The University of Lahore

Department of Computer Science & IT

CS-09204 Data Structures and Algorithm
Fall 2025

Assignment # 1a

Participant ID #	_____	**CLO: 2** **PLO:**
Total Marks:	40	**Obtained Marks:**

Instructions:

Analyze the following C++ code snippets. For each code snippet, calculate its time complexity. Explain your reasoning and, if needed, describe how the time complexity is derived. Write down the time complexity in Big O notation (e.g., O(1), O(n), O(log n), O(n^2)).

Code Snippet 1

C++ Code	Time Complexity Analysis

int main() {
  int a = 10;
  int b = 20;
  int result = a + b; // Single operation
  cout << "Result: " << result << endl;
  return 0;
}	

Code Snippet 2

C++ Code	Time Complexity Analysis

int main() {
  int n = 100;
  int sum = 0;
  for (int i = 1; i <= n; i++) { // Loop from 1 to n
  sum += i;
  }
  cout << "Sum: " << sum << endl;
  return 0;}	

Code Snippet 3

C++ Code	Time Complexity Analysis

int main() {
  int n = 5;
  for (int i = 0; i < n; i++) { // Outer loop
  for (int j = 0; j < n; j++) { // Inner loop
  cout << i * j << endl;
  }
  }
  return 0;
}	

Code Snippet 4

C++ Code	Time Complexity Analysis

int binarySearch(int arr[], int n, int target) {
  int low = 0, high = n - 1;
  while (low <= high) {
  int mid = low + (high - low) / 2;
  if (arr[mid] == target) return mid; // Element found
  else if (arr[mid] < target) low = mid + 1;
  else high = mid - 1;
  }
  return -1; // Element not found
}	

Code Snippet 5

C++ Code	Time Complexity Analysis

int main() {
  int n = 10;
  for (int i = 1; i <= n; i *= 2) { // Logarithmic loop
  for (int j = 1; j <= n; j++) { // Linear loop inside
  cout << i + j << endl;
  }
  }
  return 0;}	