## Programs for Lab session on Nov 3rd 2020:

gcd1.py

```
#Let us define a function by name gcd1
# The function gcd1 determines the greatest common divisior of two numbers

#Algorithm of gcd1(m,n)

#Step1:  Pass m, n as arguments to gcd1

#step2:  set the list of factors  factorsOf_m to an empty list

#Step3:  set the list of factors  factorsOf_n to an empty list

#Step4:  set the list of common factors of m and n factorsOf_mn to empty

#Step5: for each factor of m
#         append factor to factorsOf_m

#Step6: for each factor of n
#         append factor to factorsOf_n

#Step7: for each factor present in factorsOf_m
#         if factor is also presnet in factorsOf_n
#                 append factor to factorsOf_mn

#Step8:  determine the maximum factor in factorsOf_mn and return as the gcd
```

gcd_source_1.py

```
#Let us define a function by name gcd1

# The function gcd1 determines the greatest common divisior of two numbers


#Algorithm of gcd1(m,n)

def gcd1(m,n):


#Step1:  Pass m, n as arguments to gcd1


#step2:  set the list of factors  factorsOf_m to an empty list

    factorsOf_m = []


#Step3:  set the list of factors  factorsOf_n to an empty list

    factorsOf_n = []


#Step4:  set the list of common factors of m and n factorsOf_mn to empty

    factorsOf_mn = []




#Step5: for each factor of m

#          append factor to factorsOf_m

    for i in range(1,m+1):

        if (m%i == 0):

            factorsOf_m.append(i)




#Step6: for each factor of n

#          append factor to factorsOf_n
```

```
    for i in range(1,n+1):
        if (n%i == 0):
            factorsOf_n.append(i)

#Step7: for each factor present in factorsOf_m
#          if factor is also presnet in factorsOf_n
#                    append factor to factorsOf_mn
    for i in factorsOf_m:
        if i in factorsOf_n:
            factorsOf_mn.append(i)


#Step8:  determine the maximum factor in factorsOf_mn and return as the gcd
    return(factorsOf_mn[-1])
```

## gcd-source2.py

```
#Let us define a function by name gcd2
# The function gcd2 determines the greatest common divisior of two numbers


#Algorithm of gcd2(m,n)


#Step1:  Pass m, n as arguments to gcd1
def gcd2(m,n):


#Step3:  set the list of common factors of m and n factorsOf_mn to an empty list
    factorsOf_mn = []


#Step4: for i =1 to min(m,n)
#        if i is a factor of both m and n
#              append i to factorsOf_mn
    for i in range(1,min(m,n)+1):
#       FILL THE MIssing STATEMENT(s)
        if ((m%i) == 0 and (n%i)== 0):
            factorsOf_mn.append(i)



#Step5:  determine the maximum factor in factorsOf_mn and return as the gcd
    return(factorsOf_mn[-1])
```

**gcd3-source.py**

```
#15, 10
#10, (15-10)
# gcd(10,15) = gcd(5,10)
#18,12
#12, (18-12)
#12, 6
#gcd(18,12) = gcd(6,12)
#gcd(95,90) = gcd(90,(95-90)) = gcd(90,5)=(gcd(
# gcd(6,5)=gcd(5,1)=gcd((5-1),1)=(gcd(4-1),1)=(gcd(3-1),1)=(gcd(2-1),1)=(gcd(1,1)
def gcd3(m,n):
    if (m<n):
        (m,n)=(n,m)
    if (m==n):
        return m
    else:
        return(gcd3((m-n),n))
```