Minor Assignment-1

Basic Elements of Python Programming

1. Evaluate the following expression

(x<y) or (not(z==y) and (z<x))

(a) x=0, y=6, z=10

(0<6) or (not(10==6) and (10<0))

True or (not(False) and False)

True or (True and False)

True or False

True

(b) x=1, y=1, z=1

(1<1) or (not(1==1) and (1<1))

False or (not(True) and False)

False or (False and False)

False or False

False

2. Evaluate the following expressions involving arithmetic operators:

(a) -7*20+8/16*2+54

-140+0.5*2+54

-140+1.0+54

-85.0

(b) 7**2//9%3

=> 49//9%3

=> 5%3

=> 2

(c) (7-4**2)*10-25*8//5

=> (7-8)*10-25*8//5

=> -1*10-25*8//5

=> -10-200//5

=> -10-40 = -50

(d) 5%10+10-25*8//5

5+10-200//5

5+10-40

-25

(e) 'hello'*2-5

TypeError: unsupported operand type(s) for -: 'str' and 'int'

Name:---------- Regd. Number:----------

3. Evaluate the following expression

(a) 'hi' > 'hello' and 'bye' < 'Bye'

True and False

False

(b) 'hi' > 'hello' or 'bye' < 'Bye'

True or False

True

(c) 7 > 8 or 5 < 6 and 'I am fine' > 'I am not fine'

False or True and False

False or False

False

(d) 10 != 9 and 29 >= 29

True and True

True

(e) 10 != 9 and 29 >= 29 and 'hi' > 'hello' or 'bye' < 'Bye' and 7 <= 25

True and True and True or False and False

True and True or False and False

True or False

True

4. Evaluate the following expression

(a) 5 % 10 + 10 < 50 and 29 >= 29

5 + 10 < 50 and 29 >= 29

15 < 50 and 29 >= 29

True and True

True

(b) 7 ** 2 <= 5 // 9 % 3 or 'bye' < 'Bye'

49 <= 5 // 9 % 3 or 'bye' < 'Bye'

49 <= 0 or 'bye' < 'Bye'

False or False

False

Name:---------- Regd. Number:----------

(c) 5%10 < 8 and -25 > 1 * 8 // 5

5 < 8 and -25 > 8 // 5

5 < 8 and -25 > 1

True and False

False

(d) 7 ** 2 // 4 + 5 > 8 or 5 != 6

49 // 4 + 5 > 8 or 5 != 6

17 > 8 or 5 != 6

True or True

True

(e) 7/4 < 6 and 'I am fine' > 'I am not fine'

1.75 < 6 and 'I am fine' > 'I am not fine'

True and False

False

(f) 10 + 6 * 2 ** 2 != 9 // 4 - 3 and 29 >= 29/9

10 + 6 * 4 != 9 // 4 - 3 and 29 >= 29/9

34 != -1 and 29 >= 3.2222...3

True and True

True

(g) 'hello' * 5 > 'hello' or 'bye' < 'Bye'

True or False

True

5. Evaluate the following expression using bitwise operators:

(a) 15 & 22

15: 00001111

22: 00010110

(00000110)$_2$ = (6)$_{10}$

15 & 22 = 6

(b) 15 | 22

15: 00001111

22: 00010110

000 11111

$= 2^4 + 2^3 + 2^2 + 2^1 + 2^0$

$= 16 + 8 + 4 + 2 + 1$

$= (31)_{10}$

15 | 22 = 31

Name:---------- Regd. Number:----------

(c) -15 & 22

Binary of -15 = 2's complement of Binary of $(15)_{10}$

· 2's complement of 00001111

= $(11110001)_2$

-15: 11110001
22: 00010110

00010000

$(00010000)_2 = (16)_{10}$

(d) -15 | 22

-15: 11110001
22: 00010110

11110111

For Binary of 11110111, take 2's comp as MSB = 1

2's comp of 11110111 = 00001001

Now, converting this to decimal & add -ve sign

$(00001001)_2 = (9)_{10}$

$(11110111)_2 = (-9)_{10}$

-15 | 22 = -9

(e) ~15

~x = 1's complement of

Binary of x = 15 = 00001111

1's complement = $(11110000)_2$

For Binary of 11110000, take 2's comp., convert to decimal & add (ve)

2s comp. of 11110000 = 00010000

= 16

$(11110000)_2 = (-16)_{10}$

~15 = -16

Name:---------- Regd. Number:----------

(f) ~22

Binary of 22 = 00010110

1's complement = 11101001

Now,

Decimal of 11101001 can be evaluated by calculating the 2's comp. & add (-ve) to its decimal representation

2's comp. of 11101001 = 00010111

= $(23)_{10}$

11101001 = $(-23)_{10}$

~22 = -23

(g) ~ -20

Binary of -20 = 2's complement of binary of $(20)_{10}$

= 2s complement of 00010100

= $(11101100)_2$

~ -20 = 1's comp of 11101100

= $(00010011)_2$

= $(19)_{10}$

(h) 15 ^ 22

15: 00001111

22: 00010110

00011001

1^0 = 1
1^1 = 0
0^0 = 0

= $(25)_{10}$

15^22 = 25

(i) 8 << 3

$(8)_{10}$ = $(00001000)_2$

= 01000000

= $(64)_{10}$

(j) 40 >>3

(00101000) >>3

$(00000101)_2$

$(5)_{10}$

Name:———————— Regd. Number:————————

6. Differentiate between the following operators with the help of examples:

(a) = and ==

= is the assignment operator, which assign a value to the variable

Eg:- a=10
b=20

== is the equality operator, which compares two values & returns true if they are equal and False otherwise.

a==10 # True
a==b # False

(b) / and %

/ : is the division operator, which divides two no.& returns the quotient.

Eg:- 10/5=2.0

% : is the modulus operator which divides two nos. and returns the remainder.

Eg:- 10%5=0

(c) / and //

/ : is the division operator which divides two nos. and returns the quotient in double (decimal form)

Eg:- 9/4 = 2.25

// : is the integer division which divides two nos. and returns the quotient in Integer form

Eg:- 9//4 = 2

(d) * and **

*: is the multiplication operator, which returns the product of two numbers.

Eg:- 10 * 2 = 20

**: is the exponential operator, which raises the first no. to the power of second no.

Eg:- 10 ** 2 = 100

7. Output displayed when following ~~codes~~ commands are executed in Python shell

(a)
```
>>> a = 6
>>> a == 6
True
>>> a < 5.9
False
>>> a > 5.9
True
```

(b)
```
>>> b = 7
>>> b/6
1.1666666666666667
>>> b//6
1
>>> b/4
1.75
>>> b%4
3
>>> b%7
0
>>> b*2
14
>>> b**2
49
```

8. Construct logical expression for representing the following condition:

(a) marks scored should be greater than 300 & less than 400

(marks > 300) & (marks < 400)

(b) Whether the value of grade is an uppercase letter.

grade.isupper()

(c) The post is Engineer & experience is more than 4 years

(post == 'engineer') & (experience > 4)

9. Write Python statements for following eqn:-

(a) root 1 = $\frac{-b+\sqrt{b^2-4ac}}{2a}$

```
import math
a= int(input("Enter a")
b= int(input("Enter b")
c= int(input("Enter c")
d= (b**2)-(4*a*c)
root1 = (-b + math.sqrt(d))/(2*a)
print(root1)
```

(b) result = $\frac{2xy-9y}{2xy^3} - \frac{4yx^2}{2y}$

```
x = int(input("Enter x"))
y = int(input("Enter y"))
result = ((2*x*y - 9*y)/(2*x*(y**3))) - ((4*y*(x**2))/(2*y))
print(result)
```

(c) result = $2\cos\frac{1}{2}(x+y)\cos\frac{1}{2}(x-y) + e^x - 1 - \frac{x}{4} + \tan x - \log(v)$

```
import math
x = int(input("Enter x"))
y= int(input("Enter y"):
V= int(input("Enter V");
result= (2* math.cos(0.5*(x+y)) * math.cos*(0.5*(x-y)) +
        math.e**x - 1 - (x/4) + math.tan(x) - math.log(v)
print(result)
```

10. How does the effect of following two statement differs

(a) x += x + 10 (b) x = x + 10

x += x + 10 means the variable x will be updated by x + x + 10 i.e., 2x + 10

x = x + 10 means the variable x will be updated by x + 10

11. WAP that asks the user to enter the width & length of room. Write a program to display the area of room.

```
def calculate_area(width, length):
    area = width * length
    return area

width = float(input("Enter the width of the room"))
length = float(input("Enter the length of the room"))

area = calculate_area(width, length)
print("Area of the room is ", area, "square meter")
```

12. WAP that reads the no. of widgets and gizmos from the user. It should display the total weight of the parts

```
widget = int(input("Enter no. of widget"))
gizmos = int(input("Enter no. of gizmos"))
widget_weight = widget * 75
gizmos_weight = gizmos * 112
print("Total weight of widget is ", widget_weight, "gms")
print("Total weight of gizmos is", gizmos_weight, "gms")
print("Total weight of widget and gizmos is", widget_weight + gizmos_weight,
      "gms")
```

13. WAP that determines how quickly an object is travelling when it hits the ground.

$$v_f = \sqrt{v_i^2 + 2ad}$$ here Initial velocity = 0 m/s

```
import math
height = float(input("Enter the height "))
g = 9.8
final_velocity = math.sqrt( 2 * g * height )
print(" Object was travelling at ", final_velocity, "m/s")
```

14. WAP that reads a 4-digit integer & display the sum

```
num = int(input ("Enter the number")) # Say 3141
sum = 0
sum = sum + num %10        # sum = 0+1
num = num //10             # num = 314
sum = sum + num %10        # sum = 0+1+4
num = num //10             # num = 31
sum = sum + num %10        # sum = 0+1+4+1
num = num //10             # num = 3
sum = sum+num              # sum = 0+1+4+1+3
print(sum)
```

15. WAP that reads three integer & display them in sorted order

```
x = int (input ("Enter the number"))
y = int (input ("Enter the number"))
z = int (input ("Enter the number"))
max = max (x,y,z)
min = min (x,y,z)
mid = x+y+z -max- min
print (" Number in sorted order is", min, mid, max )
```

16. WAP that reads duration from user as days, hours, minutes and seconds and display total no. of seconds

```
day = int (input ("Enter number of days")
hour = int (input ("Enter hours"))
min = int (input ("Enter minutes"))
sec = int (input ("Enter no. of seconds"))
total_sec = (3600 *24 * day) + (3600 * hour) + (60 * min) + sec
print ("Total number of seconds", total_sec )
```