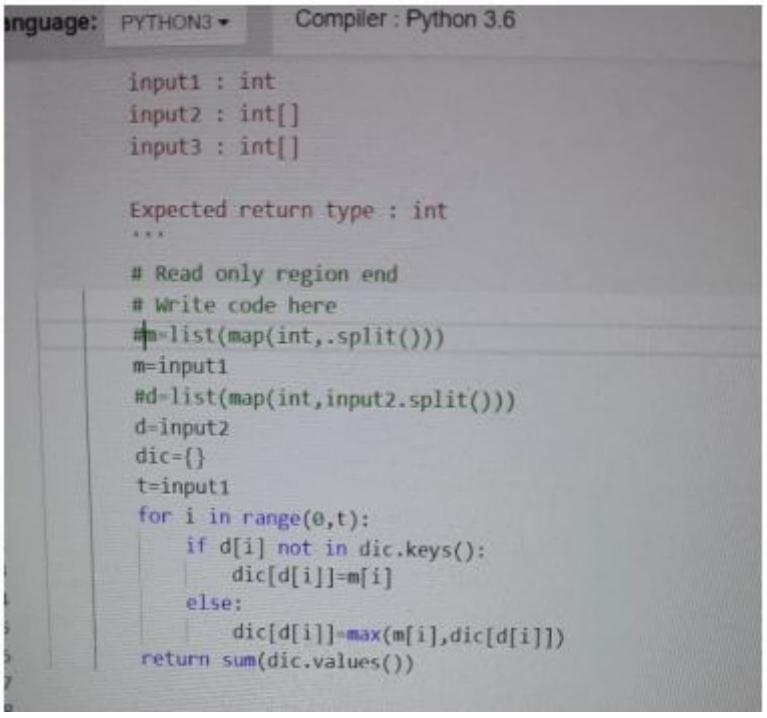
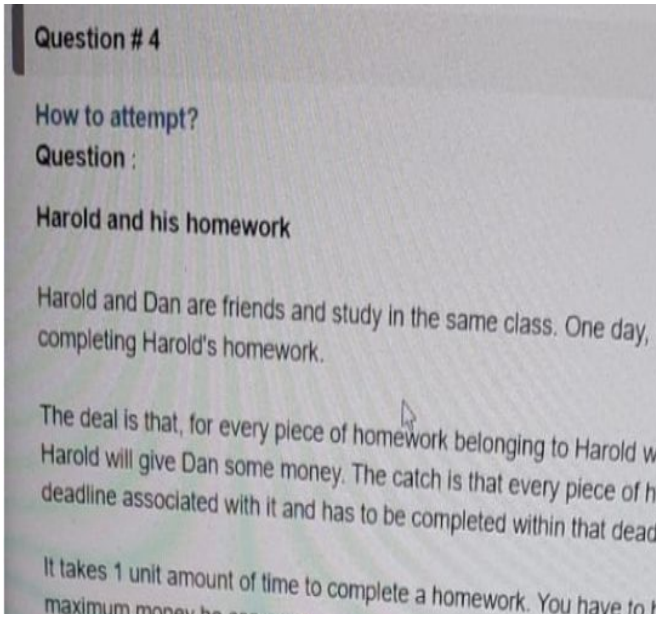
CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

1.Harold and his homework:





Question # 4

How to attempt?

Question :

Harold and his homework

Harold and Dan are friends and study in the same class. One day, completing Harold's homework.

The deal is that, for every piece of homework belonging to Harold w Harold will give Dan some money. The catch is that every piece of h deadline associated with it and has to be completed within that deac

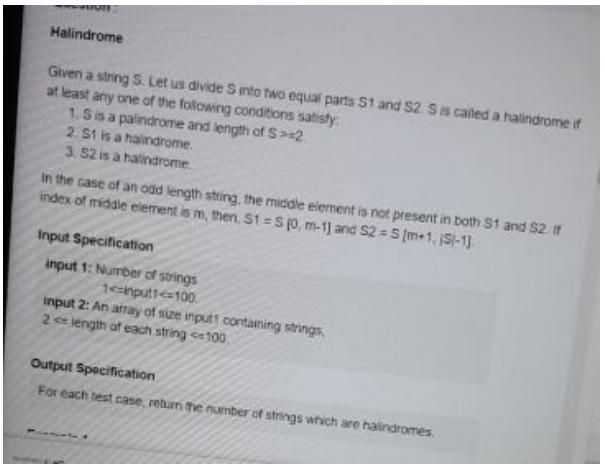
It takes 1 unit amount of time to complete a homework. You have to

NOTE: Here, m=input2
d=input3

2.Halindrome:

(change it as per the question)

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-



Halindrome

Given a string S. Let us divide S into two equal parts S1 and S2. S is called a halindrome if at least any one of the following conditions satisfy:

1. S is a palindrome and length of S >=2.
2. S1 is a halindrome.
3. S2 is a halindrome.

In the case of an odd length string, the middle element is not present in both S1 and S2. If index of middle element is m, then, S1 = S [0, m-1] and S2 = S [m+1, |S|-1].

Input Specification

input 1: Number of strings
1<=input1<=100.

input 2: An array of size input1 containing strings,
2 <= length of each string <=100

Output Specification

For each test case, return the number of strings which are halindromes.

```
def ispal(s):
    rev = ''.join(reversed(s))
    if s == rev:
        return True
    return False

def ishal(s):
    if len(s)<2:
        return False
    if len(s) == 2 and not ispal(s):
        return False
    if len(s)>=2 and ispal(s):
        return True
    else:
        if(len(s)>=4):
            s1 = s[0:len(s)//2]
            s2 = s[(len(s)//2)+1:] if len(s)%2 != 0 else s[len(s)//2:]
            if ishal(s1) or ishal(s2):
                return True

n = int(input())
a = list(input().split(","))
c = 0
for i in range(n):
    if ishal(a[i]):
        c+=1
print(c)
```

Abraham :

```
n=int(input())
c=0;
while n/2>=1:
    c+=1
    n=n/2
if(n==2):
    a=(2*(2**c))-1
else:
    a=(2**c)-1
print(a)

//Abraham in python
```

9:30 am

Planting Trees:

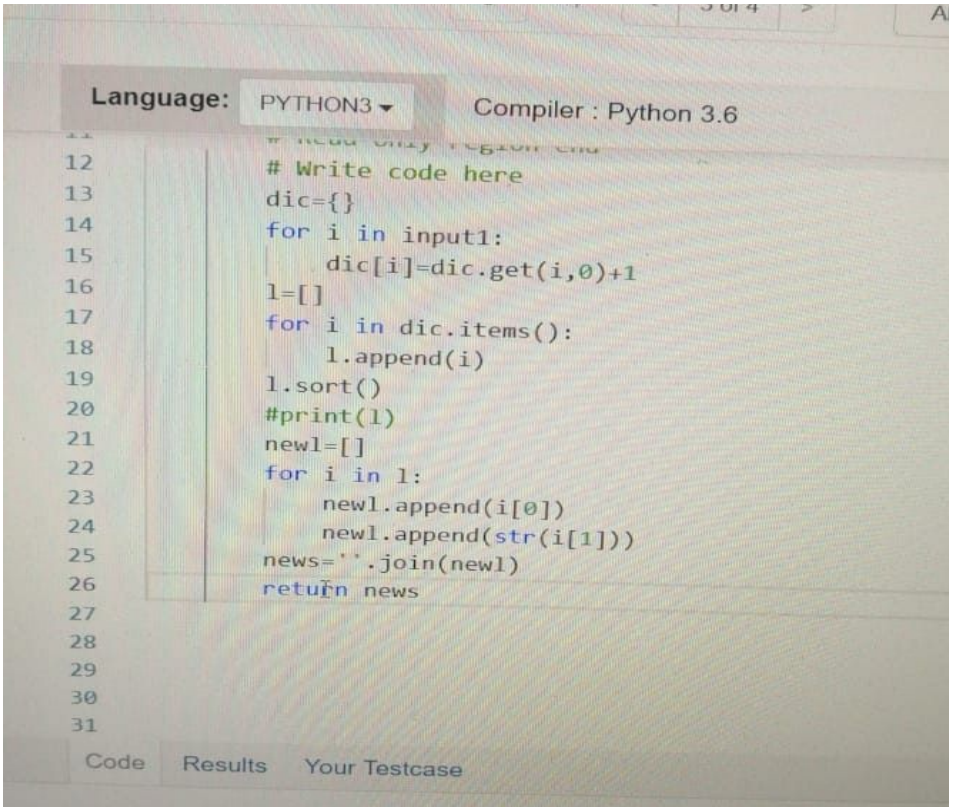
CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

```
n=int(input())
k=int(input())

trees=[2]

for i in range(n):
    trees_planted=list(range(0,(trees.pop(0)+1)%k))
    trees+=trees_planted
#print(trees)
if len(trees)==0:
    print(1)
else:
    print(len(trees))
```

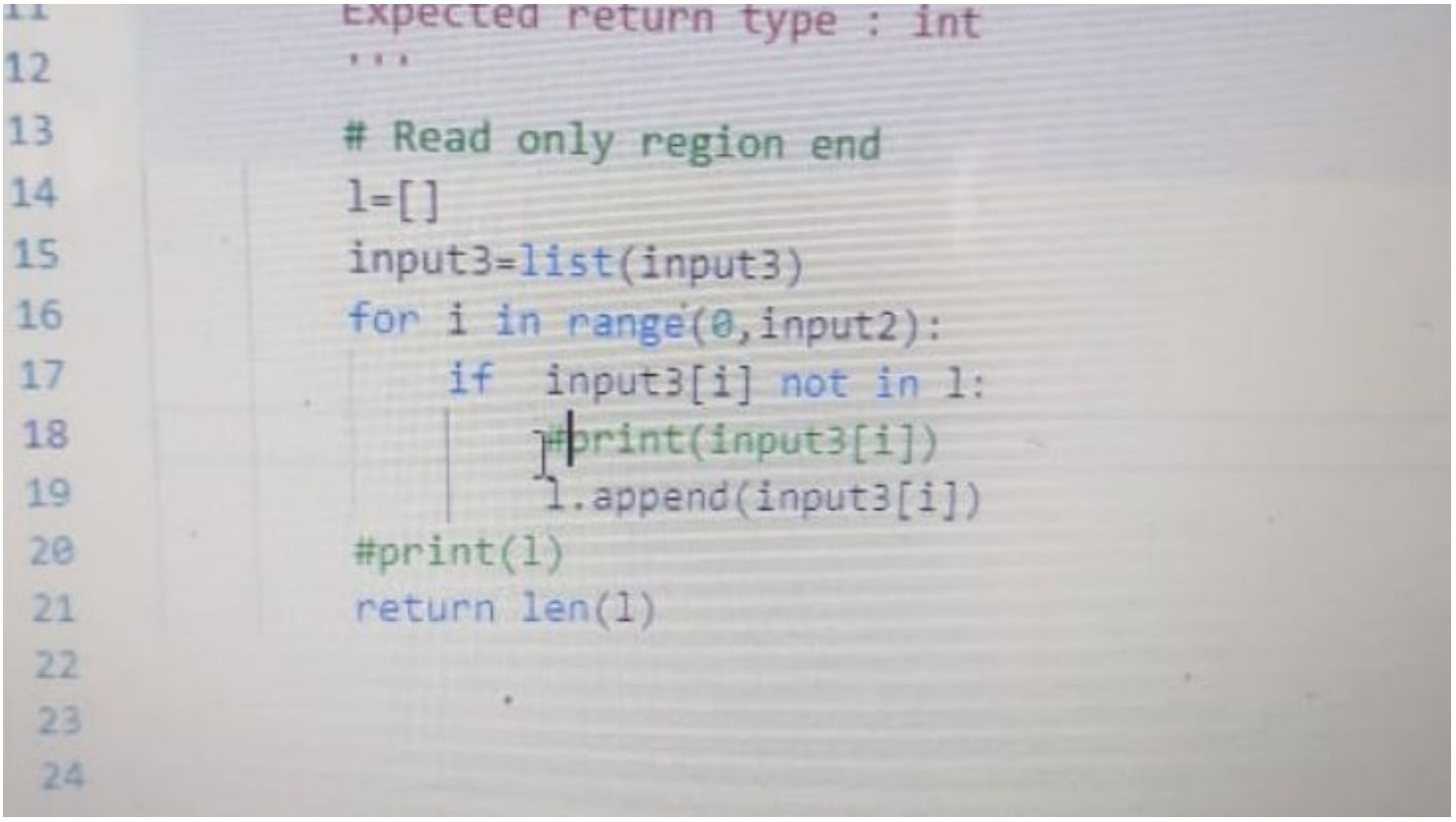
10:14 am

Frequency co:

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-



Monica and flavours some errors:



Social Network

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

```
#SOCIAL NETWORK

print("Enter Range:",end=" ")
n=int(input())
a=[]

a=list(range(n+1))
for i in range(2,int(n**(1/2))+1):
    s=0
    if(i!=-1):
        k=0
        f=1
        while(f):
            s=i*(i+k)
            if(s<n+1):
                a[s]=-1
                k=k+1
            else:
                f=0
c=0
for i in a[2:]:
    if i!=-1:
        c=c+1

```

Shell

```
Python 3.7.6 (bundled)
>>> %Run 'Social Network.py'
  Enter Range: 10
  Total: 3
```

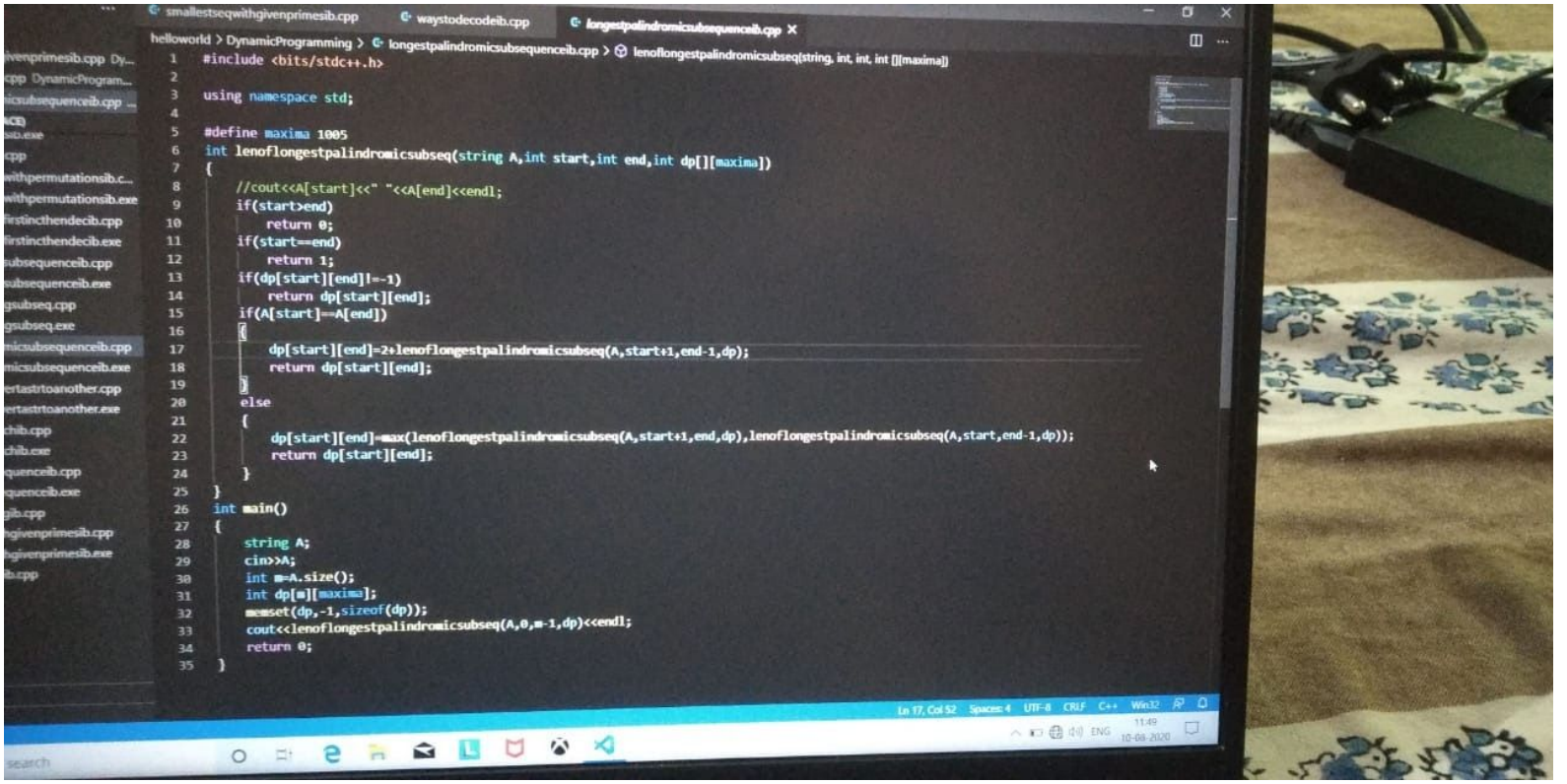
LCS with Vowels

```
str1 = input()
str2 = input ()
```

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

```
vowels = 'aeiou'
c1 = c2 = 0
for ch in str1:
  if ch in vowels:
    c1 += 1
for ch in str2:
  if ch in vowels:
    c2 += 1
print(min(c1,c2))
```

Longest palindromic subsequence



Bob and Numbers

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

```
n = int(input())
a = list(map(int,input().split()))
c = 0
for i in range(n-1):
  for j in range(i+1:n):
    s1 = sum(list(map(int,bin(i)[2:].split())))
    s2 = sum(list(map(int,bin(j)[2:].split())))
    if s1 == s2:
      c += 1
print(c
```

apple orchard solution

Sort the array .
Initialize a max = 0
iterate a loop on sorted array.
suppose sorted arr[] ={48,80,82}
now iterate

```
for (int i = 0 ; i < n ; i++)
{
int temp = arr[i] * (n-i+1);
if (temp > max)
 {
   max = temp;
  }
} // end of for loop
Print(max)
```

New----

```
n = int(input())
l = list(map(int,input().split()))
res = min(l)*n
print(res)
```

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

Evaluate a given infix expression

Use eval() in python

Python >>>

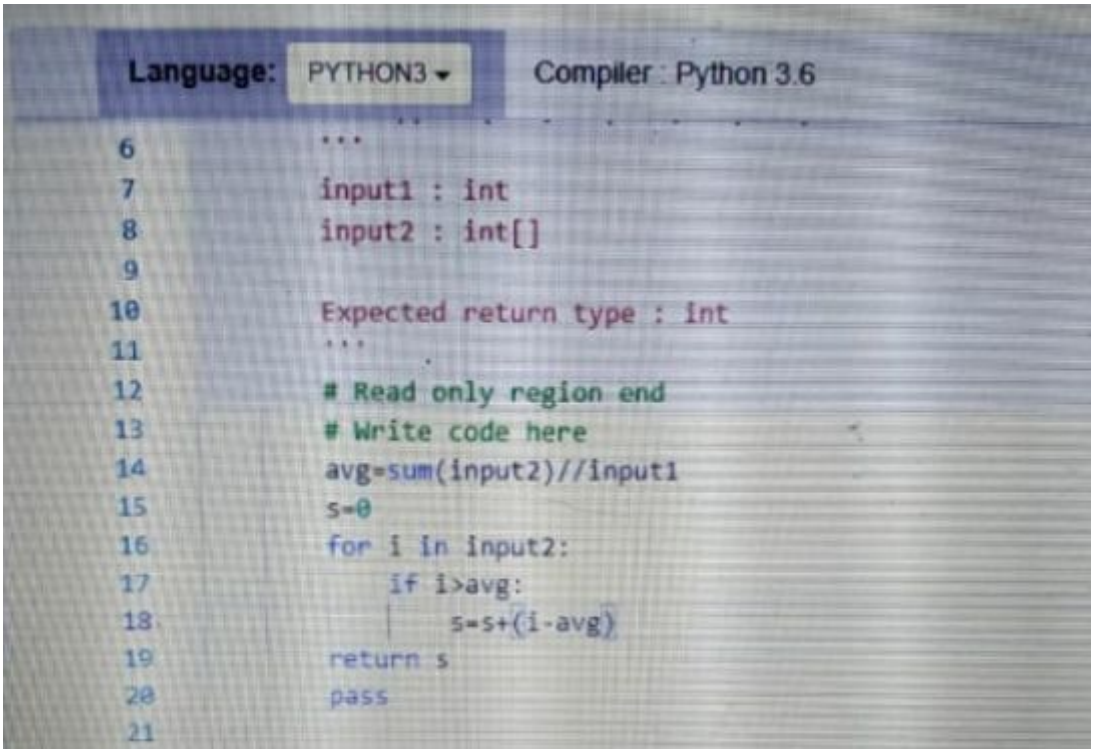
```
>>> eval("2 ** 8")
256
>>> eval("1024 + 1024")
2048
>>> eval("sum([8, 16, 32])")
56
>>> x = 100
>>> eval("x * 2")
200
```

Maximum subarray

```
def maxSubArray(self, nums: List[int]) -> int:
    max_sub_sum = 0
    prev_max = 0
    for num in nums:
        prev_max = max(prev_max + num, num)
        max_sub_sum = max(prev_max, max_sub_sum)

                #if max is 0 and 0 not in list it means that there is no num in subarray, so we choose max num from list
    if max_sub_sum is 0 and 0 not in nums:
        max_sub_sum = max(nums)

    return max_sub_sum
```

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

MOVING APPLES



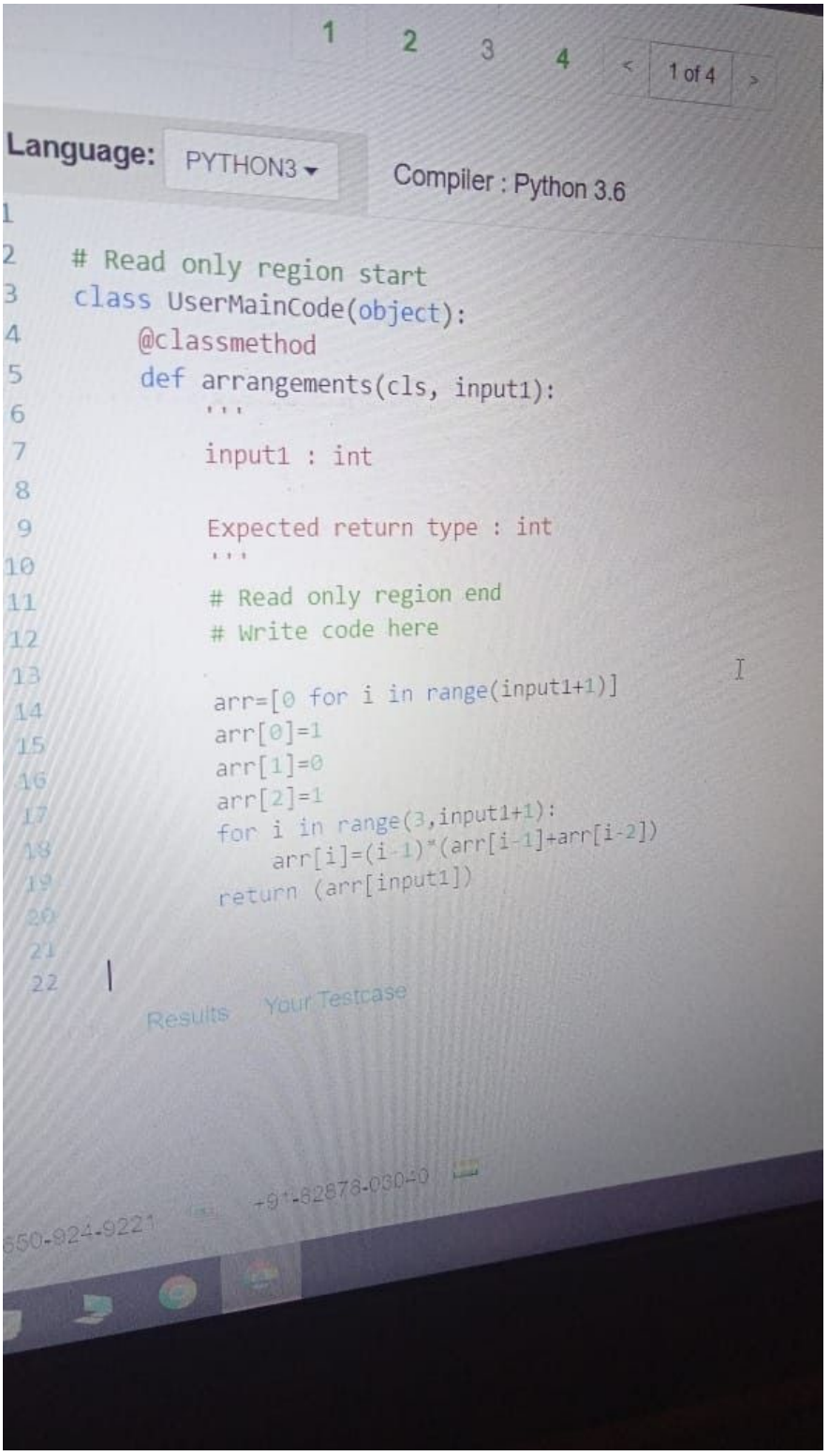
CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

NEXT GENERATOR NUMBER

```
class Solution {
    public int[] nextGreaterElements(int[] nums) {
       int[] output=new int[nums.length];
        int n=nums.length;
        Arrays.fill(output,-1);
        Stack<Integer> stk=new Stack();

          for(int i=0;i<n*2;i++){

            while(!stk.isEmpty() && nums[stk.peek()]<nums[i%n]){

                output[stk.pop()]=nums[i%n];
            }

            if(i<n) stk.push(i);

        }
        return output;

    }
}
```

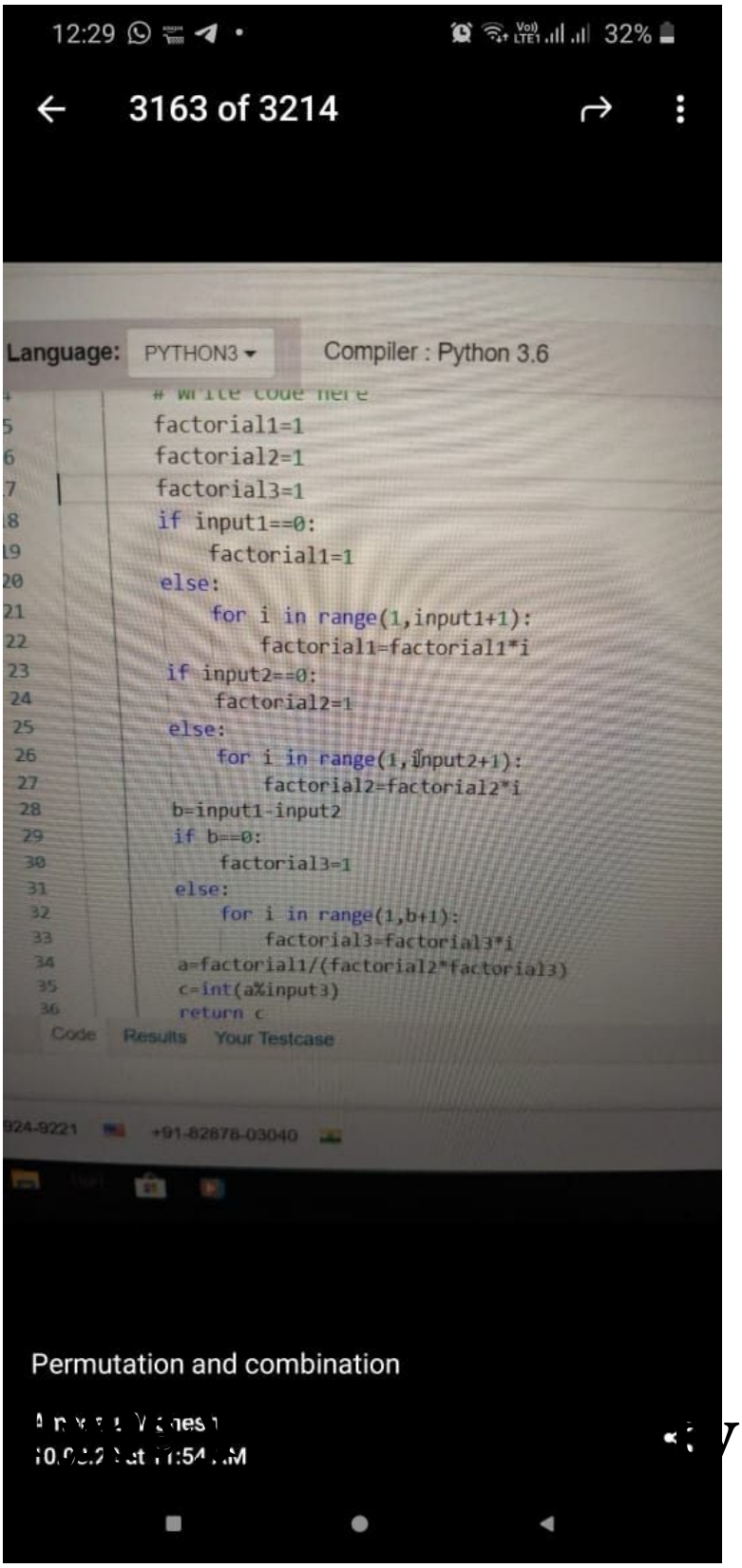
Selective arrangements

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-



CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

Permutation combination



CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

Abraham:

```
n=int(input())

a= {i:i for i in range(n)}
#print(a)
while(len(a)>1):
    l=[]
    for i in range(0,len(a),2):
        l.append(i)
    #print('l=',l)
    for i in l:
        a.pop(i)
        #print(i)
    #print(a)
    counter=0
    d1={}
    for i in a:
        key=i
        val=a[i]
        #a.pop(key)
        d1[counter]=val
        counter+=1
        #print(a)
    a=dict(d1)
    #print(a)
    #input()
print(a[0])
```

CHANGE ALL THE CODES TO AVOID PLAGIARISM :-

Think a number: