In [104]:
```
def Compression(s):
    res = s[0]
    cnt = 1
    for i in range(1, len(s)):
        if s[i] == s[i-1]:
            cnt += 1
        elif cnt > 1:
            res += str(cnt) + s[i]
            cnt = 1
        else:
            res += s[i]
    if cnt > 1:
        res += str(cnt)
    return res
Compression('aaabaaaaccaaaaba')
```

Out[104]:
```
'a3ba4c2a4ba'
```

In [110]:
```
# https://blog.csdn.net/roufoo/article/details/104901535
def GroupingOptions(people, groups):
    if groups > people:
        return 0
    dp = [[0 for _ in range(groups+1)] for _ in range(people+1)]
    for i in range(groups):
        dp[i][i] = 1
    for i in range(2, people+1):
        for j in range(1, min(i, groups)+1):
            dp[i][j] = dp[i-1][j-1] + dp[i-j][j]
    return dp[-1][-1]
GroupingOptions(8, 4)
```

Out[110]:
```
5
```

In [ ]:
```
# lc 1481
def SellingProducts(arr, k):
    if k == len(arr):
        return 0
    lookup = {}
    for a in arr:
        if a in lookup:
            lookup[a] += 1
        else:
            lookup[a] = 1
    cnt = sorted(list(lookup.values()))
    for i in range(len(cnt)):
        if k >= cnt[i]:
            k -= cnt[i]
        else:
            return len(cnt) - i
```

In [111]:
```
def nameDevices(devices):
    res = []
    lookup = {}
    for device in devices:
        if device in lookup:
            res.append(device+str(lookup[device]))
            lookup[device] += 1
        else:
            res.append(device)
            lookup[device] = 1
    return res
devices = ['switch', 'tv', 'switch', 'tv', 'switch', 'tv']
nameDevices(devices)
```

Out[111]:
```
['switch', 'tv', 'switch1', 'tv1', 'switch2', 'tv2']
```

In [112]:
```
def metrolandfestival(num, x, y):
    xx = []
    yy = []
    for i in range(len(num)):
        xx += [x[i]] * num[i]
        yy += [y[i]] * num[i]
    xx.sort()
    yy.sort()
    def caldistance(x1, y1, x2, y2, n):
        return (abs(x1-x2) + abs(y1-y2)) * n
    mx = xx[len(xx)//2]
    my = yy[len(yy)//2]
    res = 0
    for i in range(len(num)):
        res += caldistance(x[i], y[i], mx, my, num[i])
    return res
metrolandfestival([1, 2], [1, 3], [1, 3])
```

Out[112]:
```
4
```

In [ ]:
```
# String Compression
# Selling Products
# Grouping Options
# Metro Land Festival
# Device Name Systems
```

In [ ]:
```
# Java:Collections Vector's methods are synchronized and ArrayList's method
# Java:Synchronization Equivalent
# QuickSort 1324
# Determine the output Compile Error
# What happens when a public data member with the same name is provided in
# A circularly linked list is used to represent a Queue. A single variable
# B. null scurvy
```