FIRST REVISION TEST FEB – 2022
12TH COMPUTER SCIENCE
ANSWER KEY

1 Marks:

1. A) Subroutines
2. A) Arguments
3. B) ( )
4. B) Selectors
5. C) Mapping
6. A) Local Scope
7. C) Algorithm
8. D) Selection Sort
9. A) Scope
10. C) Definition
11. B) Guido Van Rossum
12. C) Comma (,)
13. B) Relational
14. D) Assignment
15. A) Ternary

2 Marks:

16. Function with Respect to programming language:

A function is a unit of code that is oft en defined within a greater code structure. Specifically, a function contains a set of code that works on many kinds of inputs, like variants, expressions and produces a concrete output.

17. Abstract Data Type:

Abstract Data type (ADT) is a type (or class) for objects whose behaviour is defined by a set of value and a set of operations.

18. Namespaces:

Namespaces are containers for mapping names of variables to objects.

19. Types of Scopes of Variables:

1. Local Scope, 2. Global Scope, 3. Enclosed Scope, 4. Built – in Scope

20. Sorting:

Arranging the elements in certain order (ascending or descending) in a data structure is called sorting.

21. Algorithmic Solution:

An algorithm that yields expected output for a valid input is called an algorithmic solution.

22. Different modes in Python program:

1. Interactive Mode
2. Script Mode

23. Command to execute the Python Script:

Run --> Run Module or F5

24. Keyword:

Keywords are special words used by Python interpreter to recognize the structure of program. As these words have specific meaning for interpreter, they cannot be used for any other purpose.

3 Marks:

25. Parameter without type:

(requires: b>=0)
(returns: a to the power of b)
let rec pow a b:=
      if b=0 then 1
      else a * pow a (b-1)

In the above function definition if expression can return *1* in the then branch, by the ***typing*** rule the entire if expression has type ***int***. Since the if expression has type *'int',* the function's return type also be *'int'*. *'b'* is compared to *0* with the equality operator, so *'b'* is also a type of *'int'*. Since *'a'* is multiplied with another expression using the * operator, *'a'* must be an int.

26.

a) & e) – Constructor

b), c), d) & f) – Selector

27. Enclosed Scope:

All programming languages permit functions to be nested. A function (method) with in another function is called nested function. *A variable which is declared inside a function which contains another function definition with in it, the inner function can also access the variable of the outer function. This scope is called enclosed scope.*

	Entire program	Output of the Program
1. Disp():	Disp( )	10
2. a:=10	a:=10	
3. Disp1():	Disp 1( ):	10
4. print a	print a	
5. Disp1()	Disp 1( ):	
6. print a	print a	
7. Disp()	Disp( )	

28. Algorithm and Program:

Algorithm	Program
• Algorithm helps to solve a given problem logically and it can be contrasted with the program	• Program is an expression of algorithm in a programming language
• Algorithm can be categorized based on their implementation methods, design techniques etc	• Algorithm can be implemented by structured or object oriented programming approach
• There is no specific rules for algorithm writing but some guidelines should be followed.	• Program should be written for the selected language with specific syntax
• Algorithm resembles a pseudo code which can be implemented in any language	• Program is more specific to a programming language

29. Pseudo code for Linear Search:

- Traverse the array using for loop
- In every iteration, compare the target search key value with the current value of the list.
- If the values match, display the current index and value of the array
- If the values do not match, move on to the next array element.
- If no match is found, display the search element not found.

30. Escape Sequence:

In Python strings, the backslash "\" is a special character, also called the **"escape"** character. It is used in representing certain whitespace characters: **"\t"** is a tab, **"\n"** is a newline, and **"\r"** is a carriage return.

For example to print the message "It's raining", the Python command is

>>> print ("It\'s rainning")

It's rainning

31. **input():**

input( ) function is used to accept data as input at run time.

Syntax:

variable = input ("prompt string")

The **input( )** takes whatever is typed from the keyboard and stores the entered data in the given variable. If prompt string is not given in **input( )** no message is displayed on the screen, thus, the user will not know what is to be typed as input.

>>> city=input ("Enter Your City: ")

Enter Your City: Madurai
>>> print ("I am from ", city)
I am from Madurai

32. **Ternary Operator:**

Ternary operator is also known as conditional operator that evaluate something based on a condition being true or false. It simply allows testing a condition in a single line replacing the multiline if-else making the code compact.

Syntax: *Variable Name = [on_true] if [Test expression] else [on_false]*

Example: min= 50 if 49<50 else 70

Output: min = 50

33. **Output:**

Enter Number1: 10

Enter Number2: 20

The sum= 30

5 Marks:

34. **A) Data abstraction:**

To facilitate data abstraction, you will need to create two types of functions: constructors and selectors

Constructors are functions that build the abstract data type
Selectors are functions that retrieve information from the data type.

For example, say you have an abstract data type called city. This city object will hold the city's name, and its latitude and longitude. To create a city object, you'd use a function like

city = makecity (name, lat, lon)

To extract the information of a city object, you would use functions like

getname(city)

getlat(city)

getlon(city)

Selectors are nothing but the functions that retrieve information from the data type. Therefore in the above code

      getname(city)
      getlat(city)
      getlon(city)

are the selectors because these functions extract the information of the city object

b)

    i)   Scope should be used for variable. Because every part of program can access the variable.

   ii)   Global Scope:

        A variable which is declared outside of all the functions in a program is known as global variable. This means, global variable can be accessed inside or outside of all the functions in a program

1. a:=10	Entire program	Output of the Program

2. Disp():	a:=10
	  Disp( )
	    a:=7
	    print a
		7

3. a:=7		10
4. print a	Disp 1( ): print a	
5. Disp()		
6. print a		

35. a) Characteristics of an algorithm:

Input	Zero or more quantities to be supplied.
Output	At least one quantity is produced.
Finiteness	Algorithms must terminate after finite number of steps.
Definiteness	All operations should be well defined. For example operations involving division by zero or taking square root for negative number are unacceptable.
Effectiveness	Every instruction must be carried out effectively.
Correctness	The algorithms should be error free.
Simplicity	Easy to implement.
Unambiguous	Algorithm should be clear and unambiguous. Each of its steps and their inputs/outputs should be clear and must lead to only one meaning.
Feasibility	Should be feasible with the available resources.
Portable	An algorithm should be generic, independent of any programming language or an operating system able to handle all range of inputs.
Independent	An algorithm should have step-by-step directions, which should be independent of any programming code.

B) Binary Search:

Binary search also called half-interval search algorithm It finds the position of a search element within a sorted array. The binary search algorithm can be done as divide-and-conquer search algorithm and executes in logarithmic time.

Pseudo code for Binary search

1. Start with the middle element:

If the search element is equal to the middle element of the array i.e., the middle value = number of elements in array/2, then return the index of the middle element.

If not, then compare the middle element with the search value,

If the search element is greater than the number in the middle index, then select the elements to the right side of the middle index, and go to Step-1.

If the search element is less than the number in the middle index, then select the elements to the left side of the middle index, and start with Step-1.

2. When a match is found, display success message with the index of the element matched.

3. If no match is found for all comparisons, then display unsuccessful message.

Example:

```
10 20 30 40 50 60 70 80 90 99
 0  1  2  3  4  5  6  7  8  9
```

First, we find index of middle element of the array by using this formula :

$$mid = low + (high - low) / 2$$

Here it is, $0 + (9 - 0) / 2 = 4$ (fractional part ignored). So, 4 is the mid value of the array.

```
60 70 80 90 99
 5  6  7  8  9
```

$$mid = low + (high - low) / 2$$

$$mid = 7$$

```
60 70
 5  6
```

$$mid = low + (high - low) / 2$$

$$mid = 5$$

36. Tokens:

Python breaks each logical line into a sequence of elementary lexical components known as Tokens. The normal token types are

1) Identifiers,
2) Keywords,
3) Operators,
4) Delimiters and
5) Literals.

1. Identifiers

An Identifier is a name used to identify a variable, function, class, module or object.

Valid: Sum, total_marks, regno, num1

Invalid : 2Name, name$, total-mark, continue

2. Keywords :

Keywords are special words used by Python interpreter to recognize the structure of program. As these words have specific meaning for interpreter, they cannot be used for any other purpose.

Example: if , for, while false , true

3 . Operators:

In computer programming languages operators are special symbols which represent computations, conditional matching etc. The value of an operator used is called operands. Operators are categorized as Arithmetic, Relational, Logical, Assignment etc. Value and variables when used with operator are known as operands.

(i) Arithmetic operators

An arithmetic operator is a mathematical operator that takes two operands and performs a calculation on them. They are used for simple arithmetic. Most computer languages contain a set of such operators that can be used within equations to perform different types of sequential calculations.

(ii) Logical operators

In python, Logical operators are used to perform logical operations on the given relational expressions. There are three logical operators they are and, or and not. Operator

iii) Relational Operators:

A Relational operator is also called as Comparative operator which checks the relationship between two operands. If the relation is true, it returns True; otherwise it returns False.

(iv) Assignment operators

In Python, = is a simple assignment operator to assign values to variable. Let a = 5 and b = 10 assigns the value 5 to a and 10 to b these two assignment statement can also be given as a,b=5,10 that assigns the value 5 and 10 on the right to the variables a and b respectively. There are various compound operators in Python like +=, -=, *=, /=, %=, **= and //= are also available.

(v) Conditional operator

Ternary operator is also known as conditional operator that evaluate something based on a condition being true or false. It simply allows testing a condition in a single line replacing the multiline if-else making the code compact.
Example:
min= 50 if 49<50 else 70 # min = 50
min= 50 if 49>50 else 70 # min = 70

4. Delimiters

Python uses the symbols and symbol combinations as delimiters in expressions, lists, dictionaries and strings (,) , { , }, [,], , ,= and so on.

5. Literals

Literal is a raw data given in a variable or constant. In Python, there are various types of literals.
1) Numeric 2) String 3) Boolean

(i) Numeric Literals

Numeric Literals consists of digits and are immutable (unchangeable). Numeric literals can belong to 3 different numerical types Integer, Float and Complex.

(ii) String Literals

In Python a string literal is a sequence of characters surrounded by quotes. Python supports single, double and triple quotes for a string A character literal is a single character surrounded by single or double quotes. The value with triple-quote "' '" is used to give multi-line string literal.

(iii) Boolean Literals

A Boolean literal can have any of the two values: True or False.

B)

1) Commands:

In Python, comments begin with hash symbol (#). The lines that begins with # are considered as comments and ignored by the Python interpreter. Comments may be single line or no multi-lines The multiline comments should be enclosed within a set of # as given below.

It is Single line Comment

''' It is multiline comment

which contains more than one line '''

2.Indentation:

Python uses whitespace such as **spaces** and **tabs** to define program blocks whereas other languages like C, C++, java use curly braces { } to indicate blocks of codes for class, functions or body of the loops and block of selection command. The number of whitespaces (spaces and tabs) in the indentation is not fixed, but all statements within the block must be indented with same amount spaces.

37. A) Insertion Sort:

Insertion sort is a simple sorting algorithm. It works by taking elements from the list one by one and inserting then in their correct position in to a new sorted list. This algorithm builds the final sorted array at the end. This algorithm uses n-1 number of passes to get the final sorted list as per the pervious algorithm as we have discussed.

Pseudo Code:

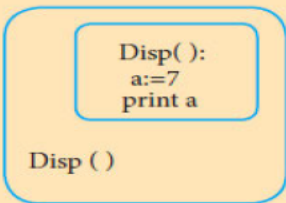Step 1 – If it is the first element, it is already sorted.

Step 2 – Pick next element

Step 3 – Compare with all elements in the sorted sub-list Step 4 – Shift all the elements in the sorted sub-list that is greater than the value to be sorted Step 5 – Insert the value Step 6 – Repeat until list is sorted

| 44	16	83	07	67	21	34	45	10	Assume 44 is a soted list of 1 item
16	44	83	07	67	21	34	45	10	inserted 16
16	44	83	07	67	21	34	45	10	inserted 83
07	16	44	83	67	21	34	45	10	inserted 07
07	16	44	67	83	21	34	45	10	inserted 67
07	16	21	44	67	83	34	45	10	inserted 21
07	16	21	34	44	67	83	45	10	inserted 34
07	16	21	34	44	45	67	83	10	inserted 45
07	10	16	21	34	44	45	67	83	inserted 10

37 B). Local Scope:

Local scope refers to variables defined in current function. Always, a function will first look up for a variable name in its local scope. Only if it does not find it there, the outer scopes are checked.

1. Disp(): 2. a:=7 3. print a 4. Disp()	Entire program Disp( ): a:=7 print a Disp ( )	Output of the Program 7

Built-in Scope:

Finally, we discuss about the widest scope. The built-in scope has all the names that are pre-loaded into the program scope when we start the compiler or interpreter. Any variable or module which is defined in the library functions of a programming language has Built-in or module scope. They are loaded as soon as the library files are imported to the program.
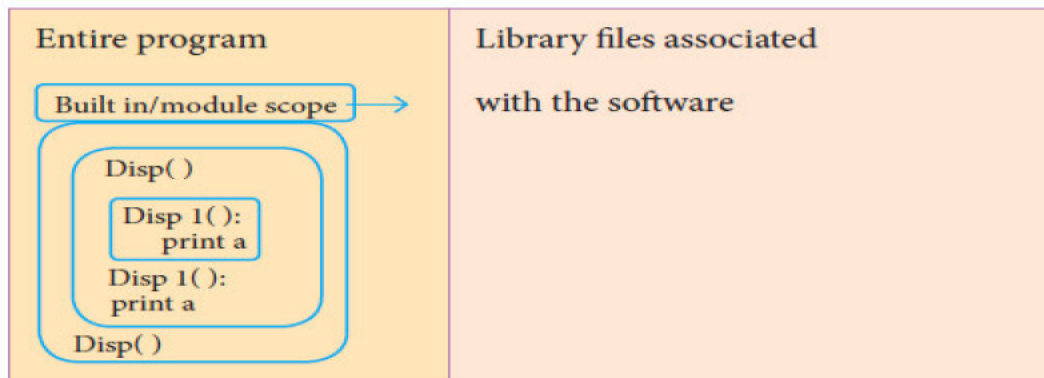
38. A) Different Types of Operators:

Operators:

In computer programming languages operators are special symbols which represent computations, conditional matching etc. The value of an operator used is called operands. Operators are categorized as Arithmetic, Relational, Logical, Assignment etc. Value and variables when used with operator are known as operands.

(i) Arithmetic operators

An arithmetic operator is a mathematical operator that takes two operands and performs a calculation on them. They are used for simple arithmetic. Most computer languages contain a set of such operators that can be used within equations to perform different types of sequential calculations.

Arithmetic Operators: +,-,*,/,%,//,**

(ii) Logical operators

In python, Logical operators are used to perform logical operations on the given relational expressions. There are three logical operators they are and, or and not Operator.

iii) Relational Operators:

A Relational operator is also called as Comparative operator which checks the relationship between two operands. If the relation is true, it returns True; otherwise it returns False.

Relational Operators: <,>,<=,>=,==,!=.

(iv) Assignment operators

In Python, = is a simple assignment operator to assign values to variable. Let a = 5 and b = 10 assigns the value 5 to a and 10 to b these two assignment statement can also be given as a,b=5,10 that assigns the value 5 and 10 on the right to the variables a and b respectively. There are various compound operators in Python like +=, -=, *=, /=, %=, **= and //= are also available.

(v) Conditional operator

Ternary operator is also known as conditional operator that evaluate something based on a condition being true or false. It simply allows testing a condition in a single line replacing the multiline if-else making the code compact.

Example:

min= 50 if 49<50 else 70 # min = 50
min= 50 if 49>50 else 70 # min = 70

38 Output:

The Sum = 110
The Difference = 90
The Product = 1000
The Quotient = 10 0
The Remainder = 10
The Exponent = 10000
The Floor Division = 3

Prepared by
PARKUNAN T
PG Assistant in Computer Science,
MHSS, Chengam-606 701
Cell: 9655 966 906.