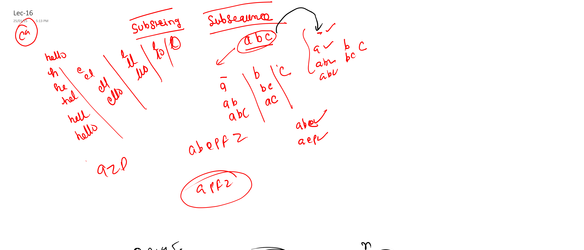
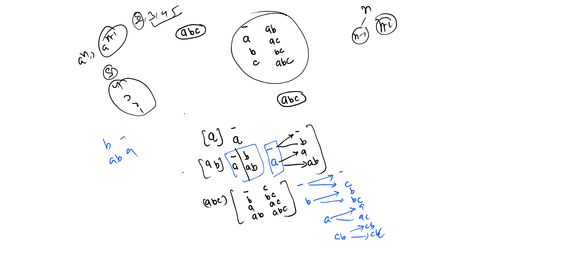
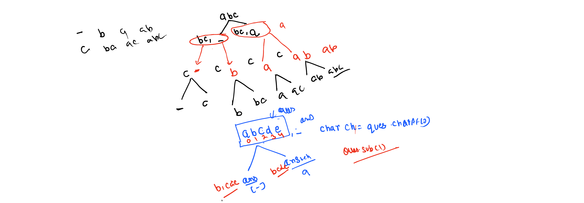
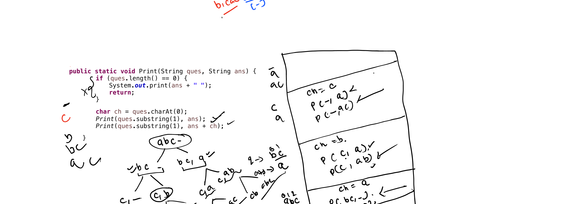
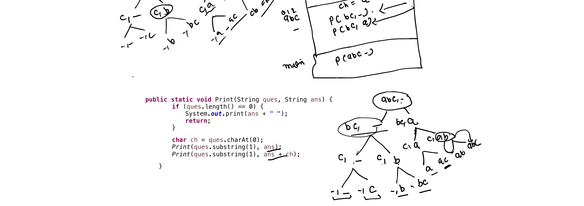

Lec-16
Subsequence
Subsetting
Subsequence
abc
hello
hi
hey
hru
hrd
hell
hello
ell
llo
elo
ello
abppp 2
a20
a1f2


abc
n
n-1

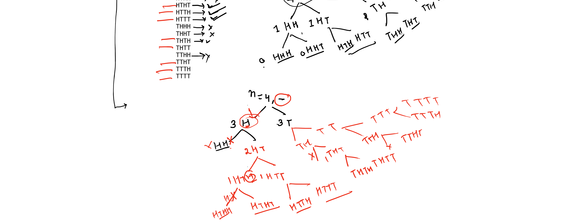
abc
b
ab

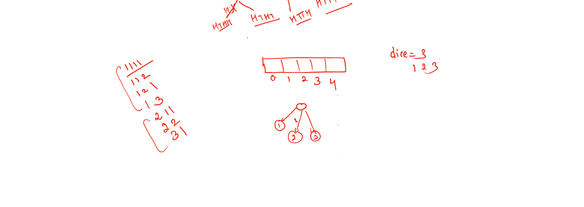

abc
bc
ac
ab
c
b
a

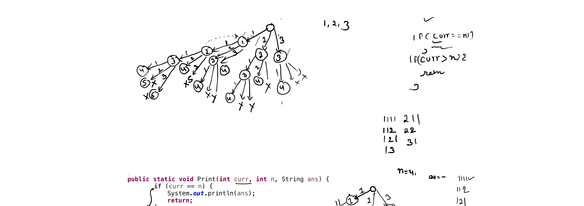

abc
bc
ac
ab
c
b
a

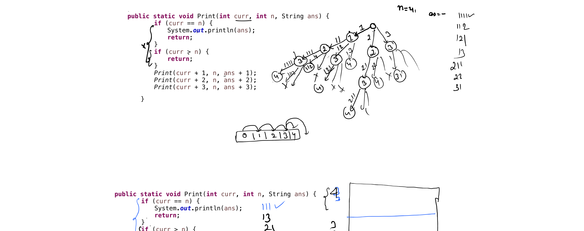

abc
bc
ac
ab
c
b
a
char ch= ques.charAt(0)
ques.substring(1)
base case
return a

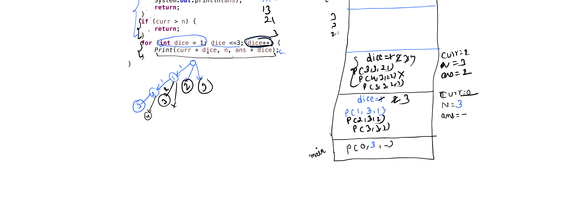
public static void Print(String ques, String ans) {
if (ques.length() == 0) {
System.out.println(ans + " ");
return;
}
char ch = ques.charAt(0);
Print(ques.substring(1), ans);
Print(ques.substring(1), ans + ch);
}
abc
bc
ac
ab
c
a
ch= c
ch= b
ch= a
print

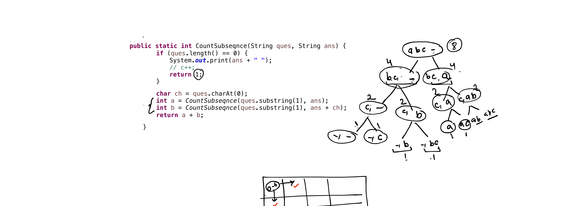

public static void Print(String ques, String ans) {
if (ques.length() == 0) {
System.out.println(ans + " ");
return;
}
char ch = ques.charAt(0);
Print(ques.substring(1), ans);
Print(ques.substring(1), ans + ch);
}
abc
bc
ac
ab
c
a

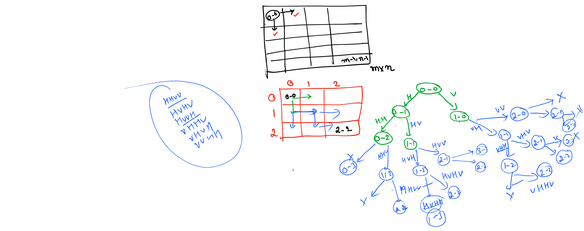

n cars
race
n=3
HHH
HHT
HTH
HTT
THH
THT
TTH
TTT


HHHH
HHHT
HHTH
HHTT
HTHH
HTHT
HTTH
HTTT
THHH
THHT
THTH
THTT
TTHH
TTHT
TTTH
TTTT
N=3, -
2H 1T
1HH 1HT
0 HHH
THH HTH
HHT
TTH
TTT
THT


n=4
3H 1T
2H 2T
1HHH 3T
0 HHHH
TTTT
TTTH
TTHT
THTT
THTH
THHT
HTTT
HTTH
HTHT
HHTT
HHTH
HHHT


dice=3
1 2 3
0 1 2 3 4
1111
112
121
211
22
31


1,2,3
if (curr == n)
if (curr > n)
return
1111 211
112 22
121 31
13
n=4, ans=
public static void Print(int curr, int n, String ans) {
if (curr == n) {
System.out.println(ans);
return;
}
if (curr > n) {
return;
}
Print(curr + 1, n, ans + 1);
Print(curr + 2, n, ans + 2);
Print(curr + 3, n, ans + 3);
}
0 1 1 2 3 4


public static void Print(int curr, int n, String ans) {
if (curr == n) {
System.out.println(ans);
return;
}
if (curr > n) {
return;
}
for (int dice = 1; dice <=3; dice++) {
Print(curr + dice, n, ans + dice);
}
}
111
13
21
3
2
1
dice=1 2 3
curr=2
n=3
ans=1
dice=3
curr=0
n=3
ans=
print(0,3,"")

public static int CountSubseqnce(String ques, String ans) {
if (ques.length() == 0) {
System.out.println(ans + " ");
// c++;
return 1;
}
char ch = ques.charAt(0);
int a = CountSubseqnce(ques.substring(1), ans);
int b = CountSubseqnce(ques.substring(1), ans + ch);
return a + b;
}
abc
bc
ac
ab
c
a
8
4
4
2
2
2
2
1
1
1
1
1
1
1
1

m*n
m*n
0 1 2
0
1
2
HHH
HHT
HTH
HTT
THH
THT
TTH
TTT