

**COMMON QUARTERLY EXAMINATION 2024 CUDDALORE - DT**  
**STANDARD -XII**  
**COMPUTER SCIENCE**

TIME:3.00HRS

MARKS:70

I ANSWER ALL THE QUESTIONS

15X1=15

Q.NO	OPTION	ANSWER	Q.NO	OPTION	ANSWER
1.	d.	Parameters	9.	a.	Lambda
2.	c.	Tuple	10.	a.	+
3.	a.	Public members	11.	d.	Either (a) or (b)
4.	d.	Half – interval Search	12.	b.	[17,23,41,10,32]
5.	b.	Guido Van Rossum	13.	d.	{1,3,6,9}
6.	a.	Interpreter	14.	c.	Methods
7.	a.	3	15.	d.	__init__()
8.	b.	:			

**PART -II**

**II Answer any six questions.( Question No.24 is compulsory)**

**6x2=12**

16.	<b>Interface</b>	<b>Implementation</b>	2
	Interface just defines what an object can do, but won't actually do it	Implementation carries out the instructions defined in the interface	
17.	A tuple is a comma-separated sequence of values surrounded with parentheses. <b>Example:</b> Color= ('red', 'blue', 'Green')		1 1
18.	The process of binding a variable name with an object is called mapping.		2
19.	<ul style="list-style-type: none"> <li>Sorting is a process of arranging group of items in an ascending or descending order. Bubble Sort, Quick Sort, Heap Sort, Merge Sort, Selection Sort are the various sorting algorithms.</li> </ul>		2
20.	<ul style="list-style-type: none"> <li>Literal is a raw data given in a variable or constant.</li> <li>In Python, there are various types of literals. They are,               <ol style="list-style-type: none"> <li>1) <b>Numeric Literals</b> consists of digits</li> <li>2) <b>String literal</b> is a sequence of characters surrounded by quotes.</li> <li>3) <b>Boolean literal</b> can have any of the two values: True or False.</li> </ol> </li> </ul>		2
21.	<ul style="list-style-type: none"> <li>The <b>break</b> statement terminates the loop containing it.</li> <li>Control of the program flows to the statement immediately after the body of the loop.</li> </ul>		2
22.	<ul style="list-style-type: none"> <li>Functions are named blocks of code that are designed to do one specific job. (or) Function blocks begin with the keyword "def" followed by function name and parenthesis ().</li> </ul>		2
23.	A substring can be taken from the original string by using [ ] slicing operator and index or subscript values. Using slice operator, you have to slice one or more substrings from a main string.		2

24.	6	2
-----	---	---

## PART -II

II Answer any six questions.( Question No.33 is compulsory)

6x3=18

25.	<ul style="list-style-type: none"> <li>The class template specifies the interfaces to enable an object to be created and operated properly.</li> <li>An object's attributes and behaviour is controlled by sending functions to the object.</li> </ul>	3						
26.	<table border="1"> <thead> <tr> <th>Concrete data types</th> <th>Abstract Data Types</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>Concrete data types or structures (CDT's) are direct implementations of a relatively simple concept.</li> </ul> </td> <td> <ul style="list-style-type: none"> <li>Abstract Data Types (ADT's) offer a high level view (and use) of a concept independent of its implementation.</li> </ul> </td> </tr> <tr> <td> <ul style="list-style-type: none"> <li>A concrete data type is a data type whose representation is known.</li> </ul> </td> <td> <ul style="list-style-type: none"> <li>Abstract data type the representation of a data type is unknown.</li> </ul> </td> </tr> </tbody> </table>	Concrete data types	Abstract Data Types	<ul style="list-style-type: none"> <li>Concrete data types or structures (CDT's) are direct implementations of a relatively simple concept.</li> </ul>	<ul style="list-style-type: none"> <li>Abstract Data Types (ADT's) offer a high level view (and use) of a concept independent of its implementation.</li> </ul>	<ul style="list-style-type: none"> <li>A concrete data type is a data type whose representation is known.</li> </ul>	<ul style="list-style-type: none"> <li>Abstract data type the representation of a data type is unknown.</li> </ul>	3
	Concrete data types	Abstract Data Types						
	<ul style="list-style-type: none"> <li>Concrete data types or structures (CDT's) are direct implementations of a relatively simple concept.</li> </ul>	<ul style="list-style-type: none"> <li>Abstract Data Types (ADT's) offer a high level view (and use) of a concept independent of its implementation.</li> </ul>						
<ul style="list-style-type: none"> <li>A concrete data type is a data type whose representation is known.</li> </ul>	<ul style="list-style-type: none"> <li>Abstract data type the representation of a data type is unknown.</li> </ul>							
27.	1.Input 2. Output 3. Finiteness 4.Definiteness 5. Effectiveness 6.Correctness 7. Simplicity 8.Unambiguous 9. Feasibility 10.Portable 11.Independent	3						
28.	<ul style="list-style-type: none"> <li>. Ternary operator is also known as <b>conditional operator</b> that evaluates something based on a condition being true or false.</li> <li>It simply allows testing a condition in a single line replacing the multiline if-else making the code compact.</li> </ul> <p><b>Syntax:</b> <i>Variable Name = [on_true] if [Test expression] else [on_false]</i></p> <p><b>Example :</b> <code>min = 50 if 49&lt;50 else 70</code> # Output: min = 50</p>	2						
29.	<p><b>Syntax:</b></p> <pre>while &lt;condition&gt;:     statement block1 [else:     Statement block2]</pre>	3						
30.	<ul style="list-style-type: none"> <li>Functions that calls itself is known as recursive.</li> <li>When a function calls itself is known as recursion.</li> <li>Recursion works like loop but sometimes it makes more sense to use recursion than loop.</li> <li>Imagine a process would iterate indefinitely if not stopped by some condition is known as infinite iteration.</li> <li>The condition that is applied in any recursive function is known as base condition.</li> <li>A base condition is must in every recursive function otherwise it will continue to execute like an infinite loop.</li> <li>Python stops calling recursive function after certain limit by default. So, It also allows you to change the limit using <code>sys.setrecursionlimit (limit_value)</code>.</li> </ul>	3						
31.	<p><b>Count:</b></p> <ul style="list-style-type: none"> <li>Returns the number of substrings occurs within the given range.</li> <li>Remember that substring may be a single character.</li> <li>Range (beg and end) arguments are optional. If it is not given, python searched in whole string.</li> <li>Search is case sensitive.</li> </ul> <p><b>SYNTAX:</b> <code>count(str, beg, end)</code></p> <p><b>Explain with Example</b></p>	2						
			1					



➤  $x$  will become 10 and  $y$  will become 20.

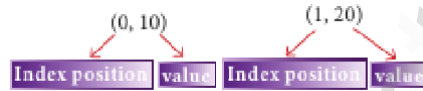
## 2. Element Selection Operator:

- It is expressed using square brackets.
- Unlike a list literal, a square-brackets expression directly following another expression does not evaluate to a list value, but instead selects an element from the value of the preceding expression.

### PAIRS:

- Any way of bundling two values together into one can be considered as a pair.
- Python provides a compound structure called Pair which is made up of list or Tuple.

**Example:** `lst:=[(0,10),(1,20)]`



35a.

The following are the desirable characteristics of a module.

1. Modules contain instructions, processing logic, and data.
2. Modules can be separately compiled and stored in a library.
3. Modules can be included in a program.
4. Module segments can be used by invoking a name and some parameters.
5. Module segments can be used by other modules.

OR

35b.

- Bubble sort is a simple sorting algorithm, it starts at the beginning of the list of values stored in an array.
- It compares each pair of adjacent elements and swaps them if they are in the unsorted order.
- This comparison and passed to be continued until no swaps are needed, which shows the values in an array is sorted.
- It is named so because, the smaller elements "bubble" to the top of the list.
- It is too slow and less efficient when compared to other sorting methods.

### **Pseudo code**

1. Start with the first element i.e., index = 0, compare the current element with the next element of the array.
2. If the current element is greater than the next element of the array, swap them.
3. If the current element is less than the next or right side of the element, move to the next element.
4. Go to Step 1 and repeat until end of the index is reached.

### **Explain with example**

36a.

- 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.

### **Explain with example**

OR

- 36b.
- When we need to construct a chain of **if** statement(s) then '**elif**' clause can be used instead of '**else**'.
  - '**elif**' clause combines **if..else-if..else** statements to one **if..elif...else**.
  - '**elif**' can be considered to be abbreviation of '**else if**'.
  - In an '**if**' statement there is no limit of '**elif**' clause that can be used, but an '**else**' clause if used should be placed at the end.
  - In the syntax of **if..elif..else** mentioned above, condition-1 is tested if it is true then statements-block1 is executed.
  - Otherwise the control checks condition-2, if it is true statements-block2 is executed and even if it fails statements-block n mentioned in **else** part is executed.

**Explain with example**

- 37a.
- Functions are named blocks of code that are designed to do one specific job.
  - **Types of Functions**
    - User defined Function
    - Built-in Function
    - Lambda Function
    - Recursion Function
- i) BUILT-IN FUNCTION:**
- Built-in functions are Functions that are inbuilt with in Python.
  - print(), echo() are some built-in function.
- ii) USER DEFINED FUNCTION:**
- Functions defined by the users themselves are called user defined function.
  - Functions must be defined, to create and use certain functionality.
  - Function blocks begin with the keyword "def " followed by function name and parenthesis ().
- iii) LAMBDA FUNCTION:**
- In Python, anonymous function is a function that is defined without a name.
  - While normal functions are defined using the **def** keyword, in Python anonymous functions are defined using the **lambda** keyword.
  - Hence, anonymous functions are also called as **lambda** functions.
- USE OF LAMBDA OR ANONYMOUS FUNCTION:**
- Lambda function is mostly used for creating small and one-time anonymous function.
- iv) RECURSIVE FUNCTION:**
- Functions that calls itself is known as recursive.
- Overview of how recursive function works**
1. Recursive function is called by some external code.
  2. If the base condition is met then the program gives meaningful output and exits.
  3. Otherwise, function does some required processing and then calls itself to continue recursion.

**Explain with example**

OR



	<p>object exit from the scope.</p> <ul style="list-style-type: none"><li>❖ <code>__del__()</code> method is used as destructor.</li></ul> <p><b><u>General format:</u></b></p> <pre>def __del__(self):     &lt;statements&gt;</pre> <p><b>Explain with example</b></p>	1
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---

www.Padasalai.Net