Name: SK Shivaanee

Roll no: 2310257

Section: S02 **Assignment 5: Problem-solving using User-defined Functions**

Aim: To learn python programming using looping statements by forming expressions and statements involving reading and printing the data appropriately for the given specification

Solve the following problems using Python (CO1, K3):

Question 1: In a shopping mall, there are three theatres (‘BigPix’, ‘IMAX’, ‘4DX’), and the ticket reservation system provides functionality to check the availability of tickets and book tickets for the customers. The number of available seats in each of the theatres is maintained as global variables T1, T2 and T3. These variables are initialized to a random number between 0 and the maximum capacity of the corresponding theatre.

Write a function, Check_Ticket(no_tickets, theatre_id), and return a boolean value indicating the ticket availability. If the ticket is not available, make a call to the function Check_Alternate to return an alternate theater_id by taking in no_tickets as its input. If the alternate theater_id is empty, then display "Ticket not available"; otherwise, call Book_Ticket(no_tickets, theatre_id) for ticket reservation and also display the updated seat availability of the corresponding theater.

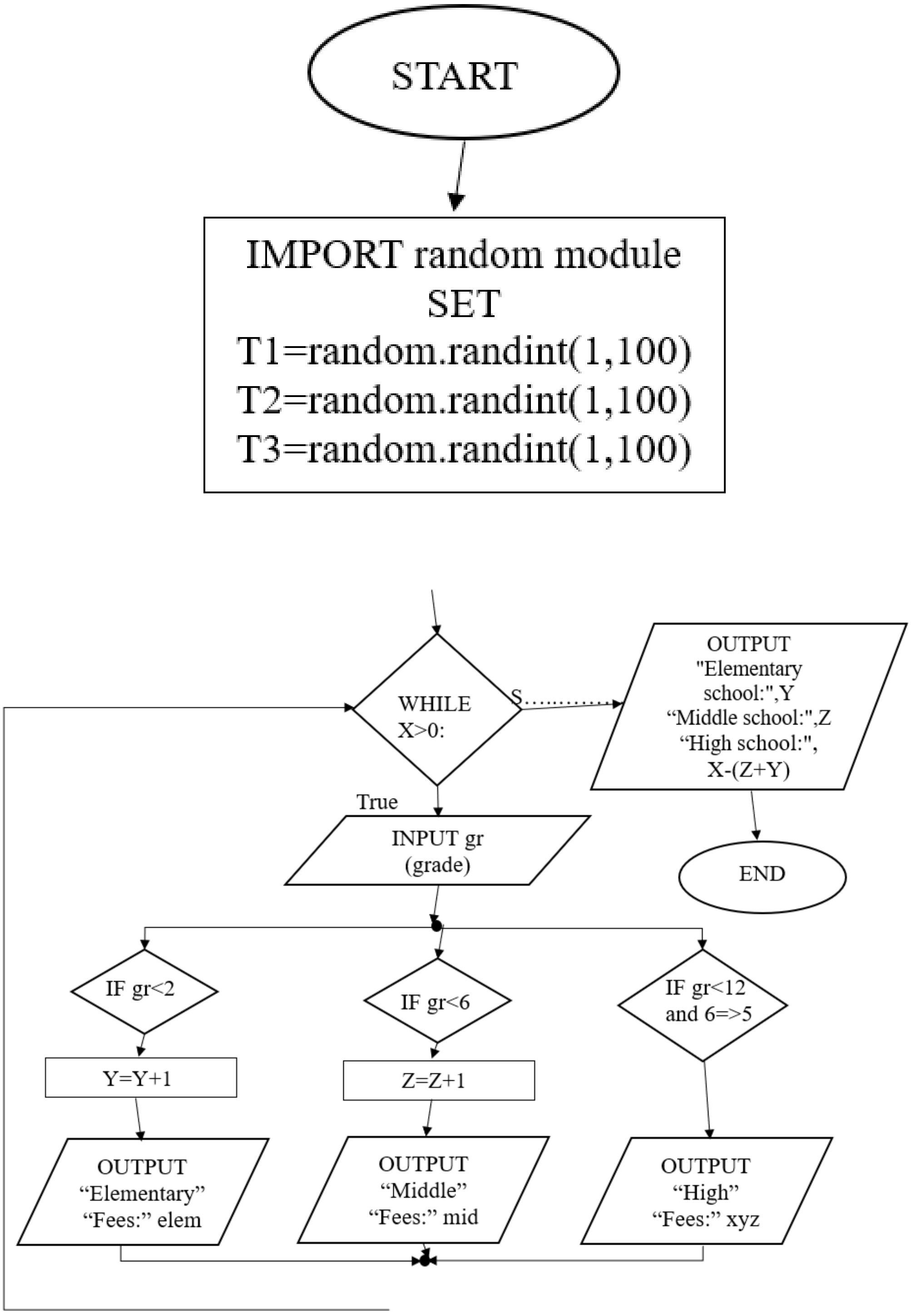
a. Identify the possible input and return value for each of the functions

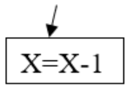
b. Write a pseudo code for each of the functions

c. Perform the ticket reservation task by invoking the appropriate functions

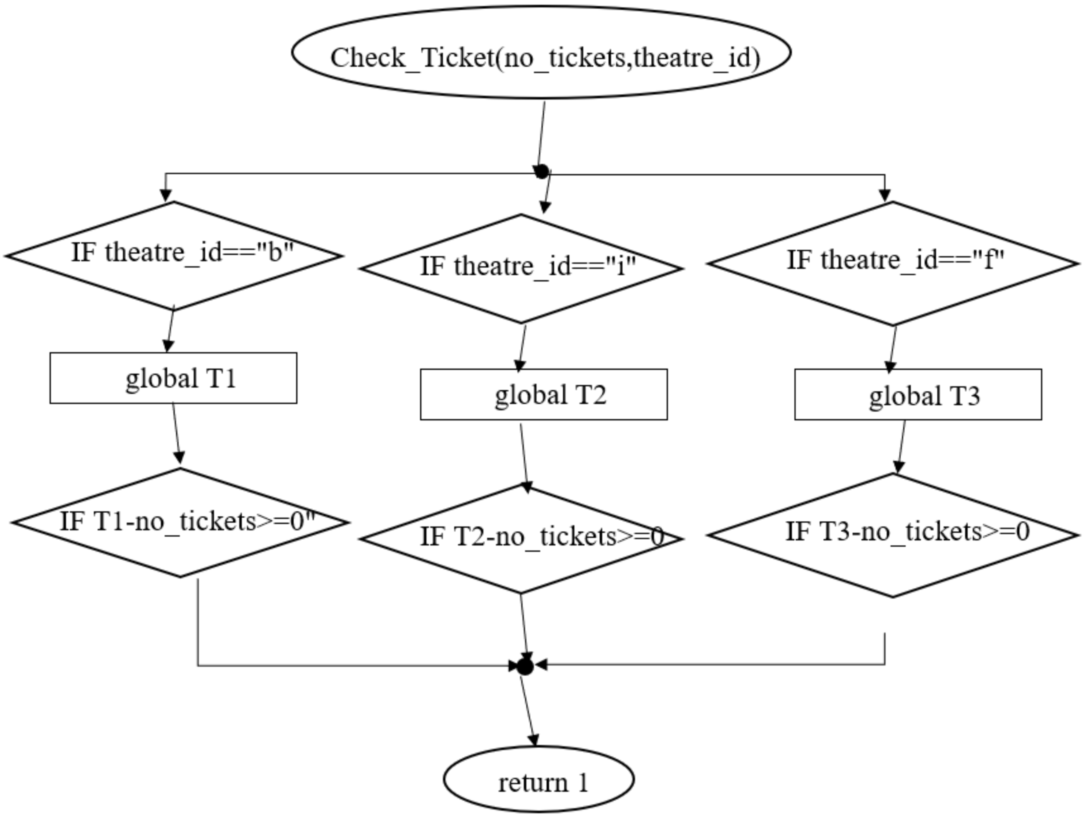
d. Repeat the task for N customers and keep the N value less than 5

Flowchart:

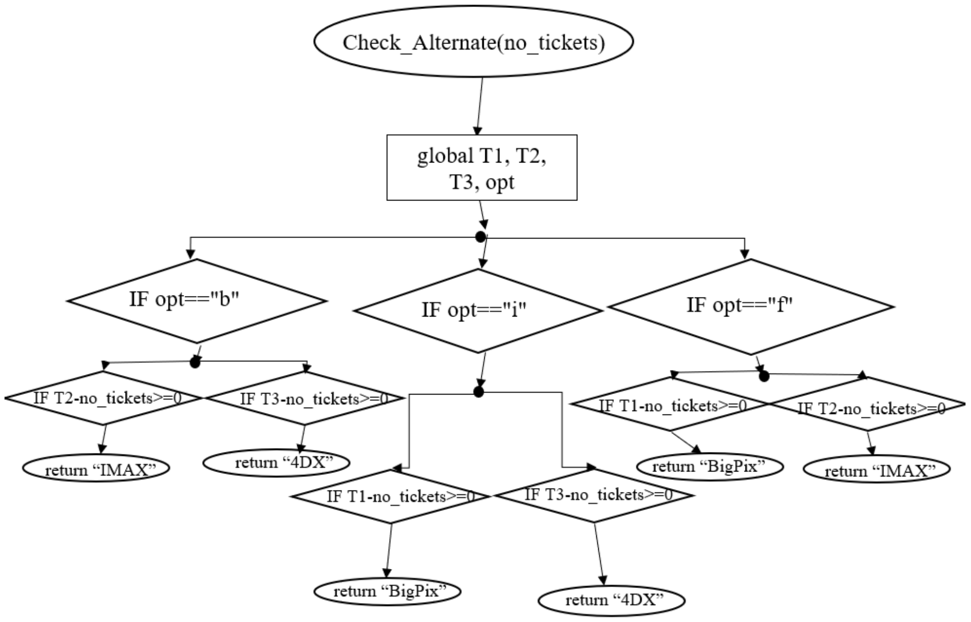

START
IMPORT random module
SET
T1=random.randint(1,100)
T2=random.randint(1,100)
T3=random.randint(1,100)
WHILE
X>0:
S
True
OUTPUT
"Elementary
school:",Y
“Middle school:",Z
“High school:",
X-(Z+Y)
END
INPUT gr
(grade)
IF gr<2
IF gr<6
IF gr<12
and 6=>5
Y=Y+1
Z=Z+1
OUTPUT
“Elementary”
“Fees:” elem
OUTPUT
“Middle”
“Fees:” mid
OUTPUT
“High”
“Fees:” xyz



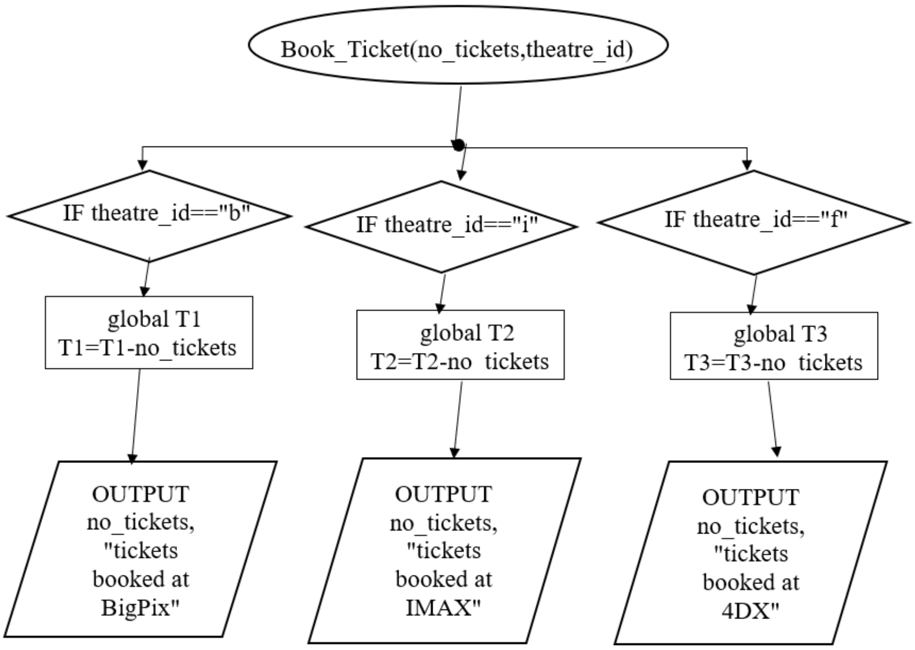
Check_ticket()



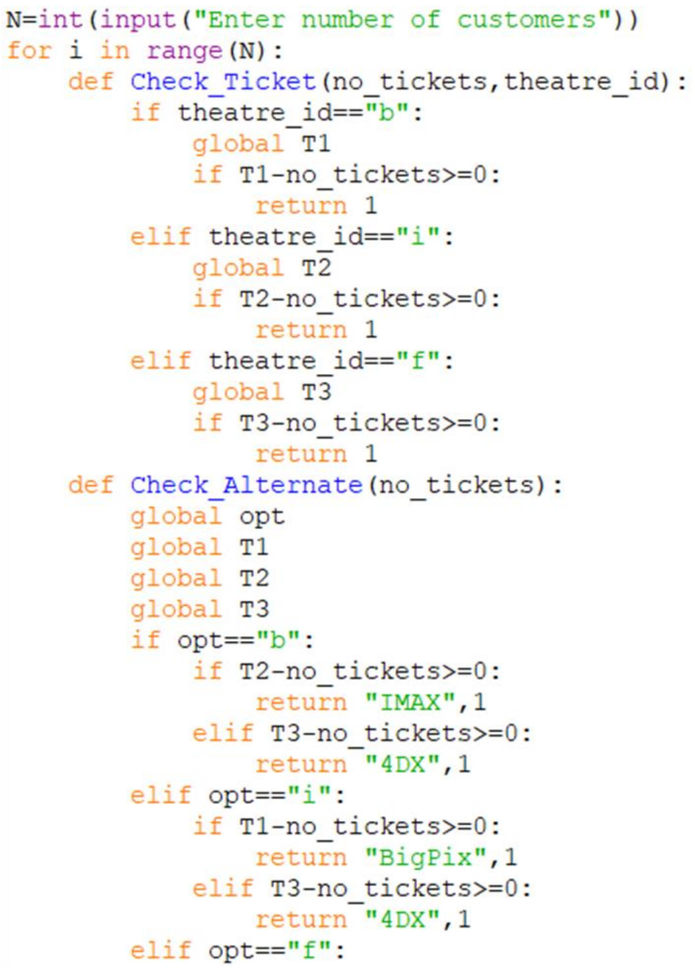
Check_Alternate()



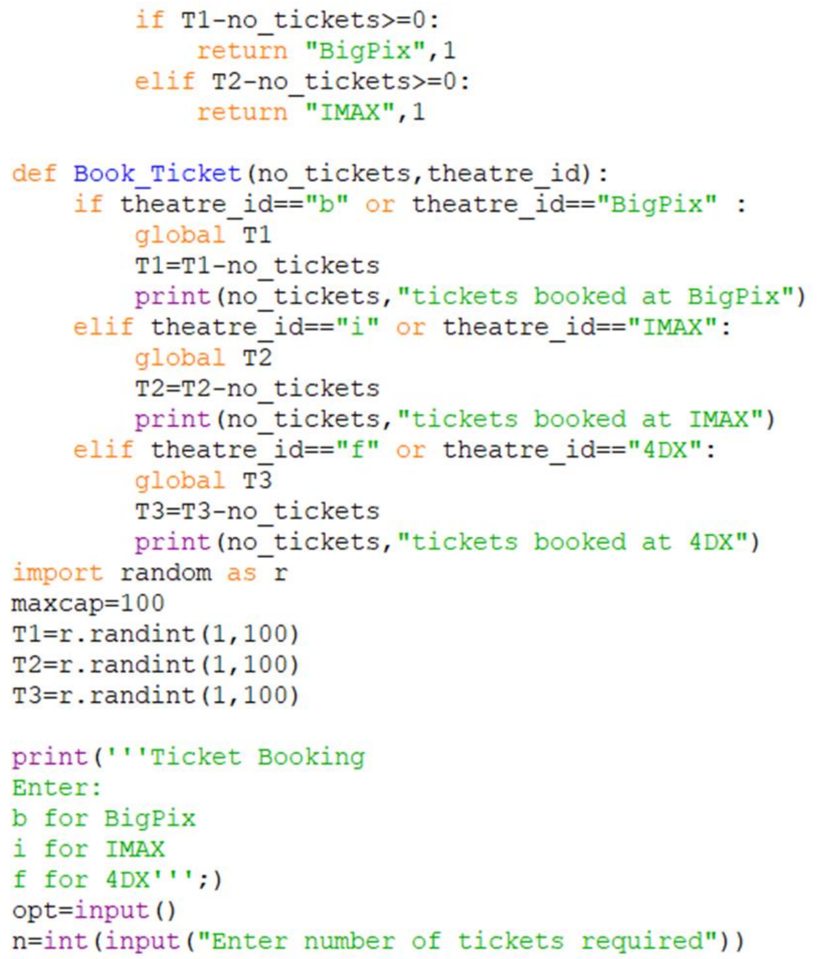
Book_ticket()


Book_Ticket(no_tickets,theatre_id)
IF theatre_id=="b"
IF theatre_id=="i"
IF theatre_id=="f"
global T1
T1=T1-no_tickets
global T2
T2=T2-no tickets
global T3
T3=T3-no tickets
OUTPUT no_tickets, "tickets booked at BigPix"
OUTPUT no_tickets, "tickets booked at IMAX"
OUTPUT no_tickets, "tickets booked at 4DX"

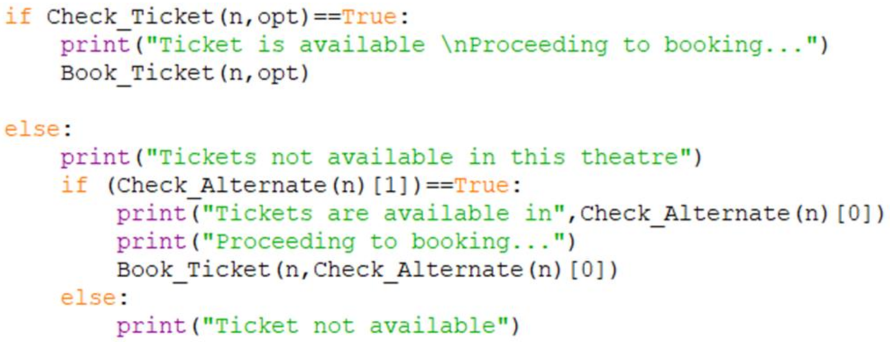
Source Code:



```
N=int(input("Enter number of customers"))
for i in range(N):
    def Check_Ticket(no_tickets,theatre_id):
        if theatre_id=="b":
            global T1
            if T1-no_tickets>=0:
                return 1
        elif theatre_id=="i":
            global T2
            if T2-no_tickets>=0:
                return 1
        elif theatre_id=="f":
            global T3
            if T3-no_tickets>=0:
                return 1
    def Check_Alternate(no_tickets):
        global opt
        global T1
        global T2
        global T3
        if opt=="b":
            if T2-no_tickets>=0:
                return "IMAX",1
            elif T3-no_tickets>=0:
                return "4DX",1
        elif opt=="i":
            if T1-no_tickets>=0:
                return "BigPix",1
            elif T3-no_tickets>=0:
                return "4DX",1
        elif opt=="f":
```

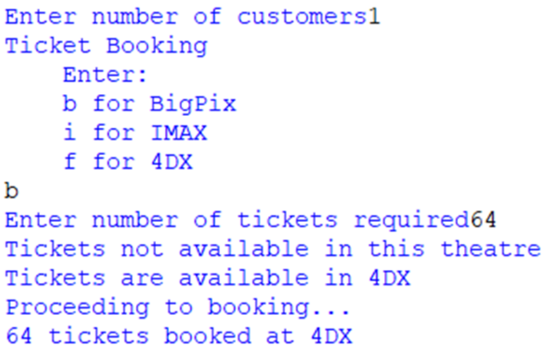


```
        if T1-no_tickets>=0:
            return "BigPix",1
        elif T2-no_tickets>=0:
            return "IMAX",1

def Book_Ticket(no_tickets,theatre_id):
    if theatre_id=="b" or theatre_id=="BigPix" :
        global T1
        T1=T1-no_tickets
        print(no_tickets,"tickets booked at BigPix")
    elif theatre_id=="i" or theatre_id=="IMAX":
        global T2
        T2=T2-no_tickets
        print(no_tickets,"tickets booked at IMAX")
    elif theatre_id=="f" or theatre_id=="4DX":
        global T3
        T3=T3-no_tickets
        print(no_tickets,"tickets booked at 4DX")
import random as r
maxcap=100
T1=r.randint(1,100)
T2=r.randint(1,100)
T3=r.randint(1,100)

print('''Ticket Booking
Enter:
b for BigPix
i for IMAX
f for 4DX''';)
opt=input()
n=int(input("Enter number of tickets required"))
```



```
if Check_Ticket(n,opt)==True:
    print("Ticket is available \nProceeding to booking...")
    Book_Ticket(n,opt)

else:
    print("Tickets not available in this theatre")
    if (Check_Alternate(n)[1])==True:
        print("Tickets are available in",Check_Alternate(n)[0])
        print("Proceeding to booking...")
        Book_Ticket(n,Check_Alternate(n)[0])
    else:
        print("Ticket not available")
```

Output:



```
Enter number of customers1
Ticket Booking
    Enter:
    b for BigPix
    i for IMAX
    f for 4DX
b
Enter number of tickets required64
Tickets not available in this theatre
Tickets are available in 4DX
Proceeding to booking...
64 tickets booked at 4DX
```

Question 2:

There are three types of courses namely Theory, Practicals and Theory cum Practice course. Internal marks will be calculated for every course for 50 marks. The internal marks for theory course is calculated by taking average of CAT 1 and CAT 2 marks which will be entered by the user. The internal marks for the practical course will be calculated by adding an average of the 5 continuous lab examinations. For the theory cum practical examinations, 60% of the marks comes from CAT 1 and CAT 2 and the remaining 40% marks are calculated from 4 continuous lab assessments.

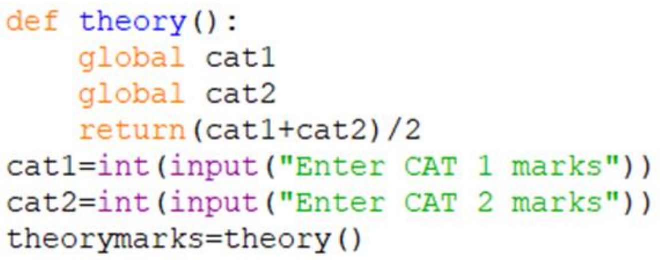
a. Identify the input and return value for each of the functions

b. Write a pseudo code for each of the functions

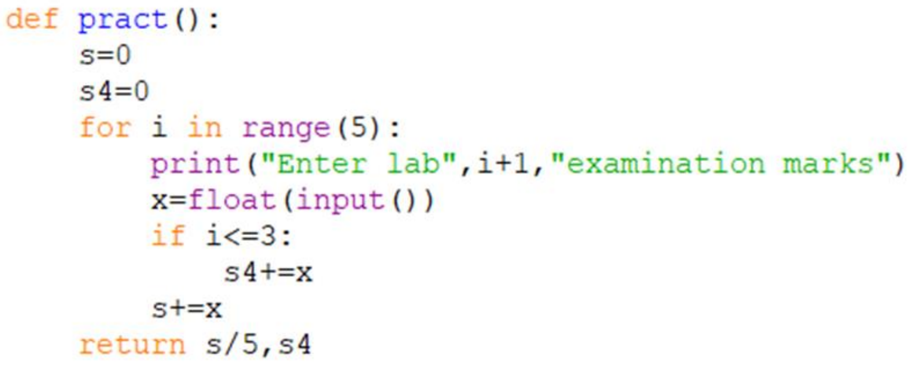
c. Write separate functions for each course type to calculate the internal marks according to the guidelines given in the question.

d. Write a function calctotalMarks() which calculates the sum of internal marks and endsemester marks given as input by the user.

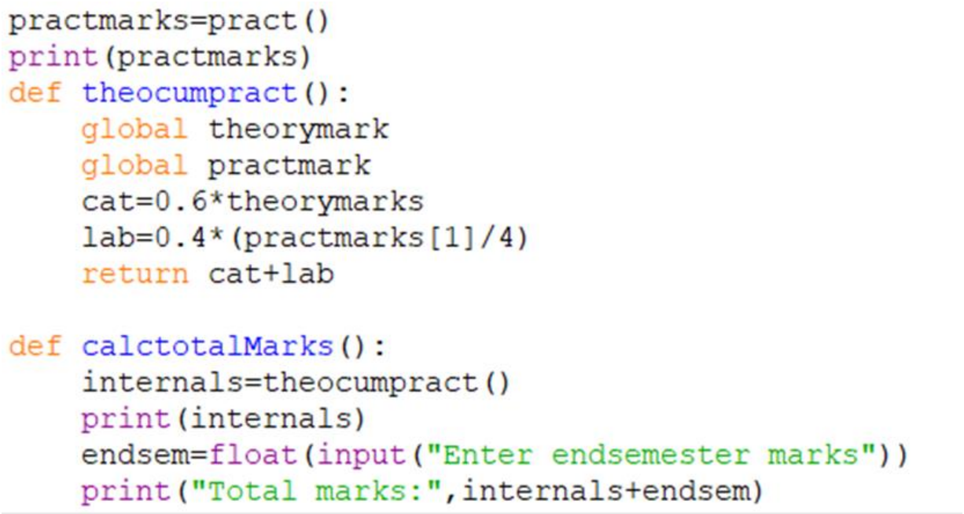
Source Code:



```
def theory():
    global cat1
    global cat2
    return(cat1+cat2)/2
cat1=int(input("Enter CAT 1 marks"))
cat2=int(input("Enter CAT 2 marks"))
theorymarks=theory()
```



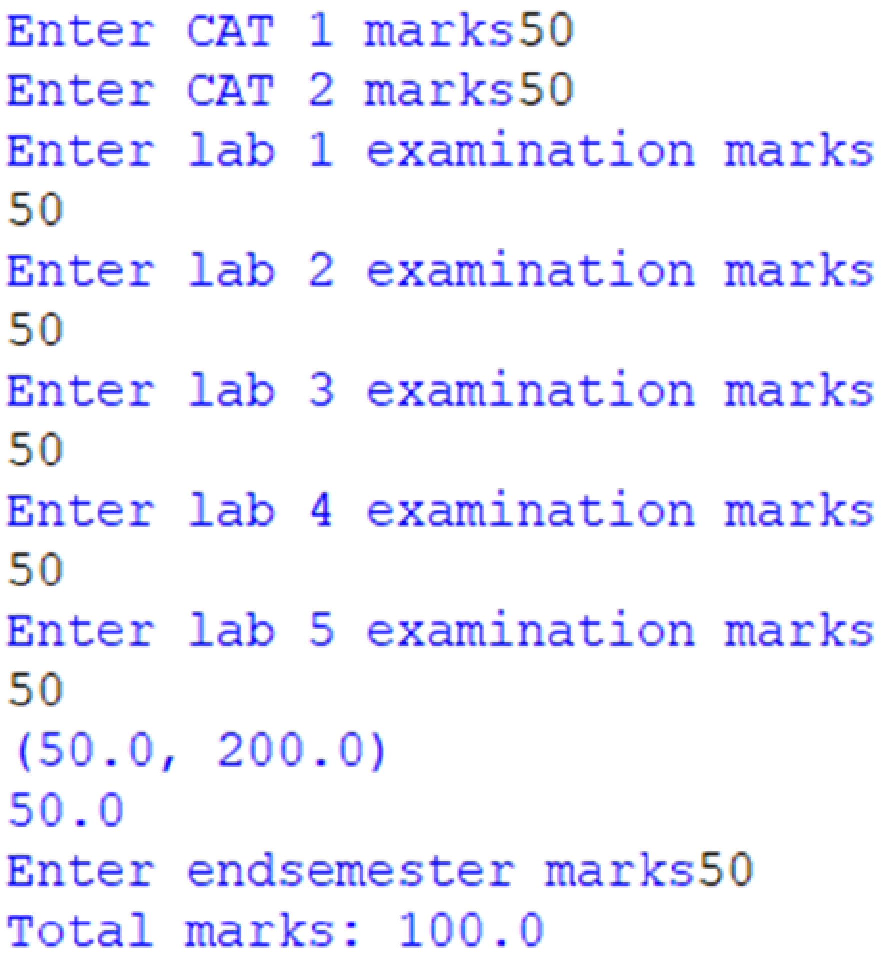
```
def pract():
    s=0
    s4=0
    for i in range(5):
        print("Enter lab",i+1,"examination marks")
        x=float(input())
        if i<=3:
            s4+=x
        s+=x
    return s/5,s4
```



```
practmarks=pract()
print(practmarks)
def theocumpract():
    global theorymark
    global practmark
    cat=0.6*theorymarks
    lab=0.4*(practmarks[1]/4)
    return cat+lab

def calctotalMarks():
    internals=theocumpract()
    print(internals)
    endsem=float(input("Enter endsemester marks"))
    print("Total marks:",internals+endsem)
```

```
calctotalMarks()
```

Output:



```
Enter CAT 1 marks50
Enter CAT 2 marks50
Enter lab 1 examination marks
50
Enter lab 2 examination marks
50
Enter lab 3 examination marks
50
Enter lab 4 examination marks
50
Enter lab 5 examination marks
50
(50.0, 200.0)
50.0
Enter endsemester marks50
Total marks: 100.0
```

Additional problems using functions for practice:

Question 1: Write a function to find the factorial of a given number.

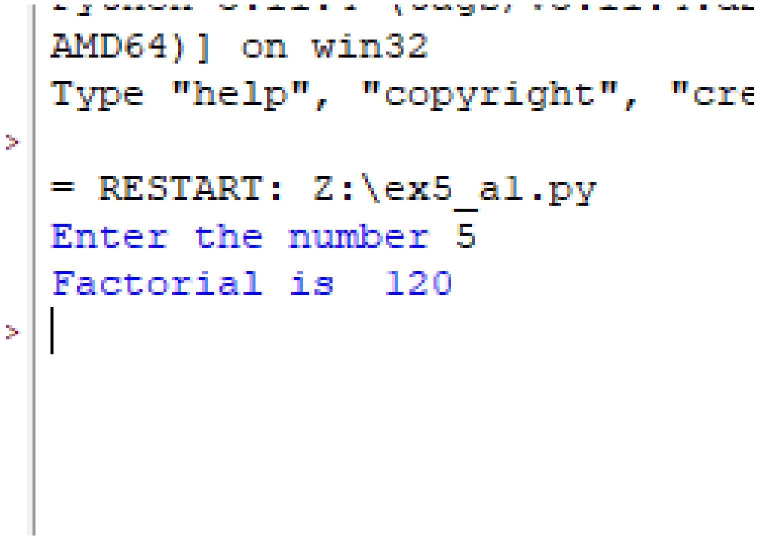
Algorithm:

```
Step 1: BEGIN
Step 2: Define Function fact()
Step 3: Initialize f=1
Step 4:  FOR i IN RANGE(1,n+1):
             f=f*i
Step 5: RETURN f
Step 6: READ the number, n
Step 7: Invoke the function fact() and assign it to variable p
Step 8: DISPLAY p
Step 9: END
```

Source code:

```
def fact(n):
   f=1
   for i in range(1,n+1):
      f=f*i
   return f
n=int(input("Enter the number"))
p=fact(n)
print("Factorial is ",p)
```

Output:



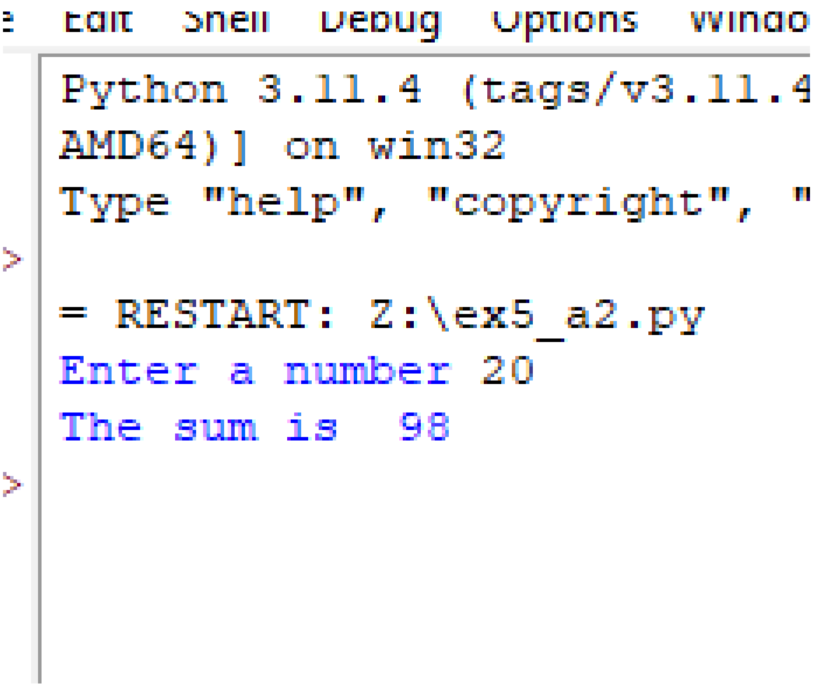
```
AMD64)] on win32
Type "help", "copyright", "cre
>
= RESTART: Z:\ex5_a1.py
Enter the number 5
Factorial is  120
>
```

Question 2: Write a function that returns the sum of multiples of 3 and 5 between 0 and limit (parameter). For example, if the limit is 20, it should return the sum of 3, 5, 6, 9, 10, 12, 15, 18, 20.

Source code:

```
def multi(n):
    sum=0
    for i in range(1,n+1):
        if(i%3==0 or i%5==0):
            sum=sum+i
    return sum
m=int(input("Enter a number"))
p=multi(m)
print("The sum is ",p)
```

Output:



```
Python 3.11.4 (tags/v3.11.4
AMD64)] on win32
Type "help", "copyright", "

= RESTART: Z:\ex5_a2.py
Enter a number 20
The sum is  98
```

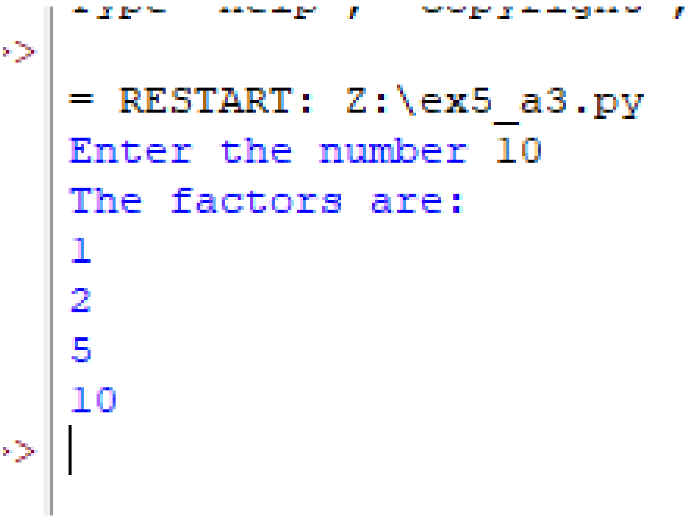
Question 3: Define a function to find the factors of the given number as an argument. If a number is not given, then display with default value

Source code:

```
def factors(n):
    print("The factors are:")
    for i in range(1,n+1):
        if n%i==0:
            print(i)
```

```
m=int(input("Enter the number"))
p=factors(m)
```

Output:



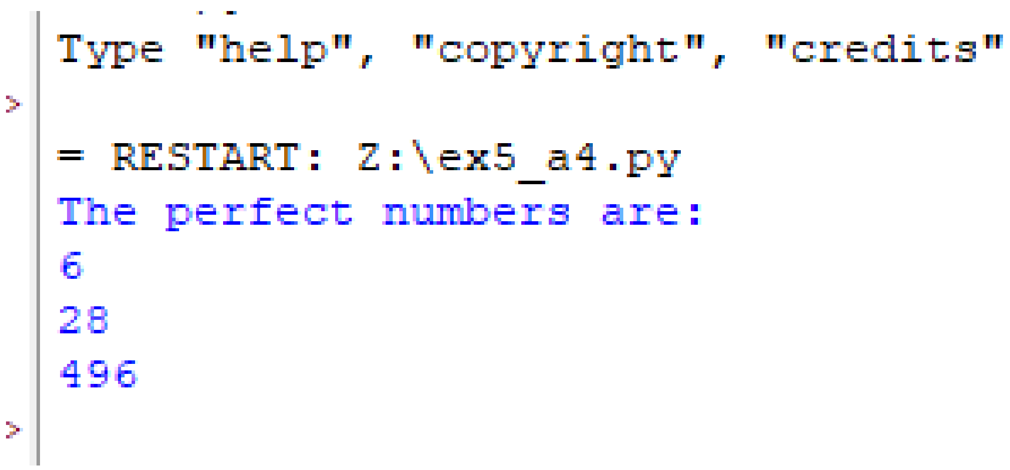
```
>>>
= RESTART: Z:\ex5_a3.py
Enter the number 10
The factors are:
1
2
5
10
>>> |
```

Question 4: Write a function “perfect()” that determines if a parameter number is a perfect number. Use this function in a program that determines and prints all the perfect numbers between 1 and 1000. [An integer number is said to be “perfect number” if its factors, including 1 (but not the number itself), sum to the number. E.g., 6 is a perfect number because 6=1+2+3].

Source code:

```
def perfect(n):
    sum=0
    for i in range(1,n):
        if n%i==0:
            sum=sum+i
    if(sum==n):
        return 1
    else:
        return 2
print("The perfect numbers are:")
for i in range(1,1001):
    if(perfect(i)==1):
        print(i)
    else:
        continue
```

Output:



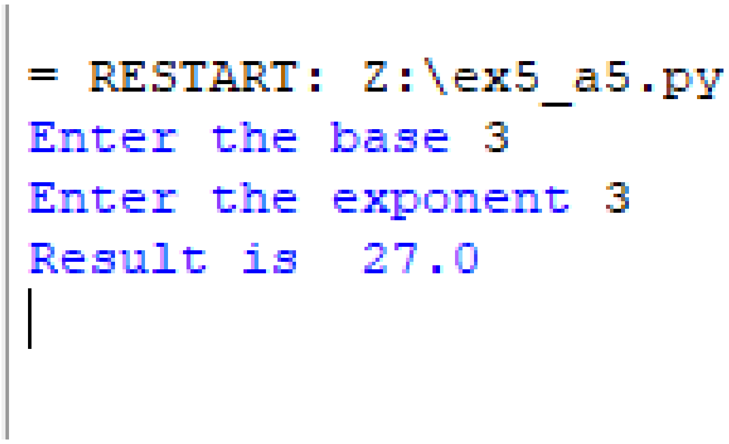
```
Type "help", "copyright", "credits"
>
= RESTART: Z:\ex5_a4.py
The perfect numbers are:
6
28
496
>
```

Question 5: Define a function calc_exp(base, exp) which computes the exponent of any number, i.e. baseexp . The base can be a float or integer and the exp is an integer greater than 0.

Source code:

```
def calc_exp(base,exp):
    p=base**exp
    return p
b=float(input("Enter the base"))
e=int(input("Enter the exponent"))
r=calc_exp(b,e)
print("Result is ",r)
```

Output:



```
= RESTART: Z:\ex5_a5.py
Enter the base 3
Enter the exponent 3
Result is  27.0
```

Learning outcome:

1. Reading inputs / Printing the result
2. Using appropriate datatypes for the given input
3. Variable assignment

4. Converting the formula into python expressions

Result:

Thus I learned to implement a simple problems in Python and solve the same using functions.