#pattternStar.py

for i in range(4):

for j in range(i+1):

print("\*", end=' ')

print("")

#patternNumbers.py

for i in range(4,0,-1):

for j in range(i):

print(j+1, end=' ')

print("")

#patternHash.py

for i in range(4,0,-1):

for j in range(i):

print("#", end=' ')

print("")

#lambdaAndReduce.py

from functools import reduce

str1 =input("Enter the elements separated by a space: ")

list1 = list(str1.split(' '))

result= reduce(lambda x,y: int(x)+int(y), list1)

print(result)

#lambdaAndMap.py

#to multiply all elements in a list by a number using map and lambda function

str1 =input("Enter the elements of an array separated by space: ")

my\_list = list(str1.split(" "))

print(my\_list)

i =int(input("Enter the number to be multiplied: "))

#result=[int(num)\*2 for num in my\_list]

def mul(list1,iter1):

return (list(map(lambda num: int(num)\*iter1, list1)))

result=mul(my\_list,i)

print(result)

#FibonacciGenerator.py

#print the list of fibonacci series using generators

no = int(input("Enter the numberof items in fibonacci series: "))

def fibon(max\_items):

f,s =0,1

for \_ in range(max\_items):

yield f

f,s = s,f+s

for i in fibon(no):

print(i,end="\t")

#FibonacciGenerator2.py

#to generate fibonacci series until a particular number using Generators

max\_number = int(input("Enter the maximum number in the series: "))

def fib():

f,s=0,1

while True:

yield f

f,s=s,f+s

out = fib()

for i in out:

if i> max\_number:

break

print(i,end="\t")