

```
In [1]: #Program:TASK 1
#Program By:Ayush Pandey
#Email Id:1805290@kiit.ac.in
#DATE:17-Sept-2021
#Python Version:3.7
#CAVEATS:None
#LICENSE:None
```

```
In [2]: #To add two numbers without carry forward

import math
def check(n, m) :
    ans= 0
    mul= 1
    summ= 0
    while (n or m) :
        #Adding each bits from the numbers
        see=((n%10)+(m%10))
        #Neglecting the carry from see
        see=see%10
        #Updating my ans
        ans=(see*mul)+ans
        n=math.floor(n/10)
        m=math.floor(m/10)
        mul= mul* 10
    #Returning my ans
    return ans
n,m=map(int,input().split(" "))
print (check(n,m))

56 34
80
```

```
In [3]: #To get the comman factors in the list of elements

def check(a):
    #Getting the minimum element in the list
    minn=min(a)
    ans=[]
    #Iterating from 1 to minimum element present in the list
    #As the comman factors of the list cannot be greater than the smallest number of the list
    for i in range(1,minn+1,1):
        flag=0
        for j in range(len(a)):
            #Checking wheteher all elements are divisible by i or not
            if(a[j]%i!=0):
                flag=1
                break
        #Flag=0 means all element present in list are divisible by i
        if(flag==0):
            ans.append(i)
    return ans
a=list(map(int,input().split(" ")))
print(check(a))

24 36 10 99 126 256
[1]
```

```
In [4]: #To form the maximum and minimum number using the number given
# 1st approach using permutations

from itertools import permutations
def check(n):
    #Taken minn and maxx to store minimum and maximum value respectively
    minn=-999999999
    maxx=-999999999
    perm=permutations(n)
    #Iterating over each and every permutation possible
    for i in list(perm):
        i="".join(i)
        see=int(i)
        #Checking whether the number formed is greater than maxx or not
        if(see>maxx):
            maxx=see
        #Checking whether the number formed is lesser than minn or not
        if(see<minn):
            minn=see
    print("Maximum is:",maxx)
    print("Minimum is:",minn)
n=input()
ob=check(n)

4562
Maximum is: 6542
Minimum is: 2456
```

```
In [5]: #To form the maximum and minimum number using the number given
#2nd approach using inbuilt sort technique

a=list(input())
#Sorting the list in ascending order
a.sort()
s=a.copy()
#After sorting joining the list
a="".join(a)
#Getting the list in decending order
s=a[::-1]
#Joining the list
s="".join(s)
print("Minimum is:",a)
print("Maximum is:",s)

4562
Minimum is: 2456
Maximum is: 6542
```

```
In [6]: #To form the maximum and minimum number using the number given
# 3rd approach using selection sort method

#Taking the input
a=list(input())
for i in range(len(a)):
    a[i]=int(a[i])
#Iterating to each and every element
for i in range(0,len(a),1):
    for j in range(i+1,len(a),1):
        #Comparing it for sorting in ascending order
        if(a[i]>a[j]):
            a[i],a[j]=a[j],a[i]
for i in range(0,len(a),1):
    a[i]=str(a[i])
s=a.copy()
#For forming the maximum number that can be formed
s=s[::-1]
print("Minimum is:","".join(a))
print("Maximum is:","".join(s))

56234
Minium is: 23456
Maximum is: 65432
```

```
In [7]: #To take a number and arrange then in ascending and decending order and substract the number
#Repeat the process till you get 6174
#Program to tell whether a number eneterd can be converted to 6174 or not

maxx=0
minn=0
#User input value
a=list(input())
flag=0
#Created a see list to store all the number which I will be getting after subtracting from maximum and minimum
see=[]
#As it is a list so joining it so that the list can contains only numbers
see.append(int("".join(a)))
while(flag!=1):
    #If length of the list is less than 4 then it is not possible to form 6174 so it will come out of the loop
    if(len(a)<4):
        break
    #Sorting the list in ascending order
    a.sort()
    #As it is a list so joining it
    minn="".join(a)
    #Converting the type to int so that I can perform operation on it
    minn=int(minn)
    #Getting the elements in decending order
    a=a[::-1]
    #As it is a list so joining it
    maxx="".join(a)
    #Converting the type to int so that I can perform operation on it
    maxx=int(maxx)
    a=maxx-minn
    if(a==6174):
        flag=1
        break
    #If a number is already present in the see list this means our process will be getting repeated that is we are getting the
    #number which is already present and after subtracting from the maximum and minimum we will be getting similar result
    if(a in see):
        break
    else:
        #Appending the number in the see list
        see.append(a)
    a=str(a)
    a=list(a)
#If flag is 0 means that the given number can be transformed to 6174
if(flag==0):
    print("No")
else:
    print("Yes")

1324
Yes
```

```
In [8]: #Improved program for non-repeating element

#To take a number and arrange then in ascending and decending order and substract the number
#Repeat the process till you get 6174
#Program to tell whether a number eneterd can be converted to 6174 or not

maxx=0
minn=0
#User input value
a=list(input())
flag=0
check=[]
#To remove the duplicate present in the list
for i in a:
    if(i not in check):
        check.append(i)

a=[]
a=check.copy()
#Created a see list to store all the number which I will be getting after subtracting from maximum and minimum
see=[]
#As it is a list so joining it so that the list can contains only numbers
see.append(int("".join(a)))
while(flag!=1):
    #If length of the list is less than 4 then it is not possible to form 6174 so it will come out of the loop
    if(len(a)<4):
        break
    #Sorting the list in ascending order
    a.sort()
    #As it is a list so joining it
    minn="".join(a)
    #Converting the type to int so that I can perform operation on it
    minn=int(minn)
    #Getting the elements in decending order
    a=a[::-1]
    #As it is a list so joining it
    maxx="".join(a)
    #Converting the type to int so that I can perform operation on it
    maxx=int(maxx)
    #print(maxx,minn)
    a=maxx-minn
    if(a==6174):
        flag=1
        break
    #If a number is already present in the see list this means our process will be getting repeated that is we are getting the
    #number which is already present and after subtracting from the maximum and minimum we will be getting similar result
    if(a in see):
        break
    else:
        #Appending the number in the see list
        see.append(a)
    a=str(a)
    a=list(a)
#If flag is 0 means that the given number can be transformed to 6174
if(flag==0):
    print("No")
else:
    print("Yes")

5332
No
```