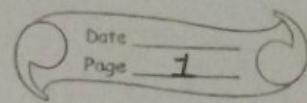


DIV:- E

Assignment :- 2.

Q. 1.  
A. Explain arithmetic operator with example.

Arithmetic operator (+, -, \*, /, %).

=>> + Addition :- Adds values on either side of the operator.

Example :-  $a+b$  will give 30.

=>> - Subtraction :- subtracts right hand operand from left hand operand.

Example :-  $a-b$  will give - 10.

=>> \* Multiplication :- multiplies values on either side of the operator.

Example :-  $a * b$  will give 200.

=>> / Division :- Divides left hand operand by right hand operand.

Example :-  $b/a$  will give 2.

=>> % Modulus :- Divides left hand operand by right hand operand and returns remainder.

Example :-  $b \% a$  will give 0.

=>> \*\* Exponent :- performs exponential (power) calculation on operators. Example :-  $a ** b$  will give 10 to the power 20.

=>> // Floor Division :- The quotient in which the digits after the decimal point are removed.

2. Explain membership operator with example.

A. In addition to the operators discussed previously, Python has membership operators, which test for membership in a sequence, such as strings, lists, or tuples.

Operator

in

Description

Evaluates to true if it finds a variable in the specified sequence and false otherwise

not in

Evaluates to true if it does not find a variable in the specified sequence and false otherwise

\*

Example of Membership operator :-

```
a = 4
list1 = [1, 2, 3, 4, 5]
if (a in list1):
    print ("a is available in list")
else:
    print ("a is not available in list")
```

Output

a is available in list

- Q. 3. Explain identity operator with example.
- A.
- ⇒ Identity operator compare the memory locations of two objects.
  - ⇒ Identity operators are used to determine whether a value is of a certain class or type.
  - ⇒ They are usually used to determine the type of data a certain variable.

operator is	Description
	The same object and false otherwise.

is not	Description
	The same object and true otherwise

\* Example :-

$x = 8$

```
if (type(x) is int):
    print ("true")
```

else

```
    print ("false")
```

Output

true.

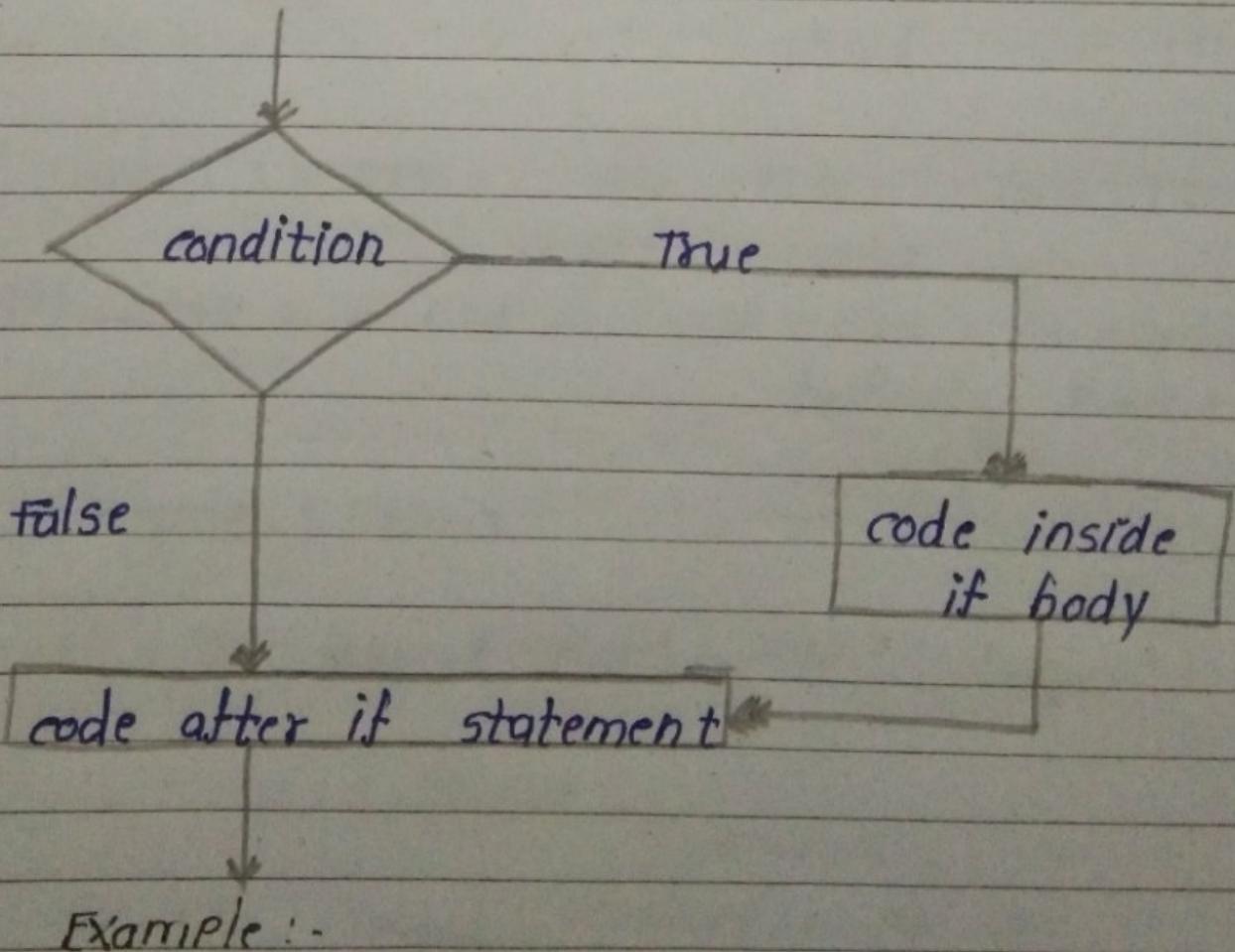
$x = 8.5$

```
if (type(x) is not int):
    print ("true")
else :
    print ("false")
```

Output

true

4. Explain if Loop with example.
- ⇒ Python if statement is used for decision-making operations.
- ⇒ It contains a body of code which runs only when the condition given in the if statement is true.
- ⇒ If the condition is false, then the optional else statement runs which contains some code for the else.



Q. 5. write a program of print the largest to three number in python.

A.

# change the values of num1, num2 and num3  
# for a different result

num1 = 10

num2 = 14

num3 = 12

# num1 = float(input("Enter first number:"))

# num2 = float(input("Enter second number:"))

# num3 = float(input("Enter third number:"))

if (num1 >= num2) and (num1 >= num3):

    largest = num1

elif (num2 >= num1) and (num2 >= num3):

    largest = num2

else:

    largest = num3

print("The largest number is", largest)

Output

The largest number is 14.

Q. 6. write a program to check whether a number is even or not.

A.

```
num = int(input("Enter any number to test whether it is odd or even:"))
```

```
if (num % 2) == 0:
```

```
    print("the number is even")
```

```
else:
```

```
    print("the provided number is odd")
```

Output

Enter any number to test whether it is odd or even :

887

887 is odd.

Q. 7. Explain for loop with example.

A.

⇒ The for loop in python has the ability to iterate over the items of any sequence, such as a list or a string.

⇒ The syntax of the loop look is :

for iterating\_var in Sequence: statement(s).

\*

### EXAMPLE :-

```
for letter in 'python':      # first example
    print 'current letter : ', letter
```

```
Fruits = [ 'banana', 'apple', 'mango' ]
for x in fruits             # second example
```

```
print 'current fruit : ', x print "good bye!"
```

Q. 8. Explain while loop with example. write a program on  $sum = 1 + 2 + 3 + \dots + n$  using while loop.

A. while loop :-

⇒ The while loop is one of the looping constructs available in python.

⇒ The while loop continues until the expression become false.

⇒ The expression has to be a logical expression and must return either a true or a false value

### EXAMPLE

```
* count = 0
while (count < 3):
    print 'the count is :',
    count = count + 1
    print "good bye!"
```

### output

the count is : 0

" : 1

" : 2

Good bye!

### Example : 2

$a = [1, 3, 5, 7]$

while a :

    print (a.pop(0)).

output
7
5
3
1

\*

### Program :

num = 16

if num < 0 :

    print ("Enter a Positive number")

else :

    sum = 0

    # use while loop to iterate until zero

    while (num > 0) :

        sum += num

        num -= 1

    print ("The sum is ", sum)

output

The sum is 136

Q. 9.

A.

How to use pass statement in python?

==>

The pass statement in python is used when a statement is required syntactically but you do not want any command or code to execute.

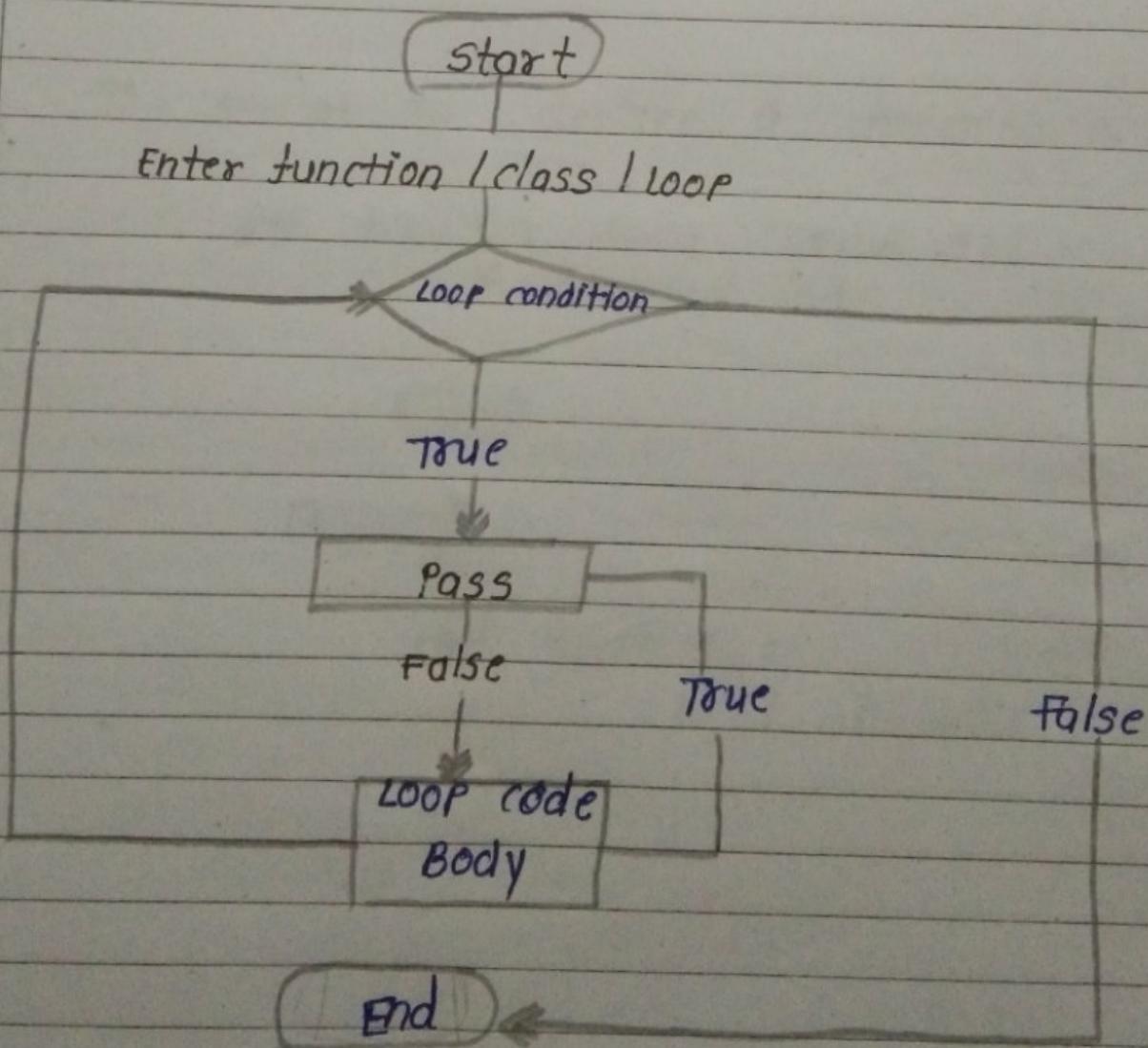
==>

The pass statement is a null operation; nothing happens when it executes.

==>

The pass statement is used as a placeholder for future code.

\*



Q. 10. How to declare function in python ? Explain with example.

A.

- ⇒ The actual body of the function can be defined separately.
  - ⇒ Add parameters to the function : They should be within the parentheses of the function.  
End your line with a colon.
  - ⇒ Add statements that the function should execute.
- \* The syntax to declare a function is :

def function-name (arguments):  
    # function body

return

EXAMPLE :-

def greet ():

    print ('Hello world!')

Q. 11. write a program in python to find whether a number is even or odd using function.

A.

# function to check a number Even or odd

```
def evenodd(num):
```

```
    if num % 2 == 0:
```

```
        print(num, " is an even number")
```

```
    else:
```

```
        print(num, " is an odd number")
```

```
evenodd(10).
```

output

\*

10 is an even number

Q. 12.

write a program in python to find fibonacci series using function.

Output like this: 0 1 1 2 3 5 8 13...

A.

```
nterms = int(input("How many terms?"))
```

```
# first two terms
```

```
n1, n2 = 0, 1
```

```
count = 0
```

```
# check if the number of terms is valid
```

```
if nterms <= 0:
```

```
    print("Please enter a positive integer")
```

# if there is only one term, return n1

elif nterms == 1:

print ("Fibonacci sequence upto ", nterms,":")

print (n1)

# generate fibonacci sequence

else :

print ('Fibonacci sequence: ")

while count < nterms :

print (n1)

nth = n1 + n2

# update values

n1 = n2

n2 = nth

count += 1

output

How many terms ? 7

Fibonacci sequence :

0

1

1

2

3

5

8

Q.13 write a program to perform pyramid pattern using range() function.

A.

1

1 2

1 2 3

1 2 3 4

Ans.

```
rows = int (input ("Enter number of rows : "))

for i in range (rows):
    for j in range (i+1):
        print (j+1, end = " ")
    print ('n').
```

Output

\* Enter number of rows : 4.

1

1 2

1 2 3

1 2 3 4

Q. 14.

Explain Recursion with example.

A.

==>

Python also accepts function recursion, which means a defined function can call itself.

```
def recurse():
```

...

```
    recurse()
```

...

```
recurse()
```

\* EXAMPLE :-

```
def factorial(x):
```

if x == 1 :

return 1

else :

return x \* factorial(x - 1)

num = 5

print ("The factorial of ", num, "is",  
 factorial(num)).

Output

The factorial of 5 is 120.

Q. 15. what is lambda function? Explain with example.

A.

- ⇒ A Lambda function in python programming is an anonymous function or a function having no name.
- ⇒ It is a small and restricted function having no more than one line.
- ⇒ Just like a normal function, a lambda function can have multiple arguments with one expression.

\* Syntax :-

Lambda arguments : expression

\* EXAMPLES :-

```
x = lambda a: a + 10  
print(x(5))
```

Output

15

\* multiply argument a with argument b and return the result.

```
x = lambda a, b: a * b  
print(x(5, 6))
```

Output

30

Q. 16. List out the characteristics of the list.

How to add and remove element in list explain with example.

A.

### I. The important characteristic of python list:-

- o index()
- o list comprehension
- o extend()
- o length of list
- o insert()
- o empty list []
- o remove()
- o pop()
- o clear()
- o count()
- o sort()
- o reverse()
- o copy()
- o slicing()
- o Nested list ()

#### \* list.append():

This method adds the elements at the end of the list.

\*

#### EXAMPLE :-

```
a = [25, "Anjali", 45, 217]
a.append("Jay")
print(a)
```

- list . extend () :

this method adds more than one element to the list.

EXAMPLE :-

```
b = [2, 45, 101]
b.extend(["Anjali", "Jay", "1830"])
print(b).
```

- \* Remove an element from the list.

- list . pop () :

this method removes the last elements from the list.

\* Example :-

```
d = [12, 34, 56, 78, 89, 0]
d.pop()
print(d).
```

- \* List . pop (index) :

```
e = [2, 4, 'Anjali', 76]
e.pop(2)
print(e)
```

Output

[2, 4, 76].

Q. 17. Explain `count()` and `index()` method in tuple.

A.

$\Rightarrow$  Tuple count () :-

method returns the number of times the specified element appears in the tuple.

EXAMPLE :-

```
vowels = ('a', 'e', 'i', 'o', 'i', 'u')
```

```
count = vowels.count('i')
```

```
print (count)
```

# output : 2

\* Tuple index () :-

The `index()` method finds the first occurrence of the specified value.

EXAMPLE :-

```
var = [1, 2, 3, 4, True, "Anjali"]
```

```
A = var.index("Anjali")
```

```
print (A).
```