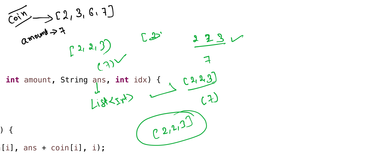
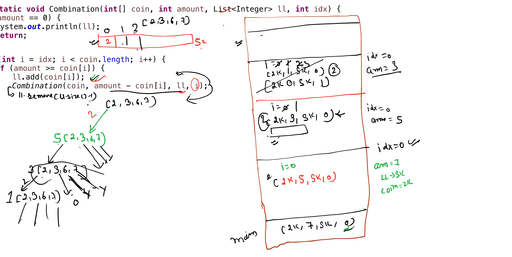
Lec-22

09/06/25 8:27 PM

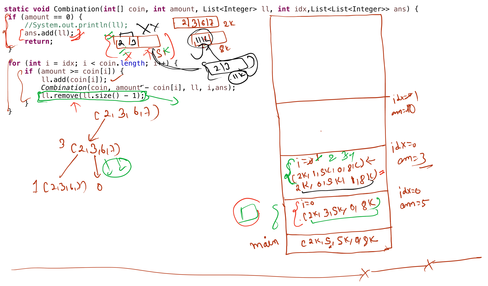
```
public static void main(String[] args) {
        // TODO Auto-generated method stub
        int[] coin = { 2, 3, 6, 7 };
        int amount = 7;
        Combination(coin, amount, "", 0);
    }

    public static void Combination(int[] coin, int amount, String ans, int idx) {
        if (amount == 0) {
            System.out.println(ans);
            return;
        }

        for (int i = idx; i < coin.length; i++) {
            if (amount >= coin[i]) {
                Combination(coin, amount - coin[i], ans + coin[i], i);
            }
        }
    }
```

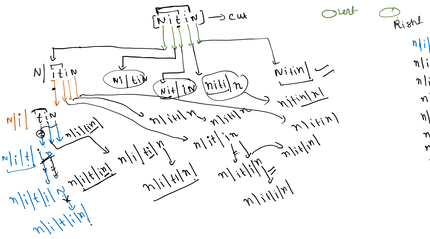


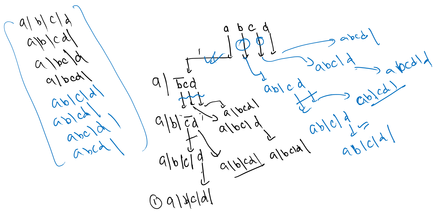

```
public static void main(String[] args) {
        // TODO Auto-generated method stub
        int[] coin = { 2, 3, 6, 7 };
        int amount = 7;
        List<Integer> ll = new ArrayList<>();
        Combination(coin, amount, ll, 0);

    }

    public static void Combination(int[] coin, int amount, List<Integer> ll, int idx) {
        if (amount == 0) {
            System.out.println(ll);
            return;
        }

        for (int i = idx; i < coin.length; i++) {
            if (amount >= coin[i]) {
                ll.add(coin[i]);
                Combination(coin, amount - coin[i], ll, i);
                ll.remove(ll.size() - 1);
            }
        }
    }
```

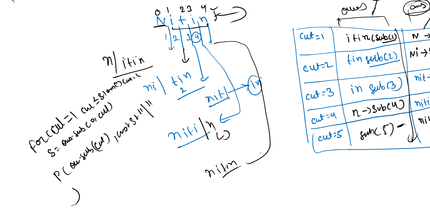


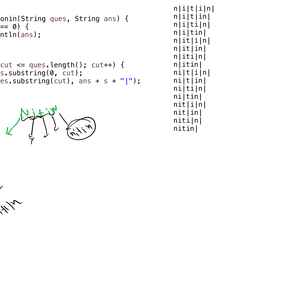

```
public static void Combination(int[] coin, int amount, List<Integer> ll, int idx,List<List<Integer>> ans) {
        if (amount == 0) {
            //System.out.println(ll);
            ans.add(ll);
            return;
        }

        for (int i = idx; i < coin.length; i++) {
            if (amount >= coin[i]) {
                ll.add(coin[i]);
                Combination(coin, amount - coin[i], ll, i,ans);
                ll.remove(ll.size() - 1);
            }
        }
    }
```










```
public static void Partitionin(String ques, String ans) {
        if (ques.length() == 0) {
            System.out.println(ans);
            return;
        }

        for (int cut = 1; cut <= ques.length(); cut++) {
            String s = ques.substring(0, cut);
            Partitionin(ques.substring(cut), ans + s + "|");
        }

    }
```