

## Revision of Dictionary

```
In [1]: c={'n1':'1234','n2':'3213','n3':'7654','n4':'4521'}
def namec(name,phone):
    if name in c:
        print("Contact %s already exists" % name)
    else:
        c[name] = phone
        print("Contact %s has been added to the list" % name)
namec("n5","9234")
c
```

Contact n5 has been added to the list

```
Out[1]: {'n1': '1234', 'n2': '3213', 'n3': '7654', 'n4': '4521', 'n5': '9234'}
```

```
In [2]: def Searchc(name):
        if name in c:
            print("Contact %s is present" % name)
        else:
            print("Contact %s is not present" % name)
Searchc('n5')
```

Contact n5 is present

```
In [23]: def Modifyc(name,num):
        if name in c:
            c[name] = num
            print("Contact %s modified" % name)
        else:
            print("Contact %s is not present" % name)
Modifyc('n1','12345678')
```

Contact n1 is not present

```
n2 : 3213
n3 : 7654
n4 : 4521
n5 : 9234
e : 3343
f : 4234
g : 4256
```

```
In [4]: def Removec(name):  
        if name in c:  
            c.pop(name)  
            print('Contact %s has ben removed' % name)  
        else:  
            print('Contact %s is not present' % name)  
        Removec('n1')  
        c
```

Contact n1 has ben removed

```
Out[4]: {'n2': '3213', 'n3': '7654', 'n4': '4521', 'n5': '9234'}
```

```
In [5]: def updatec(newc):  
        c.update(newc)  
        return c  
        newc = {"e": "3343", "f": "4234", "g": "4256"}  
        updatec(newc)
```

```
Out[5]: {'n2': '3213',  
        'n3': '7654',  
        'n4': '4521',  
        'n5': '9234',  
        'e': '3343',  
        'f': '4234',  
        'g': '4256'}
```

```
In [24]: for key,value in c.items():  
        print(key,":",value)
```

```
n2 : 3213  
n3 : 7654  
n4 : 4521  
n5 : 9234  
e : 3343  
f : 4234  
g : 4256
```

```
In [ ]:
```

## Revision of packages and modules

```
In [21]: from Packages import eo  
        eo.evenodd(23)
```

Number 23 is odd.

```
In [22]: from Packages.eo import evenodd  
        evenodd(66)
```

Number 66 is even.

## Bricks Game

Patlu and Motu works in a building construction, they have to put some number of bricks  $N$  from one place to another, and started doing their work. They decided , they end up with a fun challenge who will put the last brick.

They to follow a simple rule, In the  $i$ 'th round, Patlu puts  $i$  bricks whereas Motu puts  $i \times 2$  bricks.

There are only  $N$  bricks, you need to help find the challenge result to find who put the last brick.

- Input:
  - First line contains an integer  $N$ .
- Output:
  - Output "Patlu" (without the quotes) if Patlu puts the last bricks , "Motu"(without the quotes) otherwise.

```
In [25]: N = int(input())
def Brick(N):
    for i in range(1,N+1):
        M = i
        P = i*2
        if M == N:
            print("Motu")
        elif P == N:
            print("Patlu")
Brick(N)
```

```
13
Motu
```

## Ali and Helping innocent people

Arpasland has surrounded by attackers. A truck enters the city. The driver claims the load is food and medicine from Iranians. Ali is one of the soldiers in Arpasland. He doubts about the truck, maybe it's from the siege. He knows that a tag is valid if the sum of every two consecutive digits of it is even and its letter is not a vowel. Determine if the tag of the truck is valid or not.

We consider the letters "A","E","I","O","U","Y" to be vowels for this problem.

- Input Format
  - The first line contains a string of length 9. The format is "DDXDDD-DD", where D stands for a digit (non zero) and X is an uppercase english letter.
- Output Format
  - Print "valid" (without quotes) if the tag is valid, print "invalid" otherwise (without quotes)
- Sample Input
  - 12X345-67
- Sample output
  - invalid

```
In [31]: def Ali(N):
    N.split('-')
    ''.join(N[0:])
    c=0
    for i in N:
        if (i == 'A' or i == 'E' or i == 'I' or i == 'O' or i == 'U') and N[i]+N[i-1] == 'AI' or N[i]+N[i-1] == 'EO' or N[i]+N[i-1] == 'OU':
            c=0
            # print("valid")
        else:
            c=1
            #print("invalid")
    if c == 0:
        print("valid")
    else:
        print("invalid")
Ali("12X345-67")
```

invalid

## Roy and Profile Picture

Roy wants to change his profile picture on Facebook. Now Facebook has some restriction over the dimension of picture that we can upload. Minimum dimension of the picture can be  $L \times L$ , where  $L$  is the length of the side of square.

Now Roy has  $N$  photos of various dimensions. Dimension of a photo is denoted as  $W \times H$  where  $W$  - width of the photo and  $H$  - Height of the photo

When any photo is uploaded following events may occur:

[1] If any of the width or height is less than  $L$ , user is prompted to upload another one. Print "UPLOAD ANOTHER" in this case. [2] If width and height, both are large enough and

- (a) if the photo is already square then it is accepted. Print "ACCEPTED" in this case.
- (b) else user is prompted to crop it. Print "CROP IT" in this case.

(quotes are only for clarification)

Given  $L$ ,  $N$ ,  $W$  and  $H$  as input, print appropriate text as output.

- Input:
  - First line contains  $L$ .
  - Second line contains  $N$ , number of photos.
  - Following  $N$  lines each contains two space separated integers  $W$  and  $H$ .
- Output:
  - Print appropriate text for each photo in a new line.
- Sample Input
  - 180
  - 3
  - 640 480
  - 120 300

- 180 180
- Sample Output
  - CROP IT
  - UPLOAD ANOTHER
  - ACCEPTED

```
In [33]: L = int(input()) # 180
N = int(input())
def Profile(N):
    for i in range(1,N+1):
        S = input()
        S = S.split()
        W = int(S[0]) # 640
        H = int(S[1]) # 480

        if(W<L or H<L):
            print("UPLOAD ANOTHER")
        elif W==H:
            print("ACCEPTED")
        else:
            print("CROP IT")

Profile(N)
```

180

3

640 480

CROP IT

120 300

UPLOAD ANOTHER

180 180

ACCEPTED

## Read a given string, change the character at a given index and then print the modified string.

**Input Format** The first line contains a string, . The next line contains an integer , denoting the index location and a character separated by a space.

**Output Format** Using any of the methods explained above, replace the character at index with character .

Sample:

Input:

string="Hello apssdc"

Output:

Hello Apssdc

```
In [46]: N = input()
S = input()
S = S.split()
a = int(S[0])
b = S[1]
print(N.replace(N[a],b,a))
```

```
Aman
1 'M'
A'M'an
```

```
In [ ]:
```