HSE II YEAR PUBLIC EXAM MARCH 14 2025

TENTATIVE ANSWER KEY

COMPUTER SCIENCE

PART I

1. C) DEFINITION
2. A) CONSTRUCTORS
3. B) LEGB
4. C) HALF INTERVEL SEARCH
5. B) TERNARY
6. A) FOR
7. D) RETURN
8. A) POSIIVE OR NEGAIVE NUMBERS
9. A) ecneicS retupmoC
10. D) .
11. B) CHEN
12. B) SELECT
13. A) MODIFICATION
14. B) OS MODULE
15. B) DISTINCT

PART II

16. What is a Pair? Give an example.

Any way of bundling two values together into one can be considered as a pair. Example: lst[(0,10),(1,20)], Here, lst[0] = 10, lst[1] = 20

17. What is sorting

Sorting is a process of arranging group of items in an ascending or descending order.

Types: Bubble Sort, Selection Sort, Insertion sort.

18. 5 10 15

19. What is slicing

Slice is a substring of a main string.

A substring can be taken from the original string by using [ ] slicing operator and index or subscript values.

20. What is the purpose of Destructor

Destructor is also a special method gets executed automatically when an object exit from the scope.

In Python, __del__( ) method is used as destructor.

21. Which component of SQL to insert values in tables and which lets to create a table

To create a Table : Data Definition Language

Create Table Table name

To insert a record : Data Manipulation Language

Insert into

22. What is the use of CD command? Give example

'cd' command used to change directory and absolute path refers to the complete path where Python is installed.

Syntax: cd <absolute path>

Example: c:\>cd c:\ program files \ openoffice 4 \ program

23. How will you install matplotlib

Matplotlib can be installed using pip software.

Pip is a Package manager software for installing python packages.

24. False

welcome

PART III

25. The variables used inside the function may cause side effects though the functions which are not passed with any arguments. In such cases the function is called impure function.

When a function depends on variables or functions outside of its definition block, you can never be sure that the function will behave the same every time it's called. Example: random() function.

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

```
Example:    a = 10
            def inner():
                    a = 20
             print(a)
            inner()
            print(a)
```


27. What are the assignment operators that can be used in Python

Operator	Description	Example
Assume x=10		
=	Assigns right side operands to left variable	>>> x=10 >>> b="Computer"
+=	Added and assign back the result to left operand	>>> x+=20 # x=x+20
-=	Subtracted and assign back the result to left operand	>>> x-=5 # x=x-5
*=	Multiplied and assign back the result to left operand	>>> x*=5 # x=x*5
/=	Divided and assign back the result to left operand	>>> x/=2 # x=x/2

28. Differentiate ceil() and floor() function

Differentiate ceil() and floor() function?

ceil()	floor()
Returns the smallest integer greater than or equal to x.	Returns the largest integer less than or equal to x.
Syntax: math.ceil (x)	Syntax: math.floor (x)
Ex: >>>import math >>>print(math.ceil(26.7)) Output: 27 >>>Print(math.ceil(-26.7)) Output: -26	Ex: >>>import math >>>print(math.floor(26.7)) Output: 26 >>>Print(math.floor(-26.7)) Output: -27

29. Write the syntax of the following Python functions

I)remove ii)pop() iii)clear

Syntax:

List.remove(element)

List.pop(index of an element)

List.clear()

30.

Explain Cartesian Product with a suitable example.

- Cross product is a way of combining two relations.
- The resulting relation contains, both relations being combined.
- This type of operation is helpful to merge columns from two relations.

Example: A x B means A times B, where the relation A and B have different attributes.

31. Write the difference between reader and dictReader()

What is the difference between reader() and DictReader() function?

Reader():

- The reader function is designed to take each line of the file and make a list of all columns.
- Using this method one can read data from csv files of different formats like quotes (" "), pipe (|) and comma (,).
- csv. Reader work with list/tuple.**DictReader():**
- DictReader works by reading the first line of the CSV and using each comma separated value in this line as a dictionary key.
- DictReader is a class of csv module is used to read a CSV file into a dictionary.
- It creates an object which maps data to a dictionary.
- csv.DictReader work with dictionary.

32. 2. Mention the difference between fetchone() and fetchmany()

fetchone()	fetchmany()
• The **fetchone()** method returns the next row of a query result set or None in case there is no row left	• The **fetchmany()** method returns the next number of rows (n) of the result set.
• Using while loop and fetchone() method we can display all the records from a table.	• Displaying specified number of records is done by using **fetchmany().**

33.

```
>>> [1, 4, 9, 16, 25]
```

PART IV

34.

Explain with example Pure and impure functions.

Pure Function	Impure Function
• Pure functions will give exact result when the same arguments are passed.	• Impure functions never assure you that the function will behave the same every time it's called.
• Pure function does not cause any side effects to its output.	• Impure function causes side effects to its output.
• The return value of the pure functions solely depends on its arguments passed.	• The return value of the impure functions does not solely depend on its arguments passed.
• They do not modify the arguments which are passed to them	• They may modify the arguments which are passed.
• If we call pure functions with same set of arguments, we will always get the same return values.	• If we call impure functions with same set of arguments, we might get the different return values.
Example: sqrt() let square x:= return: x * x	• **Example: random()** let random number:= a := random() if a > 10 then return: a

OR

b) A data model describes how the data can be represented and accessed from a software after complete implementation

Hierarchical Model

Relational Model

Network Database Model

Entity Relationship Model

Object Model

i). Hierarchical Model:

In Hierarchical model, data is represented as a simple tree like structure form.

This model represents a one-to-many relationship ie parent-child relationship.

One child can have only one parent but one parent can have many children.

This model is mainly used in IBM Main Frame computers.

ii). Relational Model

The Relational Database model was first proposed by E.F. Codd in 1970 .

The basic structure of data in relational model is tables (relations).

All the information's related to a particular type is stored in rows of that table.

Hence tables are also known as relations in a relational model.

A relation key is an attribute which uniquely identifies a particular tuple (row in a relation (table))

iii.) Network Model

Network database model is an extended form of hierarchical data model.

In a Network model, a child may have many parent nodes.

It represents the data in many-to-many relationships.

This model is easier and faster to access the data.

In this database model, relationship are created by dividing the object into entity and its characteristics into attributes.

It was developed by Chen in 1976.

ER model constructed by,

Rectangle represents the entities.

Ellipse represents the attributes .

Attributes describes the characteristics and each entity.

Diamond represents the relationship in ER diagrams

Example: Doctor diagnosis the Patient

v.)Object Model

Object model stores the data in the form of objects, attributes and methods, classes and Inheritance.

This model handles more complex applications, such as Geographic information System (GIS), scientific experiments, engineering design and manufacturing.

35. Type of Constraints:

Unique Constraint: This constraint ensures that no two rows have the same value in the specified columns.

Primary Key Constraint: This constraint declares a field as a Primary key which helps to uniquely identify a record. The primary key does not allow NULL values and therefore a field declared as primary key must have the NOT NULL constraint.

Default constraint: This constraint is used to assign a default value for the field. When no value is given for the specified field having DEFAULT constraint, automatically the default value will be assigned to the field.

Check Constraint: This constraint helps to set a limit value placed for a field. When we define a check constraint on a single column, it allows only the restricted values on that field.

Table Constraint: When the constraint is applied to a group of fields of the table, it is known as Table constraint. The table constraint is normally given at the end of the table definition

Eg:

CREATE TABLE Student(Admno integer NOT NULL PRIMARY KEY,Exam no integer NOT NULL UNIQUE,Name char(20)NOT NULL,Gender char(1) DEFAULT = "M", Age integer (CHECK<=19));

OR

Explain the different types of operators used in Python. (Jul 2022, PTA–1)

Arithmetic operators:

An arithmetic operator is a mathematical operator that takes two operands and performs a calculation on them.

Operator (Operation)	Examples	Result
+ (Addition)	100 + 10	**110**
- (Subtraction)	100 – 10	**90**
* (Multiplication)	100 * 10	**1000**
/ (Division)	10 / 3	**3.3333**
% (Modulus)	10 % 3	**1**
** (Exponent)	10 ** 2	**100**
// (Floor Division)	10 // 3 (Integer division)	**3**

Relational or Comparative 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.

Operator (Operation)	Examples	Result
== (is Equal)	5==5	**True**
> (Greater than)	5 > 4	**True**
< (Less than)	5 < 4	**False**
>= (Greater than or Equal to)	5 >= 4	**True**
<= (Less than or Equal to)	5 <= 4	**False**
!= (Not equal to)	5 != 4	**True**

36.

Explain while loop with example. (PTA–4)

while loop is entry check loop. The condition is placed in the beginning of the body of the loop. The statements in the loop will not be executed even once if the condition is false at the time of entering the loop.

Syntax:

```
while <condition>:
        statements block 1
[else:
        statements block2]
```

The **statements block1** is kept executed till the condition is True. If the **else** part is written, it is executed when the condition is tested False.

OR

2. Tabulate the different mode with its meaning.

Python File Modes:

Mode	Description
'r'	• Open a file for reading. (default)
'w'	• Open a file for writing. Creates a new file if it does not exist or truncates the file if it exists.
'x'	• Open a file for exclusive creation. If the file already exists, the operation fails.
'a'	• Open for appending at the end of the file without truncating it. Creates a new file if it does not exist.
't'	• Opren in text mode. (default)
'b'	• Open in binary mode.
'+'	• Open a file for updating (reading and writing)

OR

37. Explain recursive function with an example.

Functions that calls itself is known as recursive. When a function calls itself is known as recursion.Recursion works like loop but sometimes it makes more sense to use recursion than loop.Imagine a process would iterate indefinitely if not stopped by some condition is known as infinite iteration. The condition that is applied in any recursive function is known as base condition. A base condition is must in every recursive function otherwise it will continue to execute like an infinite loop.

Python stops calling recursive function after certain limit by default. So, It also allows you to change the limit using sys.setrecursionlimit (limit_value).

Overview of how recursive function works:

Recursive function is called by some external code.
If the base condition is met then the program gives meaningful output and exits.
Otherwise, function does some required processing and then calls itself to continue recursion.

EXAMPLE:

```
def fact(n):
        if n == 0:
                return 1
        else:
                return n * fact (n-1)
print (fact (0))
print (fact (5))
Output:     1
            120
```

OR

Syntax of getopt method:

<opts>,<args>=getopt.getopt(argv, options, [long_options])

- Here is the detail of the parameters −
- **argv** -- This is the argument list of values to be parsed (splited). In our program the complete command will be passed as a list.
- **options** -- This is string of option letters that the Python program recognize as, for input or for output, with options (like 'i' or 'o') that followed by a colon (:). Here colon is used to denote the mode.
- **long_options** -- This parameter is passed with a list of strings. Argument of Long options should be followed by an equal sign ('=').

(ii) Python's OS Module:

- The OS module in Python provides a way of using operating system dependent functionality.
- The functions that the OS module allows you to interface with the Windows operating system where Python is running on.
- **os.system():** Execute the C++ compiling command in the shell.
- For Example to compile C++ program g++ compiler should be invoked.
- **Command:** os.system ('g++' + <varaiable_name1> '-<mode>' + <variable_name2>
- **Example:**
  os.system('g++ ' + cpp_file + ' -o ' + exe_file) -- g++ compiler compiles the file cpp_file and –o (output) send to exe file

What is the purpose of sys, os, getopt module in Python. Explain

(i) Python's sys Module:

- This module provides access to some variables used by the interpreter and to functions that interact strongly with the interpreter.
- **sys.argv** is the list of command-line arguments passed to the Python program.
- **argv** contains all the items that come along via the command-line input, it's basically an array holding the command-line arguments of the program.
- To use **sys.argv**, you will first have to import sys.
- **sys.argv[0]** is always the name of the program as it was invoked.
- **sys.argv[1]** is the first argument you pass to the program.

38.

```
>>>
==================================
{1, 2, 3, 4, 5, 6, 7, 8}
{4, 5}
{1, 2, 3}
{8, 6, 7}
{1, 2, 3, 6, 7, 8}
>>>
```

OR

b) COUNT() function

The SQL COUNT() function returns the number of rows in a table satisfying the criteria specified in the WHERE clause. COUNT() returns 0 if there were no matching rows.

```
import sqlite3
connection = sqlite3.connect("Academy.db")
cursor = connection.cursor()
cursor.execute("SELECT COUNT(*)  FROM student ")
result = cursor.fetchall()
print(result)
```

Output:

[(7,)]

AVG():

The following SQL statement in the python program finds the average mark of all students.

```
import sqlite3
```

```
connection = sqlite3.connect("Academy.db")
cursor = connection.cursor()
cursor.execute("SELECT AVG(AVERAGE)  FROM student ")result =
cursor.fetchall()
print(result)
```

OUTPUT

```
[(84.65714285714286,)]
```

The following SQL statement in the python program finds the sum of all average in the Average field of "Student table".

```
import sqlite3
connection = sqlite3.connect("Academy.db")
cursor = connection.cursor()
cursor.execute("SELECT SUM(AVERAGE)  FROM student ")result =
cursor.fetchall()
print(result)
```


OUTPUT

```
[(592.6,)]
```

MAX() AND MIN() FUNCTIONS

The MAX() function returns the largest value of the selected column.

The MIN() function returns the smallest value of the selected column.

The following example show the highest and least average student's name

```
import sqlite3
connection = sqlite3.connect("Organization.db")
cursor = connection.cursor()
print("Displaying the name of the Highest Average")
cursor.execute("SELECT sname,max(AVERAGE)  FROM student ")result
cursor.fetchall()
print(result)
print("Displaying the name of the Least Average")
```

```
cursor.execute("SELECT sname,min(AVERAGE)  FROM student ")result
cursor.fetchall()
print(result)
```

OUTPUT

Displaying the name of the Highest Average

[('PRIYA', 98.6)]

Displaying the name of the Least Average

[('TARUN', 62.3)]

PREPARED BY

PRADEEP KUMAR

ANNAI VELANKANNI MHSS

SOWRIPALAYAM

COIMBATORE

CELL : 98941 34608