

Problem Statements

Rahul is a very busy person he don't want to waste his time . He keeps account of duration of each and every work. Now he don't even get time to calculate duration of works, So your job is to count the durations for each work and give it to rahul.

Input:

First line will be given by N number of works Next N line will be given SH,SM,EH and EM each separated by space(SH=starting hr, SM=starting min, EH=ending hr, EM=ending min)

Output: N lines with duration HH MM(hours and minutes separated by space)

- SAMPLE INPUT
 - 2
 - 1 44 2 14
 - 2 42 8 23
 - SAMPLE OUTPUT
 - 0 30
 - 5 41

```
In [1]: 1 def duration(sh,sm,eh,em):
2         a = (sh*60+sm)
3         b = (eh*60+em)
4         c = b - a
5         wh = c//60
6         wm = c%60
7         print(wh,end=" ")
8         print(wm)
9         print()
10
11
12 n=int(input())
13 for i in range(n):
14     s=input()
15     s=s.split()
16     sh=int(s[0])
17     sm=int(s[1])
18     eh=int(s[2])
19     em=int(s[3])
20     duration(sh,sm,eh,em)
```

```
2
1 44 2 14
0 30

2 42 8 23
5 41
```

```

In [18]: 1 # Function to Toggle String characters
          2
          3 s = "Abc"
          4
          5 s.isupper() # False --> Not all character are capital
          6 s.islower() # False
          7
          8 s.lower() # Convert a character to Lowercase character
          9 s.upper() # Convert a character to uppercase character
         10
         11 s.split() # output --> ['Abc']
         12 list(s) # output --> ['A', 'b', 'c']
         13 #t = []
         14
         15 s = list(s) # output --> ['A', 'b', 'c']
         16 "".join(s) # output --> 'Abc'
         17

```

Out[18]: 'Abc'

```

In [19]: 1 # Function to Toggle String characters
          2
          3 def toggleString(s):
          4     # Convert the string into a list of characters
          5     s = list(s)
          6     t = []
          7     for c in s:
          8         if c.islower():
          9             t.append(c.upper())
         10         else:
         11             t.append(c.lower())
         12     return "".join(t)
         13
         14 toggleString("Abc")
         15

```

Out[19]: 'aBC'

```

In [21]: 1 ord('a')

```

Out[21]: 97

```

In [55]: 1 # Char sum
          2 s= input()
          3 sums = 0
          4
          5 for c in range(0,len(s)):
          6     a=ord(s[c])
          7     e = a - 96
          8     sums = sums+e
          9 print(sums)

```

abc
6

```
In [47]: 1 s="abs"
         2 s[0]
```

Out[47]: 'a'

```
In [2]: 1 str=input()
        2 sum = 0
        3 for i in str:
        4     a = ord(i) - 96
        5     sum = sum + a
        6 print(sum)
```

jdk
25

```
In [4]: 1 def highrem(n):
        2     r=0
        3     for i in range(1,n+1):
        4         rem=n%i
        5         if rem>r:
        6             r = rem
        7             j = i
        8     return j
        9
       10
       11
       12 t=int(input())
       13 for i in range(t):
       14     n=int(input())
       15     print(highrem(n))
```

2
4
3
5
3

```
In [10]: 1 def isPrime(n):
        2     count = 0
        3     for i in range(1,n+1):
        4         if(n%i==0):
        5             count+=1
        6     if(count==2):
        7         print(n, "is prime")
        8     else:
        9         print("not prime")
       10
       11 n=int(input())
       12 isPrime(n)
```

7
7 is prime

```
In [21]: 1 def PrimeRange(n):
2
3     for i in range(2,n+1):
4         count=0
5         for j in range(1,i+1):
6             if(i%j==0):
7                 count=count+1
8         if(count==2):
9             print(i,end=" ")
10
11 n=int(input())
12 PrimeRange(n)
13
```

29

2 3 5 7 11 13 17 19 23 29

```
In [ ]: 1
2
3
4
5
```

```
In [ ]: 1
```