## Code Listing

## 1 Problem-1: Anagrams

Created two Dictionaries for both strings with kry as character and value as its frequency. Compared both Dictionaries using '=' operator

```
def main():
    ans = []
    while True:
        x = list(map(int, input().split()))
        if not x:
5
          break
        else:
            ans.append(x)
    print('initial: ',ans)
9
    x = len(ans) - 1
10
    total = 0
11
    while x:
12
      curr_row = 1
13
14
      for i in range(0,curr_row):
        left = i - 1
15
        right = i
16
        if left < 0 or left > curr_row -1:
17
18
                left_ans = 0
        else:
19
20
             left_ans = ans[curr_row - 1][left]
21
        if right > curr_row - 1:
22
23
            right_ans = 0
        else:
24
25
             right_ans = ans[curr_row - 1][right]
        ans[curr_row][i] = ans[curr_row][i] + max(left_ans, right_ans
26
        total = max(total, ans[curr_row][i])
27
      curr_row += 1
28
29
      x = x - 1
      print('Modified: ',ans)
30
    return total
31
32 print(main())
```

Listing 1: Problem-1: Anagrams

## 2 Freedom Struggle Reloaded

```
def getSelectAsMinister():
        Python program to free the country from the dictator.
    Submitted by Manoj Dongare - manojdon777@gmail.com
5
    testcases = int(input())
    for test in range(testcases):
     n, k = map(int, input().split())
9
     position = 1
while n//k > 0:
10
11
12
       n = n//k
       position = position * k
      print(position)
14
15 getSelectAsMinister()
```

Listing 2: Problem-1: Anagrams

## 3 Problem-3: Number Pyramid

```
def main():
    ans = []
    while True:
       x = list(map(int, input().split()))
        if not x:
5
6
         break
        else:
            ans.append(x)
   print('initial: ',ans)
9
10
    x = len(ans) - 1
    total = 0
11
12
    while x:
13
      curr_row = 1
      for i in range(0,curr_row):
14
15
        left = i - 1
        right = i
16
        if left < 0 or left > curr_row -1:
17
               left_ans = 0
18
        else:
19
            left_ans = ans[curr_row - 1][left]
20
21
        if right > curr_row - 1:
22
23
            right_ans = 0
24
25
            right_ans = ans[curr_row - 1][right]
        ans[curr_row][i] = ans[curr_row][i] + max(left_ans, right_ans
26
        total = max(total, ans[curr_row][i])
27
      curr_row += 1
28
      x = x - 1
      print('Modified: ',ans)
30
31
    return total
32 print(main())
```

Listing 3: Problem-1: Anagrams