Theory Question

27\_jan\_se NAME: Vrinda Mavadhiya

## 1. What are arrays in C++?

An **array** in C++ is a collection of elements of the **same data type** stored in **contiguous memory locations**. Arrays allow you to store multiple values in a single variable.

**Single-dimensional Array:**

* A linear list of elements.
* Syntax: datatype arrayName[size];

int numbers[5] = {1, 2, 3, 4, 5};

**Multi-dimensional Array:**

* Arrays of arrays (e.g., 2D arrays like matrices).
* Syntax: datatype arrayName[row][column];

int matrix[2][3] = {{1, 2, 3}, {4, 5, 6}};

2. Explain string handling in C++ with examples**.**

C++ provides **two types of strings**:

* **C-style strings** (character arrays)
* **C++ string class** (std::string)

**C-style String:**

char name[10] = "John";

**C++ String Class (std::string):**

#include <iostream>

#include <string>

using namespace std;

int main()

{

string name = "John";

cout << "Name: " << name << endl;

return 0;

}

## 3. How are arrays initialized in C++? Provide examples of both 1D and 2D arrays.

**1D Array Initialization:**

int arr1[5] = {10, 20, 30, 40, 50};

int arr2[] = {1, 2, 3}; // Size auto-detected

**2D Array Initialization:**

int arr2D[2][3] =

{

{1, 2, 3},

{4, 5, 6}

};

You can also initialize partially:

int arrPartial[2][3] = { {1}, {4, 5} }; // Remaining values are 0

## 4. Explain string operations and functions in C++.

Using std::string, C++ provides many built-in string operations:

**Examples:**

#include <iostream>

#include <string>

using namespace std;

int main()

{

string s1 = "Hello";

string s2 = "World";

// Concatenation

string s3 = s1 + " " + s2;

cout << "Concatenated: " << s3;

// Length of string

cout << "Length: " << s3.length();

// Access characters

cout << "First char: " << s3[0] ;

// Substring

cout << "Substring: " << s3.substr(6, 5) ; // "World"

// Compare

if (s1 == "Hello")

{

cout << "s1 is Hello" ;

}

// Append

s1.append(" C++");

cout << "Appended: " << s1;

}