**Recursion Part 2**

**Q.1) Given a “2\*n” board and tiles size “2\*1”,count the number of ways to tile the given board using the 2\*1 tiles.**

**Note: (A tile can either be placed horizontally or vertically).**

**Answer:**

**2 2**

**N 1**

**You have to find the ways how many ways you can fill that bord**

**You can fill also vertically and horizontal.**

**If check are of square is 2\*n and area of tile is 2cm^2 so n number tiles can fit in the board.**

**Q.2) Remove Duplicate from string**

**Explanation: “appnnacollege”**

**Suppose you have that string and in that some element is repeted means two time so remove that character and generate new string and return it.**

**Ans:**

**Approch of solving**

1. **Creating string builder that append each character**
2. **Take one array that store the iteration of character.**

**Solving appproch with recursion.**

**Base case: If current index is ‘n’(size of main string), print the newString and return it;**

**Kam:**

**Check each element with their map element value if it is true.**

**{ then that element is already in a main string. In this increase the index of size without appending newString value.**

**}**

**Else**

**{ call to function and increase the index size by plus one and append the current element with newString.**

**}**

**Code:**

**Function(String str, int indexOfStr, Stringbuilder newStr, Boolean map[]){**

**If(n==str.length()){**

**System.out.println(newStr);**

**Return;**

**}**

**Char currentCharacter=str.charAt(index);**

**if(Map[currentCharacter-‘a’]==true)**

**{**

**Function(str, index+1, newStr,map[]);**

**}**

**Else{**

**Function(str,index+1,newStr.append[character], map[]);**

**}**

**}**