20×8 = 160 bits

A - 65

01000001 — 8 bits

B - 66

01000010

C - 67

D - 68

E - 69

(i) Fixed Length Coding

000

001

010 — A

011 — B

100 — C

101 — D

110 — E

111

2^3 ⇒ 8 diff. combinations from 3 bits

20×3 = 60 bits → Message*

5×3 + 5×8 = 55 bits → Table**

Total = 60 + 55 = 115 bits (Answer)***

"better than 160 bits"

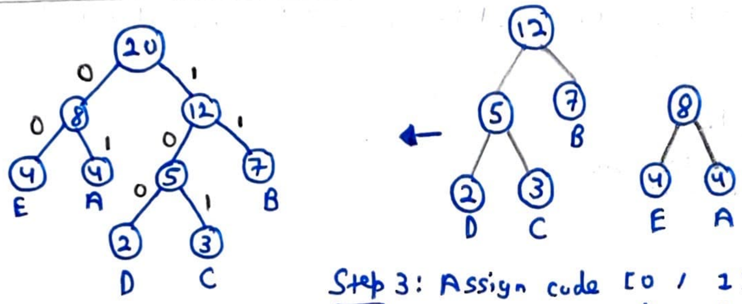
(ii) Variable Length Coding

Char	Frequency /Count	
A	4	01
B	7	11
C	3	101
D	2	100
E	4	00

Step 1: Arrange in ascending order

2	3	4	4	7
~~D~~	~~C~~	~~E~~	~~A~~	B

Step 2: Draw Tree, merge 2 min.



Step 3: Assign code [0 / 1] → left / right

• Traverse root to leaf

Char	Code	
(4) A	10	01
(7) B	11	
(3) C	101	
(2) D	001	100
(4) E	00	

Note: No code is prefix of another code. → Prefix rule

Step 4: Calculations

*Message → 4×2 + 7×2 + 3×3 + 3×2 + 2×4

= 8 + 14 + 9 + 6 + 8 = 45 bits

**Table → 8×5 + 2+2+2+3+3

= 40 + 12 = 52 bits

Total = 45 + 52 = 97 bits (Answer)***

- David Huffman in 1951
- Encoding follows the prefix rule
- Most generated characters will get the small code and least generated will get big code.
- Time complexity - O(n log n)

2

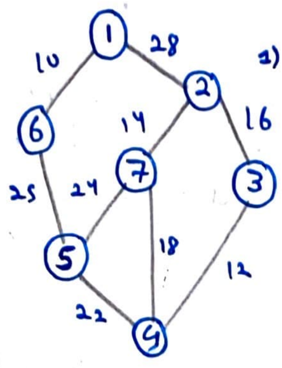
Minimum Cost Spanning Tree
↳ Kruskal's and Prims algo.

$\{ {}^nC_r = n! / r!\,(n-r)! \}$

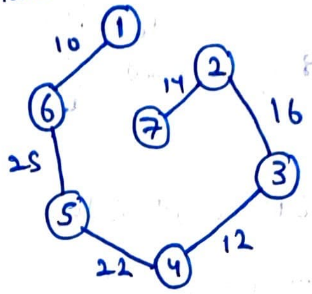
No. of MST formed from a graph with E, V = ${}^{E}C_{V-1}$ - cycles

$|E| = 6 \quad \therefore {}^{6}C_{5} = 6!/5!\,(6-1)! = 6$

*Kruskal
[Always select min cost edge]



1) Ascending order → 10, 12, 14, 16, 18, 22, 24, 25, 28
2) Pick min, make connections, no cycles
3) Draw



Answer ⇒ 10 + 12 + 14 + 16 + 22 + 25 = 99 ***

3

Graph Coloring {Welsh Powell algo} ***

↳ Chromatic number - smallest no. of colors needed to color a graph such that no two adjacent nodes have same colors.

Applications - Map coloring, Sodoku, Time Table, Bipartite graphs.

4 + Hamiltonian Cycle

5 The Knight's Tour Problem (Warnsdorff's Rule)