

Name:Chockalingam.A

SRN:PES2UG21CS147

SEM-7

Day 3 Programs:

1. Write a program to design scientific calculator operations(any 10) using modules.

```
1.mojo U x
MOJO > DAY-3 > 1.mojo
1  from python import Python
2  import math
3
4  @value
5  struct MyCalculator:
6      var a: Int
7      var b: Int
8
9      fn add(self, a: Int, b: Int):
10         print(a + b)
11
12     fn sub(self, a: Int, b: Int):
13         print(a - b)
14
15     fn mul(self, a: Int, b: Int):
16         print(a * b)
17
18     fn div(self, a: Int, b: Int):
19         print(a / b)
20
21     fn exp(self, a: Int, b: Int):
22         print(a ** b)
23
24     fn square(self, a: Int):
25         print(a ** 2)
26
27     fn rem(self, a: Int, b: Int):
28         print(a % b)
29
30     fn SQRT(self, a: Int):
31         print(math.sqrt(a))
32
33     fn fact(self, a: Int):
34         print(math.factorial(a))
35
36     fn reciprocate(self, a: Int):
37         print(1/a)
38
```

```

MOJO > DAY-3 > 🔥 1.2.mojo
1  from python import Python
2  from MyCaIcuIator import Mycalculator
3
4  fn main() raises:
5      var py = Python.import_modules("builtins")
6      var x: Int = atol(py.input())
7      var y: Int = atol(py.input())
8      var calc = Mycalculator(x, y)
9      calc.add(x, y)
10     calc.sub(x, y)
11     calc.mul(x, y)
12     calc.div(x, y)
13     calc.exp(x, y)
14     calc.rem(x, Y)
15     calc. square(x)
16     calc.SQRT(x)
17     calc. reciprocate (x)
18     calc.fact (x)

```

```

100
50
150
50
5000
2.0
0
0
10000
10
0.01
94894792402599

```

2. Write a program to read student marks information and generate a result with cgpa using biotin functions.

2.mojo U X

MOJO > DAY-3 > 2.mojo

```
1 from python import Python
2 fn main() raises:
3     var py = Python.import_module("builtins")
4     var OOAD: Int = atol(py.input())
5     var CC: Int = atol(py.input())
6     var CD: Int = atol(py.input())
7     var NLP: Int = atol(py.input())
8     var TDL: Int = atol(py.input())
9     var marks = [OOAD, CC, CD, NLP, TDL]
10    print(py.sum(marks)/(10* py.len(marks)))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$ mojo 2.mojo

90

99

94

95

99

9.54

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$

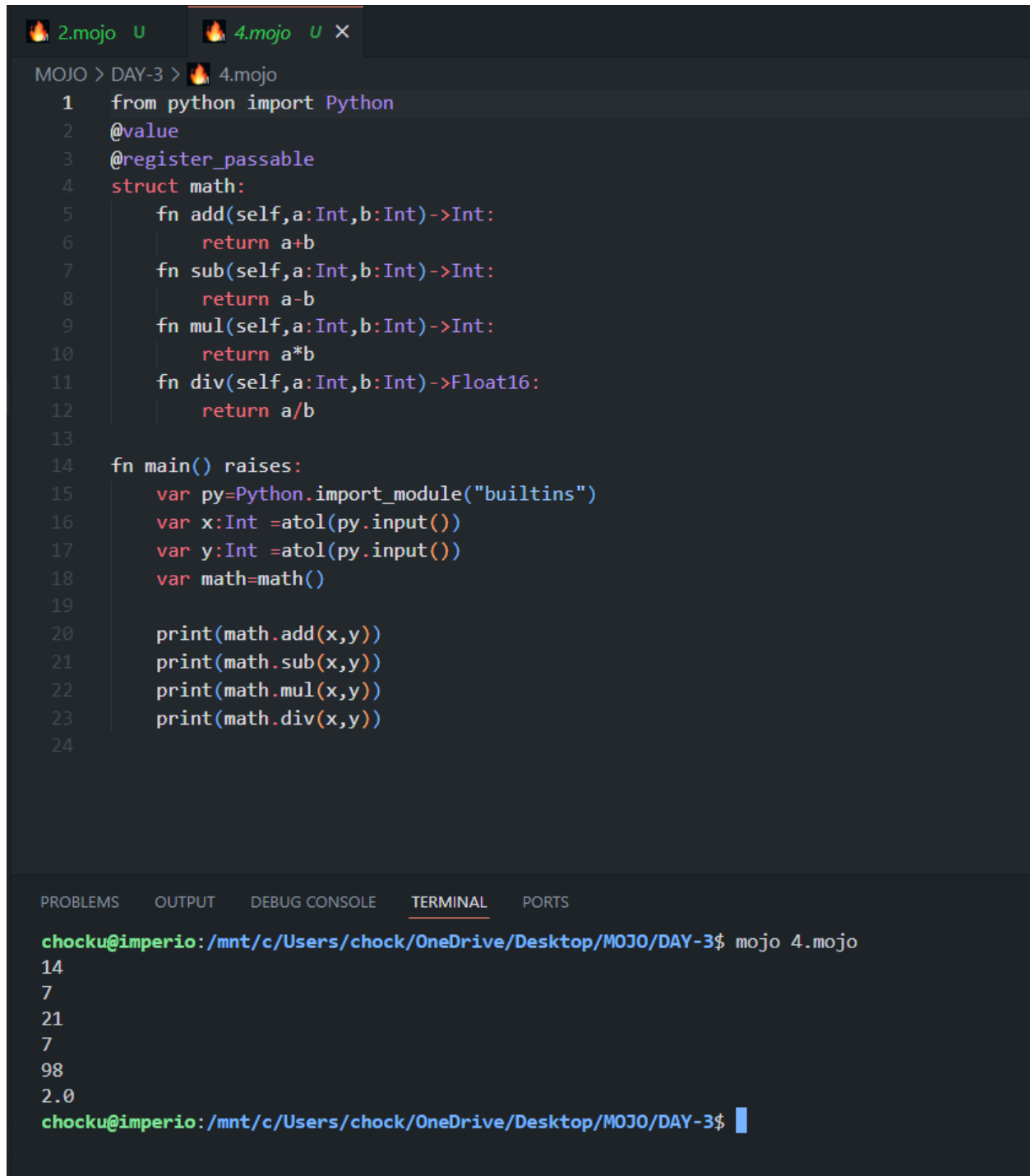
3. WAP to implement a 'person' struct with data members: name, age, email id and develop member functions: validate_data() with all necessary constructors using @value decorator.

```
MOJO > DAY-3 > 3.mojo
1  from python import Python
2  @value
3  struct person:
4      var name: String
5      var age: Int
6      var email: String
7
8      fn validate_data(self) raises:
9          var time = Python.import_module("time")
10         var os = Python.import_module("os")
11         os.system("clear")
12         print("Validating.")
13         time.sleep(1)
14         os.system("clear")
15         time.sleep(1)
16         os.system("clear")
17         print("Validating..")
18         time.sleep(1)
19         print("Validated user: ",self.name, self.age, self.email)
20
21 fn main() raises:
22     var py = Python.import_module("builtins")
23     var name: String = py.input()
24     var age: Int = atol(py.input())
25     var email: String = py.input()
26     var person = person(name, age, email)
27     person.validate_data()

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Validating..
Validated user:  chocku 21 xyz@gmail.com
chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3$
```

4. WAP to demonstrate all arithmetic operations in CPU register using @register_passable decorator.



The image shows a code editor with two tabs: '2.mojo U' and '4.mojo U X'. The active tab '4.mojo' contains the following code:

```
1  from python import Python
2  @value
3  @register_passable
4  struct math:
5      fn add(self,a:Int,b:Int)->Int:
6          return a+b
7      fn sub(self,a:Int,b:Int)->Int:
8          return a-b
9      fn mul(self,a:Int,b:Int)->Int:
10         return a*b
11     fn div(self,a:Int,b:Int)->Float16:
12         return a/b
13
14 fn main() raises:
15     var py=Python.import_module("builtins")
16     var x:Int =atol(py.input())
17     var y:Int =atol(py.input())
18     var math=math()
19
20     print(math.add(x,y))
21     print(math.sub(x,y))
22     print(math.mul(x,y))
23     print(math.div(x,y))
24
```

Below the code editor is a terminal window with tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The terminal shows the command 'mojo 4.mojo' being executed, followed by the output of the program:

```
chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3$ mojo 4.mojo
14
7
21
7
98
2.0
chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3$
```

5. WAP to form 10 numbers from SIMD datatype and calculate its square. Also print the original and square values of the numbers.

MOJO > DAY-3 > 5.mojo

```
1 fn main():
2     var vec=SIMD[DType.int32,16](1,2,3,4,5,6,7,8,9,10)
3     for i in range(0,10):
4         print(vec[i],end=" ")
5     print()
6     var vec2=vec**2
7     for i in range(0,10):
8         print(vec2[i],end=" ")
9     print()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$ mojo 5.mojo

1 2 3 4 5 6 7 8 9 10

1 4 9 16 25 36 49 64 81 100

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$

6. WAP to read a .CSV file using Pandas library in Mojo.

MOJO > DAY-3 > 6.mojo

```
1  from python import Python
2  fn main() raises:
3      var py = Python.import_module("pandas")
4      var df = py.read_csv("iris.data.csv")
5      print(df.head())
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$ mojo 6.mojo

```
5.1 3.5 1.4 0.2 Iris-setosa
0 4.9 3.0 1.4 0.2 Iris-setosa
1 4.7 3.2 1.3 0.2 Iris-setosa
2 4.6 3.1 1.5 0.2 Iris-setosa
3 5.0 3.6 1.4 0.2 Iris-setosa
4 5.4 3.9 1.7 0.4 Iris-setosa
```

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$

7. WAP to generate random numbers from 1-128 using necessary python library.

MOJO > DAY-3 > 7.mojo

```
1  from python import Python
2  fn main() raises:
3      var py=Python.import_module("random")
4      print(py.randrange(1,128))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$ mojo 7.mojo
76

chocku@imperio:/mnt/c/Users/chock/OneDrive/Desktop/MOJO/DAY-3\$