Advanced Recursion

Steps Revisited:

1.Base Case:

First try to make a base case in order to set up a termination condition.

2. Edge Case (Optional):

This is used to avoid accessing array at the outliers.

3. Solving Smaller Sub-Problem:

Relax, this will be taken care by the recursive call. :)

4. Solving the Final Step:

Recall PMI, we have to work on our final step of calling.

Replace Character Recursively

Given an input string S and two characters c1 and c2, you need to replace every occurrence of character c1 with character c2 in the given string.
Do this recursively.

Input Format :
Line 1 : Input String S
Line 2 : Character c1 and c2 (separated by space)

Output Format :
Updated string

Constraints :
1 <= Length of String S <= 10^6

Sample Input :
abacd
a x
Sample Output :
xbxcd

Return Subsets of an array

Merge Sort

Sort an array A using Merge Sort.

Input format :
Line 1 : Integer n i.e. Array size
Line 2 : Array elements (separated by space)

Output format :
Array elements in increasing order (separated by space)

Constraints :
1 <= n <= 1000

Sample Input:
6
2 6 8 5 4 3
Sample Output:
2 3 4 5 6 8

Quick-Sort

Sort an array A using Quick Sort.

Input format :
Line 1 : Integer n i.e. Array size
Line 2 : Array elements (separated by space)

Output format :
Array elements in increasing order (separated by space)

Constraints :
1 <= n <= 1000

Sample Input:
6
2 6 8 5 4 3
Sample Output:
2 3 4 5 6 8

Return Keypad Code

Given an integer n, using phone keypad find out and print all the possible strings that can be made using digits of input n.
Note : The order of strings are not important. Just print different strings in new lines.

Input Format :
Integer n

Output Format :
All possible strings in different lines

Constraints :
1 <= n <= 10^6

Sample Input:
23
Sample Output:
ad
ae
af
bd
be
bf
cd
ce
cf

Break - Question

Staircase

A child is running up a staircase with N steps, and can hop either 1 step, 2 steps or 3 steps at a time. Implement a method to count how many possible ways the child can run up to the stairs. You need to return number of possible ways W.

Input format :
Line 1 : Integer N (No. of steps)

Output Format :
Line 1 : Integer W i.e. Number of possible ways

Constraint :
(1 <= N <= 30)

Sample Input 1:
4
Sample Output :
7

Introduction to Dynamic Programming

Fibonacci Numbers Revisit

Break Question-Min Steps Revisit

Minimum Count

Given an integer N, find and return the count of minimum numbers, sum of whose squares is equal to N.
That is, if N is 4, then we can represent it as : {1^2 + 1^2 + 1^2 + 1^2} and {2^2}. Output will be 1, as 1 is the minimum count of numbers required.
Note : x^y represents x raise to the power y.

Input Format :
Integer N

Output Format :
Required minimum count

Constraints :
1 <= N <= 1000

Sample Input 1 :
12
Sample Output 1 :
3

Sample Output 1 Explanation :
12 can be represented as :
1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1
1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 1^1 + 2^2
1^1 + 1^1 + 1^1 + 1^1 + 2^2 + 2^2
2^2 + 2^2 + 2^2
As we can see, the output should be 3.

Minimum Cost Path

Given an integer matrix of size m*n, you need to find out the value of minimum cost to reach from the cell (0, 0) to (m-1, n-1).
From a cell (i, j), you can move in three directions : (i+1, j), (i, j+1) and (i+1, j+1).
Cost of a path is defined as the sum of values of each cell through which path passes.

Input Format :
Line 1 : Two integers, m and n
Next m lines : n integers of each row (separated by space)

Output Format :
Minimum cost

Constraints :
1 <= m, n <= 20

Sample Input 1 :
```
3 4
3 4 1 2
2 1 8 9
4 7 8 1
```
Sample Output 1 :
```
13
```

Lowest Common Subsequence

A subsequence is a sequence that can be derived from another sequence by deleting some elements without changing the order of the remaining elements. Longest common subsequence (LCS) of 2 sequences is a subsequence, with maximal length, which is common to both the sequences.

Given two strings,S and T, find the value of longest common subsequence and print it as a line of space-separated integers.

Input Format
The first line contains two space separated integers n and m, the sizes of sequences S and T.
The next line contains string S.
The next line contains string T.

Constraints
1<= n <= 100
1<= m <= 100

Output Format
Print the length of the longest common subsequence as a string on one line.

Sample Input
```
5 6
abcde
cadbce
```

Sample Output
```
4
```

Explanation
The longest common subsequence is “abce” so the length is 4.