**Module 1**

**Previous Year Theory Questions**

**1.Let the variable x be "dog" and the variable y be "cat". Write the values returned by the following operations: - [May 2024] (3)**

**a) x\*4 + ' ' + 4\*y b) x\*len(x+y)**

**ANS:**

1. dogdogdogdog catcatcatcat
2. dogdogdogdogdogdog

**2.What will be the output if the following code fragments are executed? - [May 2024] (3)**

**for j in range(2,10,4):**

**print(j)**

**ANS:**

2

6

**3.Jack says that he will not bother with analysis and design but proceed directly to coding his programs. Why is that not a good idea? - [June 2023] (3)**

**ANS:**

Skipping analysis and design to jump directly into coding is a poor practice because it often leads to misunderstandings of user requirements, inefficient and disorganized code, and increased debugging and maintenance efforts. Without proper planning, the software may lack scalability, flexibility, and clarity, resulting in higher development costs and poor user experiences. Analysis and design provide a blueprint for building reliable, maintainable, and efficient systems, ultimately saving time and resources in the long run.

**4.Write the output of the following python statements :**

**i) round(12.57) ii) 5//2 iii) int(6.5) - [June 2023] (3)**

**ANS:**

i) 13

ii) 2

iii) 6

**5.What is the output of the following print statement in Python?**

**(a) print (9//2) (b) print (9/2) - [June 2022] (3)**

**ANS:**

1. 4
2. 4.5

**6.What is the output of the following Python code. Justify your answer. - [May 2023] (3)**

**x = 'abcd'**

**for i in range(len(x)):**

**print(i)**

**ANS:**

0

1

2

3

**7.Enumerate the various selection structures and control statement types in Python and elucidate each with suitable examples.** **- [May 2024] (8)**

**ANS:**

i)if Statement

* The if statement is used to execute a block of code only if a specified condition evaluates to True. If the condition is False, the block is skipped.
* Example

age = 18

if age >= 18:

    print("You are an adult.")

ii)if-else Statement

* The if-else statement allows you to specify an alternative block of code to execute when the condition evaluates to False. If the condition is True, the first block runs; otherwise, the second block runs.
* Example:

age = 16

if age >= 18:

    print("You are an adult.")

else:

    print("You are a minor.")

iii)if-elif-else Statement

* The if-elif-else statement provides multiple conditions to test. It evaluates the conditions sequentially and executes the block corresponding to the first True condition. If none of the conditions are True, the else block is executed.

* Example:

age = 70

if age < 18:

    print("You are a minor.")

elif age < 60:

    print("You are an adult.")

else:

    print("You are a senior citizen.")

iv)Ternary Operator (Conditional Expression)

* The ternary operator in Python allows you to assign a value based on a condition in a single line. It follows the syntax: value\_if\_true if condition else value\_if\_false.
* Example:

age = 20

status = "Adult" if age >= 18 else "Minor"

print(status)

v)assert Statement

* The assert statement is used to test if a condition is True. If the condition is False, it raises an AssertionError exception. This is typically used for debugging or testing purposes.
* Example:

age = 25

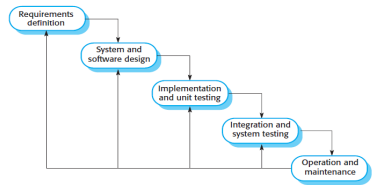
assert age >= 18, "Age must be 18 or older"

**8.Discuss the steps involved in the waterfall model of software development process with the help of a neat diagram.**

**- [January 2024] (8)**

**ANS:**

Waterfall Model



**Requirements Definition:**

* In this initial phase, detailed requirements for the software are gathered by interacting with stakeholders and analyzing user needs.
* The requirements are documented and serve as the foundation for the entire development process.
* No development begins until all requirements are clearly defined and approved.

**System and Software Design:**

* Based on the gathered requirements, system architecture and design specifications are created.
* This includes designing data structures, algorithms, user interfaces, and overall system architecture.
* The design guides the development team in implementing the software correctly.

**Implementation and Unit Testing:**

* The system design is divided into modules, and coding begins.
* Developers implement the software modules and individually test each module for functionality using unit tests.
* Unit testing ensures that each part works as expected before integrating them.

**Integration and System Testing:**

* Once individual modules are coded and tested, they are integrated into a complete system.
* The system undergoes thorough testing to ensure that all components work together as intended.
* System testing validates the software against the initial requirements.

**Operation and Maintenance:**

* After successful testing, the software is deployed to the production environment and made available to users.
* Maintenance includes fixing bugs, improving performance, and making updates as required to ensure the system continues to meet user needs.

**9.Illustrate the use of range() in Python.- [January 2024] (6)**

**ANS:**

The range() function in Python is a versatile tool used to generate a sequence of numbers, making it particularly useful for loops and iterations. It can accept up to three arguments: the starting point, the stopping point, and an optional step value to determine the interval between numbers in the sequence. By default, the sequence starts at zero and increments by one if no additional parameters are provided. When a step value is specified, the sequence can either increase or decrease accordingly, allowing for flexible iteration patterns. This function ensures efficient looping and enables developers to manage iteration ranges without manually creating lists or sequences..

i)**Basic Usage:**

Generates numbers from 0 to stop - 1.

for i in range(5):

print(i)

ii)**Specifying Start and Stop:**

Generates numbers from start to stop - 1.

for i in range(2, 6):

print(i)

iii)**Using a Step Value:**

Generates numbers incrementing by step.

for i in range(1, 10, 2):

print(i)

iv)**Negative Step Value:**

Generates numbers in descending order.

for i in range(10, 0, -2):

print(i)

**10.Mention the different types of loop and control statements allowed in Python and explain each type with suitable examples.**

**- [June 2022] (8)**

**ANS:**

Loops in Python

i) **for** Loop

\*Used to iterate over a sequence (like lists, strings, or ranges).

\*Executes the block of code once for each element in the sequence.

Example:

fruits = ["apple", "banana", "cherry"]

for fruit in fruits:

print(fruit)

ii) **while** Loop

\*Repeats a block of code as long as a specified condition is true.

Example:

count = 1

while count <= 5:

print(count)

count += 1

Control Statements in Python

Control statements alter the flow of a loop or skip parts of a block of code.

i) **break** statement

\*Terminates the current loop and resumes execution at the next statement.

Example:

for fruit in ["apple", "banana", "cherry"]:

if fruit == "banana":

break

print(fruit)

ii) **continue** statement

\*Skips the remaining code inside the loop for the current iteration and moves to the next iteration.

Example:

for fruit in ["apple", "banana", "cherry"]:

     if fruit == "banana":

         continue

     print(fruit)

iii) **pass** statement

\*Does nothing; acts as a placeholder for future code.

Example:

for fruit in ["apple", "banana", "cherry"]:

if fruit == "banana":

pass  # Placeholder

     print(fruit)

**11.Write the syntax and semantics of the multiway-if statement.**

**ANS:**

Python uses if-elif-else to achieve multi-branch decision-making. The syntax allows checking multiple conditions sequentially and executing corresponding code blocks.

Syntax

if condition1:

    # Code block executed if condition1 is True

    statement1

elif condition2:

    # Code block executed if condition2 is True

    statement2

elif condition3:

    # Code block executed if condition3 is True

    statement3

else:

    # Code block executed if no condition is True

    statement4

**Semantics**

1. **Condition Evaluation:**
   * Each condition is evaluated from top to bottom.
   * The first condition that evaluates to True executes its associated block.
   * Subsequent conditions are ignored once a True condition is found.
2. **Single Execution Path:**
   * Only one block of code runs even if multiple conditions are True.
3. **Optional elif and else:**
   * The elif and else clauses are optional.
   * There can be zero or more elif clauses, but at most one else clause.

**12.Explain type conversion with an example.**

**ANS:**

Type conversion in Python refers to converting one data type into another, either implicitly or explicitly. Implicit conversion (type coercion) happens automatically when Python converts a smaller data type to a larger one, such as converting an integer to a float during arithmetic operations. Explicit conversion (type casting) uses functions like int(), float(), or str() to convert data manually.

**Example:** Implicit conversion

num = 5

result = num + 2.5

print(result)

**Example:** Explicit conversion

age = "25"

age\_int = int(age)

print(age\_int + 5)