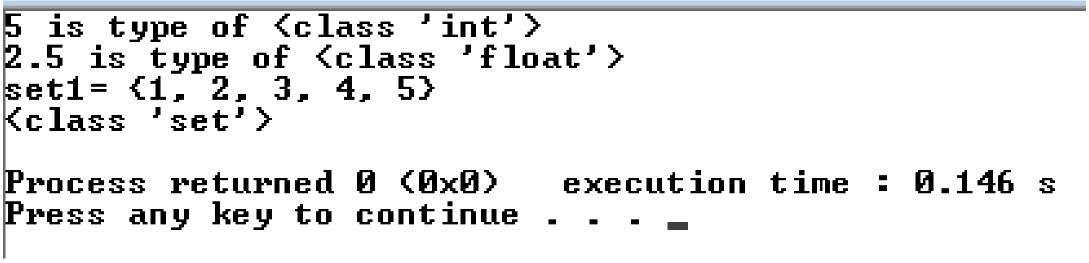
PDF_CODE

HIT137 Software Now
Week 2

TypeFindprogram.py

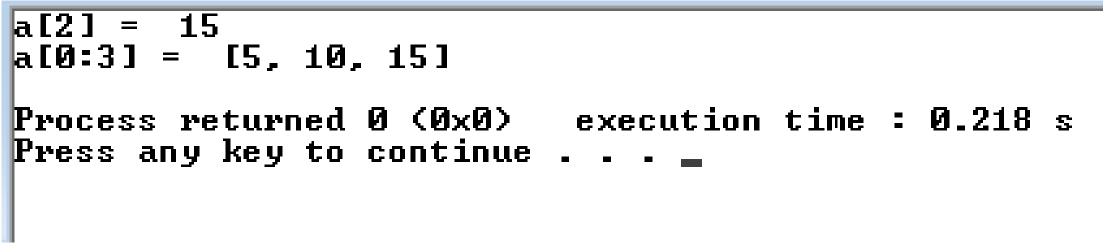
```
#example 1 - type find
a=5
print(a,"is type of", type(a))
b=2.5
print(b,"is type of", type(b))
#example 2 - set
set1={5,2,3,1,4}
print("set1=",set1)
print(type(set1))
```



```
5 is type of <class 'int'>
2.5 is type of <class 'float'>
set1= {1, 2, 3, 4, 5}
<class 'set'>

Process returned 0 (0x0)   execution time : 0.146 s
Press any key to continue . . .
```

ListIndexprogram.py

```
a = [5,10,15,20,25,30,35,40]
print("a[2] = ", a[2])
print("a[0:3] = ", a[0:3])
```

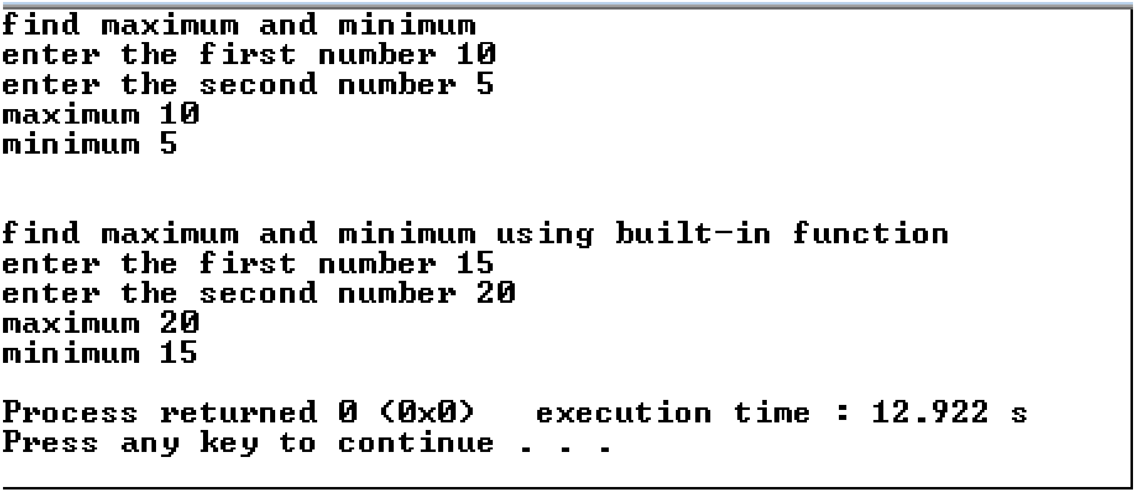


```
a[2] =  15
a[0:3] =  [5, 10, 15]

Process returned 0 (0x0)   execution time : 0.218 s
Press any key to continue . . .
```

---Refer to PPT

MaxandMinprogram.py

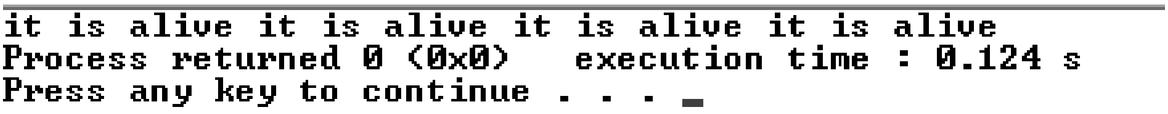
```
#example 1 - find maximum and minimum
first=int(input("enter the first number "))
second=int(input("enter the secomd number "))
if first>second:
    maximum=first
    minimum=second
else:
    maximum=second
    minimum=first
print("maximum",maximum)
print("minimum",minimum)

print("\n")

#example 2 - find maximum and minimum using built-in function
first=int(input("enter the first number "))
second=int(input("enter the secomd number "))
print("maximum",max(first,second))
print("minimum",min(first,second))
```



```
find maximum and minimum
enter the first number 10
enter the second number 5
maximum 10
minimum 5


find maximum and minimum using built-in function
enter the first number 15
enter the second number 20
maximum 20
minimum 15

Process returned 0 (0x0)   execution time : 12.922 s
Press any key to continue . . .
```

Print-n-times-program.py

```
for eachPass in range(4): # loop will run 4 times from 0 to 3
    print("it is alive", end=" ")
```



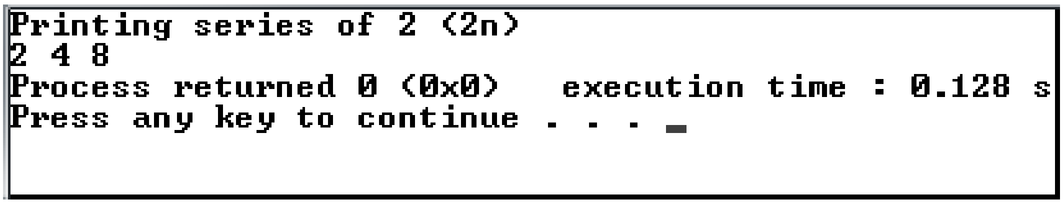
```
it is alive it is alive it is alive it is alive
Process returned 0 (0x0)   execution time : 0.124 s
Press any key to continue . . . _
```

SeriesandProductprogram.py

```
#Printing series of 2 (2n)
print("Printing series of 2 (2n)")
number=2
exponent=3
product=1
for eachPass in range(exponent):
    product=product*number
    print(product, end=" ")
```

number	exponent	product	eachPass
2	3	1	
		2	0
		4	1
		8	2

Printing series of 2 (2n)
2 4 8



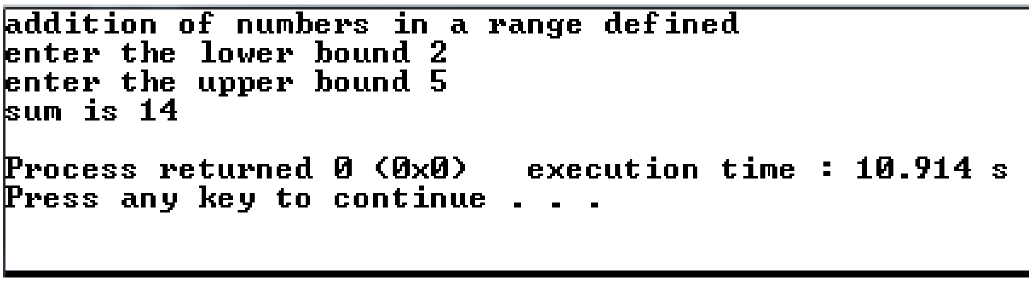
```
Printing series of 2 (2n)
2 4 8
Process returned 0 (0x0)   execution time : 0.128 s
Press any key to continue . . . _
```

NumberAdditionprogram.py

```
#addition of numbers in a range defined
print("addition of numbers in a range defined")
lower=int(input("enter the lower bound "))
upper=int(input("enter the upper bound "))
sum=0
for count in range(lower,upper+1):
    sum=sum+count
print("sum is", sum)
```

lower	upper	sum	count (2,6)
2	5	0	
		2	2
		5	3
		9	4
		14	5

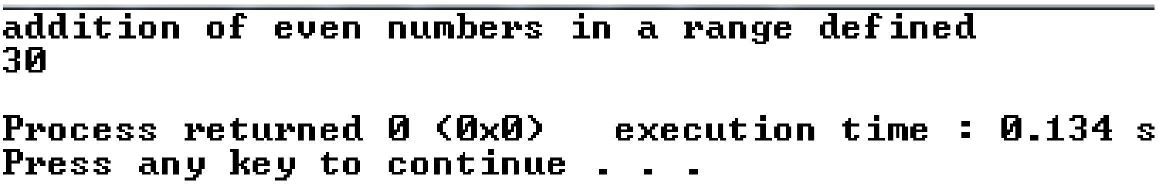
sum is 14



```
addition of numbers in a range defined
enter the lower bound 2
enter the upper bound 5
sum is 14

Process returned 0 (0x0)   execution time : 10.914 s
Press any key to continue . . .
```

EvenNumberAdd.py

```
#addition of even numbers in a range defined
print("addition of even numbers in a range defined")
sum=0
for count in range(2,11,2):
    sum=sum+count
print(sum)
```

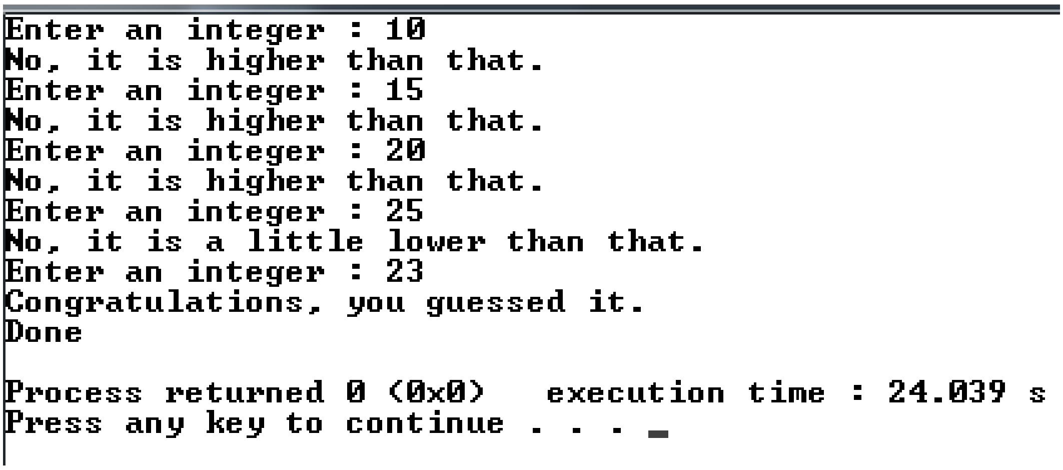


```
addition of even numbers in a range defined
30

Process returned 0 (0x0)   execution time : 0.134 s
Press any key to continue . . .
```

--Refer to PPT

NumberGuess.py

```
# Number guess problem
number = 23
running = True
while running:
         guess = int(input('Enter an integer : '))
         if guess == number:
             print('Congratulations, you guessed it.')
             running = False  # this causes the while loop to stop
         elif guess < number:
             print('No, it is higher than that.')
         else:
             print('No, it is a little lower than that.')
print('Done')
```

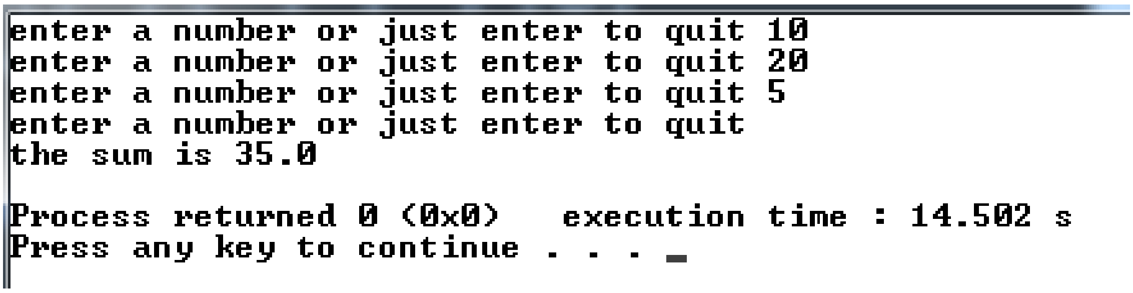


```
Enter an integer : 10
No, it is higher than that.
Enter an integer : 15
No, it is higher than that.
Enter an integer : 20
No, it is higher than that.
Enter an integer : 25
No, it is a little lower than that.
Enter an integer : 23
Congratulations, you guessed it.
Done

Process returned 0 (0x0)   execution time : 24.039 s
Press any key to continue . . . _
```

AddUserInput.py

```
#adding the user input numbers
sum=0.0
data=input("enter a number or just enter to quit ")
while data != "":
    number=float(data)
    sum=sum+number
    data=input("enter a number or just enter to quit ")
print("the sum is", sum)
```

pass	sum	data
	0.0	
1	0.0+10.0	10.0
2	10.0+20.0	20.0
3	30.0+5.0	5.0
4		enter
the sum is 35.0		



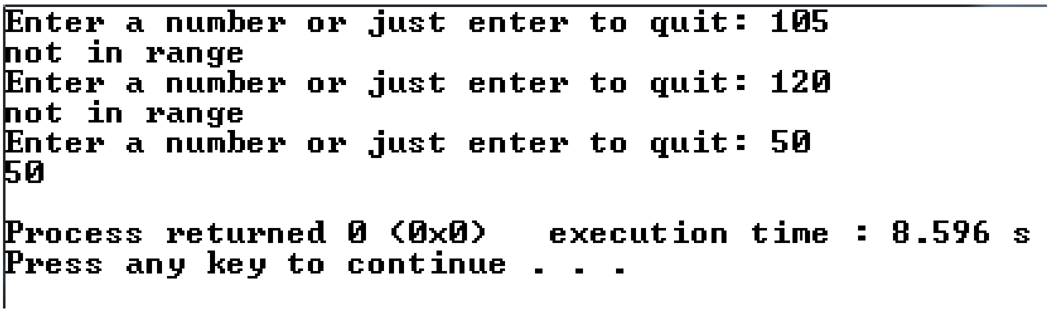
```
enter a number or just enter to quit 10
enter a number or just enter to quit 20
enter a number or just enter to quit 5
enter a number or just enter to quit
the sum is 35.0

Process returned 0 (0x0)   execution time : 14.502 s
Press any key to continue . . . _
```

TrueBreak.py

```
#while Loop and break statement
# use of True and break
sum=0.0
while True:
    data=input("Enter a number or just enter to quit: ")
    if data =="":
        break
    number=float(data)
    sum=sum+number
print("The sum is", sum)
```

Rangefind.py

```
#while Loop and break statement
while True:
    number=int(input("Enter a number or just enter to quit: "))
    if number >=0 and number <=100:
        break
    else:
        print("not in range")
print(number)
```

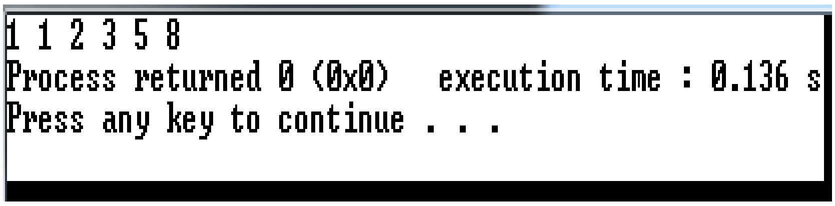


```
Enter a number or just enter to quit: 105
not in range
Enter a number or just enter to quit: 120
not in range
Enter a number or just enter to quit: 50
50

Process returned 0 (0x0)   execution time : 8.596 s
Press any key to continue . . .
```

---Refer to PPT

Fibonacci.py

```
#fibonacci sequence
a, b = 0, 1
while b < 10:
    print(b, end=" ")
    a, b = b, a+b
```



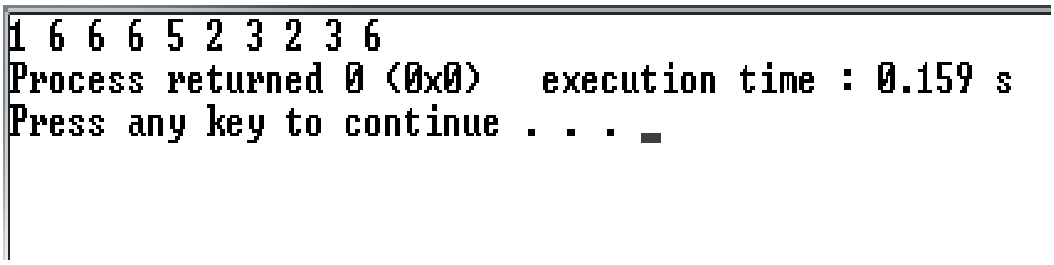
```
1 1 2 3 5 8
Process returned 0 (0x0)   execution time : 0.136 s
Press any key to continue . . .
```

a	b	b<10
0	1	True
1	1	True
1	2	True
2	3	True
3	5	True
5	8	True
8	13	False

1 1 2 3 5 8

RandomNumber.py

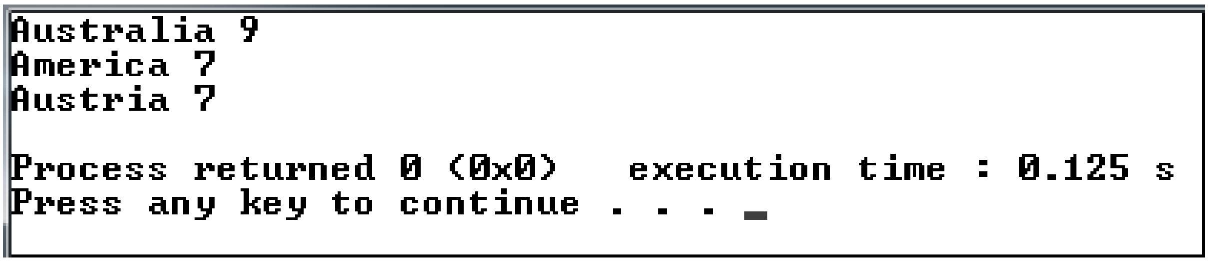
```
import random
for roll in range(10):
    print(random.randint(1,6), end=" ")
```



```
1 6 6 6 5 2 3 2 3 6
Process returned 0 (0x0)   execution time : 0.159 s
Press any key to continue . . .
```

Country.py

```
# Country Name and Length
country = ['Australia', 'America', 'Austria']
for c in country:
      print (c, len(c))
```



```
Australia 9
America 7
Austria 7

Process returned 0 (0x0)   execution time : 0.125 s
Press any key to continue . . .
```