**GE3151-Problem Solving and Python Programming**

**PART-A**

**UNIT I - ALGORITHMIC PROBLEM SOLVING**

**1. What is an algorithm?(Jan-2018)**

Algorithm is an ordered sequence of finite, well defined, unambiguous instructions for completing a task. It is an English-like representation of the logic which is used to solve the problem. It is a step-by-step procedure for solving a task or a problem. The steps must be ordered, unambiguous and finite in number.

**2. Write an algorithm to find minimum of three numbers.**

**ALGORITHM :** Find Minimum of three numbers

Step 1: Start

Step 2: Read the three numbers A, B, C

Step 3: Compare A,B and A,C. If A is minimum, perform step 4 else perform step 5.

Step 4:Compare B and C. If B is minimum, output “B is minimum” else output “C is minimum”

.Step 5: Stop

**3. List the building blocks of algorithm.**

The building blocks of an algorithm are

* Statements
* Sequence
* Selection or Conditional
* Repetition or Control flow
* Functions

An action is one or more instructions that the computer performs in sequential order (from first to last). A decision is making a choice among several actions. A loop is one or more instructions that the computer performs repeatedly.

**4. Define statement. List its types.**

The instructions in Python, or indeed in any high-level language, are designed as components for algorithmic problem solving, rather than as one-to-one translations of the underlying machine language instruction set of the computer. Three types of high-level programming language statements. Input/output statements make up one type ofstatement. An input statement collects a specific value from the user for a variable within the program. An output statement writes a message or the value of a program variable to the user’s screen.

**5. Write the pseudocode to calculate the sum and product of two numbers and display it.**

INITIALIZE variables sum, product, number1, number2 of type real

PRINT “Input two numbers”

READ number1, number2

COMPUTE sum = number1 + number2

PRINT “The sum is"

, sum

COMPUTE product = number1 \* number2

PRINT “The Product is"

, product

END program

**6. How does flow of control work?**

Control flow (or flow of control) is the order in which individual statements, instructions or function calls of an imperative program are executed or evaluated. A control flow statement is a statement in which execution results in a choice being made as to which of two or more paths to follow.

**7. Write the algorithm to calculate the average of three numbers and display it.**

Step 1: Start

Step 2: Read values of X,Y,Z

Step 3: S = X+Y+Z

Step 4: A = S/3

Step 5: Write value of A

Step 6: Stop

**8. Give the rules for writing Pseudocode.**

* Write one statement per line.
* Capitalize initial keywords.
* Indent to show hierarchy.
* End multiline structure.
* Keep statements language independent.

**9. What is a function?**

Functions are named sequence of statements that accomplish a specific task. Functions usually "take in" data, process it, and "return" a result. Once a function is written, it can be used over and over and over again. Functions can be "called" from the inside of other functions.

**10. Define a flowchart.**

* A flowchart is a diagrammatic representation of the logic for solving a task.
* A flowchart is drawn using boxes of different shapes with lines connecting them to show the flow of control.
* The purpose of drawing a flowchart is to make the logic of the program clearer in a visual form.

**UNIT II – DATA TYPES, EXPRESSIONS, STATEMENTS**

**1. What is meant by interpreter?**

An interpreter is a computer program that executes instructions written in a programming language. It can either execute the source code directly or translate the source code in a first step into a more efficient representation and executes this code.

**2. How will you invoke the python interactive interpreter?**

The Python interpreter can be invoked by typing the command "python" without any parameter followed by the "return" key at the shell prompt.

**3. What are the commands that are used to exit from the python interpreter in UNIX and windows?**

CTRL+D is used to exit from the python interpreter in UNIX and CTRL+Z is used to exit from the python

interpreter in windows.

**4. Define a variable and write down the rules for naming a variable.**

A name that refers to a value is a variable . Variable names can be arbitrarily long. They can contain both

letters and numbers, but they have to begin with a letter. It is legal to use uppercase letters, but it is good to begin variable names with a lowercase letter.

**5. Write a snippet to display “Hello World” in python interpreter.**

In script mode:

>>>print("Hello World")

Hello World

In Interactive Mode:

>>> "Hello World"

'Hello World'

**6. List down the basic data types in Python.**

* Numbers
* String
* List
* Tuple
* Dictionary

**7. Define keyword and enumerate some of the keywords in Python. (Jan-2019)**

A keyword is a reserved word that is used by the compiler to parse a program. Keywords cannot be used as variable names. Some of the keywords used in python are:

* and
* del
* from
* not
* while
* is
* continue

**8. What do you mean by an operand and an operator? Illustrate your answer with relevant example.**

An operator is a symbol that specifies an operation to be performed on the operands. The data items that an operator acts upon are called operands. The operators +, -, , / and \*\* perform addition, subtraction,

multiplication, division and exponentiation.

Example: 20+32.In this example, 20 and 32 are operands and + is an operator.

**9. Explain the concept of floor division.**

The operation that divides two numbers and chops off the fraction part is known as floor division.

Example:>>> 5//2= 2

**10. Define an expression with example.**

An expression is a combination of values, variables, and operators. An expression is evaluated using

assignment operator. Example:Y= X + 17

**UNIT III - CONTROL FLOW, FUNCTIONS, STRINGS**

**1. Define Boolean Expression with example.**

A boolean expression is an expression that is either true or false. The valuestrue and false are called Boolean values. The following examples use the operator “==” which compares two operands and produces True if they are equal and False otherwise:Example :>>> 5 == 6 False

True and False are special values that belongs to the type bool; they are not strings:

**2. What are the different types of operators?**

* Arithmetic Operator (+, -,

\*

, /, %,

\*\*

, // )

* Relational operator ( ==

, !=

, <>, < , > , <=

, >=)

* Assignment Operator ( =

, +=

,\*=

, - =

, /=

, %=

,

\*\*= )

* Logical Operator (AND, OR, NOT)
* Membership Operator (in, not in)
* Bitwise Operator (& (and), | (or) ,

^ (binary Xor),

~(binary 1’s complement , << (binary left shift), >>

(binary right shift))

* Identity(is, is not)

**3. Explain modulus operator with example**.

The modulus operator works on integers and yields the remainder when the first operand is divided by the second. In Python, the modulus operator is a percent sign (%). The syntax isthe same as for other operators:

Example:

>>> remainder = 7 % 3

>>> print remainder

1

So 7 divided by 3 is 2 with 1 left over.

4. Explain ‘for loop’ with example.

for loops are traditionally used when you have a block of code which you want to repeat a fixed number

of times. he Python for statement iterates over the members of a sequence in order, executing the

block each time.

The general form of a for statement is

Syntax:

Example:x = 4

for i in range(0, x):

print i

Output:0 1 2 3

**5. Explain relational operators**.

The == operator is one of the relational operators; the others are:

X! = y # x is not equal to y

x > y # x is greater than y

x < y # x is less than y

x >= y # x is greater than or equal to y

x <= y # x is less than or equal to y

**6. Explain while loop with example.(or)Explain flow of execution of while loop with Example.(Jan 2019)**

The statements inside the while loop is executed only if the condition is evaluated to true.

Syntax:

Example:

# Program to add natural numbers upto, sum = 1+2+3+...+10

n = 10

# initialize sum and counter

sum = 0

i = 1

while i <= n:

sum = sum + i

i = i+1 # update counter

for variable in sequence:

code block

while condition:

statements

# print the sum

print("The sum is"

, sum)

**7. Explain if-statement and if-else statement with example (or) What are conditional and alternative**

**executions?**

If statement:

The simplest form of if statement is:

Syntax:

Example:

if x > 0:

print 'x is positive'

The boolean expression after ‘if’ is called the condition. If it is true, then the indented statement gets

executed. If not, nothing happens.

If-else:

A second form of ifstatement is alternative execution, in which there are two possibilities and the condition

determines which one gets executed. The syntax looks like this:

Example:

if x%2 == 0:

print 'x is even'

else:

print 'x is odd'

If the remainder when x is divided by 2 is 0, then we know that x is even, and the program displays a

message to that effect. If the condition is false, the second set of statements is executed. Since the condition must be true or false, exactly one of the alternatives will be executed.

**8. What are chained conditionals?**

Sometimes there are more than two possibilities and we need more than two branches. One way to express

a computation like that is a chained conditional:

Eg:

if x < y:

print 'x is lessthan y'

elif x > y:

print 'x is greater than y'

else:

print 'x and y are equal'

elif is an abbreviation of “else if.

” Again, exactly one branch will be executed. There is no limit on the

number of elif statements. If there is an else clause, it hasto be at the end, but there doesn’t have to be one.

**9. What is a break statement?**

When a break statement is encountered inside a loop, the loop is immediately terminated and the program

control resumes at the next statement following the loop.

Eg:

while True:

line = raw\_input('>')

if line ==

'done':

break

print line

print'Done!'

if (condition):

statement

**10. What is a continue statement?**

The continue statement works somewhat like a break statement. Instead of forcing termination, it forces the

next iteration of the loop to take place, skipping any code in between.

Example:

for num in range(2,10):

if num%2==0;

print “Found an even number”

, num

continue

print “Found a number”

, num

**UNIT IV LISTS, TUPLES, DICTIONARIES**

**1. What is a list?(Jan-2018)**

A list is an ordered set of values, where each value is identified by an index. The values that make up a

list are called its elements. Lists are similar to strings, which are ordered sets of characters, except that the elements of a list can have any type.

**2. Relate String and List? (Jan 2018)(Jan 2019)**

String:

String is a sequence of characters and it is represented within double quotes or single quotes. Strings are immutable.

Example: s=”hello”

List:

A list is an ordered set of values, where each value is identified by an index. The values that make up a list are called its elements. Lists are similar to strings, which are ordered sets of characters, except that the elements of a list can have any type and it is mutable.

Example:

b= [’a’

,

’b’

,

’c’

,

’d’

, 1, 3]

**3. Solve a)[0] \* 4 and b) [1, 2, 3] \* 3.**

>>> [0] \* 4

[0, 0, 0, 0]

>>> [1, 2, 3] \* 3

[1, 2, 3, 1, 2, 3, 1, 2, 3]

**4. Let list = [’a’**

,

’b’

,

’c’

,

’d’

,

’e’

,

’f’]. Find a) list[1:3] b) t[:4] c) t[3:] .

>>> list = [’a’

,

’b’

,

’c’

,

’d’

,

’e’

,

’f’]

>>> list[1:3]

[’b’

,

’c’]

>>> list[:4]

[’a’

,

’b’

,

’c’

,

’d’]

>>> list[3:]

[’d’

,

’e’

,

’f’]

**5. Mention any 5 list methods.**

* append()
* extend ()
* sort()
* pop()
* index()
* insert
* remove()

**6. State the difference between lists and dictionary.**

Lists Dictionary

 List is a mutable type meaning that it

can be modified.

 List can store a sequence of objects in a

certain order.

 Example: list1=[1,

’a’

,

’apple’]

 Dictionary is immutable and is a key

value store.

 Dictionary is not ordered and it requires

that the keys are hashable.

 Example: dict1={‘a’:1,

‘b’:2}

**7. What is List mutability in Python? Give an example**.

Python represents all its data as objects. Some of these objects like lists and dictionaries are mutable, i.e., their content can be changed without changing their identity. Other objects like integers, floats, strings and tuples are objects that cannot be changed.

Example:

>>> numbers = [17, 123]

>>> numbers[1] = 5

>>> print numbers [17, 5]

**8. What is aliasing in list? Give an example.**

An object with more than one reference has more than one name, then the object is said to be aliased.

Example:If a refers to an object and we assign b = a, then both variables refer to the same object:

>>> a = [1, 2, 3]

>>> b = a

>>> b is a True

**9. Define cloning in list.**

In order to modify a list and also keep a copy of the original, it is required to make a copy of the list itself,

not just the reference. This process is sometimes called cloning, to avoid the ambiguity of the word

“copy”

.

Example:

def Cloning(li1):

li\_copy = li1[:]

return li\_copy

# Driver Code

li1 = [4, 8, 2, 10, 15, 18]

li2 = Cloning(li1)

print("Original List:"

, li1)

print("After Cloning:"

, li2)

Output:

Original List: [4, 8, 2, 10, 15, 18]

After Cloning: [4, 8, 2, 10, 15, 18]

**10. Explain List parameters with an example.**

Passing a list as an argument actually passes a reference to the list, not a copy of the list. For example, the

function head takes a list as an argument and returns the first element:

Example: def head(list):

return list[0]

Here’s how it is used:

>>> numbers = [1, 2, 3]

>>> head(numbers)

>>> 1

**UNIT V -FILES, MODULES, PACKAGES**

**1. What is a text file?**

A text file is a file that contains printable characters and whitespace, organized in to lines separated by

newline characters.

**2. Write a python program that writes “Hello world” into a file.**

f =open("ex88.txt"

,

'w')

f.write("hello world")

f.close()

**3. Write a python program that counts the number of words in a file.**

f=open("test.txt"

,

"r")

content =f.readline(20)

words =content.split()

print(words)

**4. What are the two arguments taken by the open() function?**

The open function takes two arguments : name of the file and the mode of operation.

Example: f = open("test.dat"

,

"w")

**5. What is a file object?**

A file object allows usto use, access and manipulate all the user accessible files. It maintains the state about

the file it has opened.

Example: f = open("test.dat"

,

"w") // f is the file object.

**6. What information is displayed if we print a file object in the given program?**

f= open("test.txt"

,

"w")

print f

The name of the file, mode and the location of the object will be displayed.

**7. What is an exception?**

Whenever a runtime error occurs, it creates an exception. The program stops execution and prints an error

message.

Example:

#Dividing by zero creates an exception:

print 55/0

ZeroDivisionError: integer division or modulo

**8. What are the two parts in an error message?**

The error message has two parts: the type of error before the colon, and specification about the error after

the colon.

Example:

>>> 10 \* (1/0)

Traceback (most recent call last):

File "<stdin>"

ZeroDivisionError: integer division or modulo by zero

**9. What are the error messages that are displayed for the following exceptions?**

a. Accessing a non-existent list item

b. Accessing a key that isn’t in the dictionary

c. Trying to open a non-existent file

a. IndexError: list index out of range

b. KeyError: what

c. IOError: [Errno 2] No such file or directory: 'filename'

**10. How do you handle the exception inside a program when you try to open a non-existent file?**

filename = raw\_input('Enter a file name: ')

try:

f = open (filename,

"r")

except IOError:

print 'There is no file named'

, filename