

Complex Problems

Problem 1: Simple Calculator

Write a basic calculator program that performs addition, subtraction, multiplication, and division.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int main() { 5 char op; 6 float num1, num2; 7 8 cout << "Enter operator (+, -, *, /): "; 9 cin >> op; 10 cout << "Enter two operands: "; 11 cin >> num1 >> num2; 12 13 switch(op) { 14 case '+': cout << num1 + num2; break; 15 case '-': cout << num1 - num2; break; 16 case '*': cout << num1 * num2; break; 17 case '/': cout << num1 / num2; break; 18 default: cout << "Invalid operator!"; 19 } 20 return 0; 21 } 22</pre>	<pre>/tmp/ea6CaTL9nZ.o Enter operator (+, -, *, /): * Enter two operands: 8 8 64 === Code Execution Successful ===</pre>

Problem 2: Sum of Digits

Write a program to find the sum of the digits of a given number.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int sumOfDigits(int n) { 5 int sum = 0; 6 while (n != 0) { 7 sum += n % 10; 8 n /= 10; 9 } 10 return sum; 11 } 12 13 int main() { 14 int num; 15 cout << "Enter a number: "; 16 cin >> num; 17 cout << "Sum of digits: " << sumOfDigits(num); 18 return 0; 19 } 20</pre>	<pre>/tmp/6ZhV0oQUj9.o Enter a number: 80 Sum of digits: 8 === Code Execution Successful ===</pre>

Problem 3: Sum of First N Natural Numbers

Develop a program that calculates the sum of the first N natural numbers.

main.cpp	Output
<pre> 1 #include <iostream> 2 using namespace std; 3 4 int main() { 5 int n; 6 cout << "Enter a number: "; 7 cin >> n; 8 cout << "Sum of first " << n << " natural numbers: " << (n * (n + 1)) / 2; 9 return 0; 10 } 11 </pre>	<pre> /tmp/m2scnZtk7W.o Enter a number: 89 Sum of first 89 natural numbers: 4005 === Code Execution Successful === </pre>

Problem 4: Count Vowels in a String

Write a program that counts the number of vowels in a given string.

main.cpp	Output
<pre> 1 #include <iostream> 2 #include <string> 3 using namespace std; 4 5 int countVowels(const string &str) { 6 int count = 0; 7 for (char c : str) { 8 if (c == 'a' c == 'e' c == 'i' c == 'o' c == 'u' 9 c == 'A' c == 'E' c == 'I' c == 'O' c == 'U') { 10 count++; 11 } 12 } 13 return count; 14 } 15 16 int main() { 17 string str; 18 cout << "Enter a string: "; 19 getline(cin, str); 20 cout << "Vowel count: " << countVowels(str); 21 return 0; 22 } </pre>	<pre> /tmp/8SLJm7CEc9.o Enter a string: waqas zafar Vowel count: 4 === Code Execution Successful === </pre>

Problem 5: Simple Interest Calculator

Create a program to calculate simple interest based on principal, rate, and time.

main.cpp	Output
<pre> 2 using namespace std; 3 4 int main() { 5 float principal, rate, time; 6 cout << "Enter the principal amount (initial loan or investment): "; 7 cin >> principal; 8 cout << "Enter the annual interest rate (as a percentage, e.g., 5 for 5%): "; 9 cin >> rate; 10 cout << "Enter the time period (in years): "; 11 cin >> time; 12 float simpleInterest = (principal * rate * time) / 100; 13 14 float totalAmount = principal + simpleInterest; 15 16 cout << "\n--- Interest Calculation ---\n"; 17 cout << "Principal Amount: PKR" << principal << endl; 18 cout << "Annual Interest Rate: " << rate << "%" << endl; 19 cout << "Time Period: " << time << " years" << endl; 20 cout << "Simple Interest: \$" << simpleInterest << endl; 21 cout << "Total Amount after " << time << " years: \$" << totalAmount << endl; 22 23 return 0; </pre>	<pre> /tmp/9U01YYqF3U.o Enter the principal amount (initial loan or investment): 100000 Enter the annual interest rate (as a percentage, e.g., 5 for 5%): 2 Enter the time period (in years): 3 --- Interest Calculation --- Principal Amount: PKR100000 Annual Interest Rate: 2% Time Period: 3 years Simple Interest: \$6000 Total Amount after 3 years: \$106000 === Code Execution Successful === </pre>

Problem 6: Multiplication Table Generator

Write a program to generate and display the multiplication table of any given number.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int main() { 5 int n; 6 cout << "Enter a number: "; 7 cin >> n; 8 for (int i = 1; i <= 10; ++i) { 9 cout << n << " * " << i << " = " << n * i << endl; 10 } 11 return 0; 12 } 13</pre>	<pre>/tmp/Ka4tDAyVEq.o Enter a number: 12 12 * 1 = 12 12 * 2 = 24 12 * 3 = 36 12 * 4 = 48 12 * 5 = 60 12 * 6 = 72 12 * 7 = 84 12 * 8 = 96 12 * 9 = 108 12 * 10 = 120 === Code Execution Successful ===</pre>

Problem 7: Count Positive and Negative Numbers

Write a program that reads a list of numbers and counts how many are positive, negative, and zero.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int main() { 5 int n, num, pos = 0, neg = 0, zero = 0; 6 cout << "Enter the number of elements: "; 7 cin >> n; 8 for (int i = 0; i < n; ++i) { 9 cout << "Enter a number: "; 10 cin >> num; 11 if (num > 0) pos++; 12 else if (num < 0) neg++; 13 else zero++; 14 } 15 cout << "Positive: " << pos << "\nNegative: " << neg << "\nZero: " << zero; 16 return 0; 17 }</pre>	<pre>/tmp/FzG07pSCb6.o Enter the number of elements: 5 Enter a number: 1 Enter a number: -2 Enter a number: 0 Enter a number: 0 Enter a number: -6 Positive: 1 Negative: 2 Zero: 2 === Code Execution Successful ===</pre>

Problem 8: Leap Year Checker

Write a program that checks if a given year is a leap year or not.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 bool isLeapYear(int year) { 5 return (year % 4 == 0 && year % 100 != 0) (year % 400 == 0); 6 } 7 8 int main() { 9 int year; 10 cout << "Enter a year: "; 11 cin >> year; 12 cout << year << (isLeapYear(year) ? " is a leap year." : " is not a leap year."); 13 return 0; 14 } 15</pre>	<pre>/tmp/oNe0uH1dM3.o Enter a year: 2028 2028 is a leap year. === Code Execution Successful ===</pre>

Problem 9: Temperature Converter

Write a program that converts temperatures from Celsius to Fahrenheit and vice versa.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 void celsiusToFahrenheit(float celsius) { 5 cout << "Fahrenheit: " << (celsius * 9/5) + 32 << endl; 6 } 7 8 void fahrenheitToCelsius(float fahrenheit) { 9 cout << "Celsius: " << (fahrenheit - 32) * 5/9 << endl; 10 } 11 12 int main() { 13 int choice; 14 float temp; 15 cout << "1. Celsius to Fahrenheit\n2. Fahrenheit to Celsius\nChoose an option: "; 16 cin >> choice; 17 cout << "Enter temperature: "; 18 cin >> temp; 19 if (choice == 1) celsiusToFahrenheit(temp); 20 else fahrenheitToCelsius(temp); 21 return 0; }</pre>	<pre>/tmp/ku3uyf7Lur.o 1. Celsius to Fahrenheit 2. Fahrenheit to Celsius Choose an option: 2 Enter temperature: 90 Celsius: 32.2222 === Code Execution Successful ===</pre>

Problem 10: Swap Two Numbers

Write a program that swap two number.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int main() { 5 int a, b; 6 cout << "Enter number A: "; 7 cin >> a; 8 cout << "Enter number B: "; 9 cin >> b; 10 a = a + b; 11 b = a - b; 12 a = a - b; 13 cout << "After swapping: a = " << a << ", b = " << b; 14 return 0; 15 } 16</pre>	<pre>/tmp/lfwAu01pAN.o Enter number A: 12 Enter number B: 34 After swapping: a = 34, b = 12 === Code Execution Successful ===</pre>

Problem 11: Add Two numbers without the + operator.

Write a program that Add Two numbers without + operator.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int add(int a, int b) { 5 while (b != 0) { 6 int carry = a & b; 7 a = a ^ b; 8 b = carry << 1; 9 } 10 return a; 11 } 12 13 int main() { 14 int x, y; 15 cout << "Enter two numbers: "; 16 cin >> x >> y; 17 cout << "Sum: " << add(x, y); 18 return 0; }</pre>	<pre>/tmp/5CMaDYuS01.o Enter two numbers: 50 120 Sum: 170 === Code Execution Successful ===</pre>

Problem 12: Factorial Calculation.

Write an algorithm to calculate the factorial of a given positive integer using iteration.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 long long factorial(int n) { 4 long long result = 1; 5 for (int i = 1; i <= n; ++i) { 6 result *= i; 7 } 8 return result; 9 } 10 int main() { 11 int n; 12 cout << "Enter a non-negative integer to find its factorial: "; 13 cin >> n; 14 if (n < 0) { 15 cout << "Factorial is not defined for negative numbers." << endl; 16 } else { 17 cout << "Factorial of " << n << " is: " << factorial(n) << endl; 18 } 19 return 0; 20 }</pre>	<pre>/tmp/JbMTFIDyxw.o Enter a non-negative integer to find its factorial: 8 Factorial of 8 is: 40320 === Code Execution Successful ===</pre>

Problem 19: Armstrong Number

Write a program to check if a given number is an Armstrong number (a number that is equal to the sum of its own digits raised to the power of the number of digits).

main.cpp	Output
<pre>4 5 bool isArmstrong(int num) { 6 int original = num, sum = 0; 7 int digits = log10(num) + 1; 8 9 while (num > 0) { 10 int digit = num % 10; 11 sum += pow(digit, digits); 12 num /= 10; 13 } 14 return sum == original; 15 } 16 int main() { 17 int num; 18 cout << "Enter a number: "; 19 cin >> num; 20 if (isArmstrong(num)) { 21 cout << num << " is an Armstrong number." << endl; 22 } else { 23 cout << num << " is not an Armstrong number." << endl; 24 } 25 return 0; }</pre>	<pre>/tmp/TAeJvfw10e.o Enter a number: 80 80 is not an Armstrong number. === Code Execution Successful ===</pre>

Problem 20: Find the Second Largest Number

Write a program that finds the second largest number in an array of integers.

main.cpp	Run	Output
<pre>5 int findSecondLargest(int arr[], int n) { 6 int first = INT_MIN, second = INT_MIN; 7 8 for (int i = 0; i < n; ++i) { 9 if (arr[i] > first) { 10 second = first; 11 first = arr[i]; 12 } else if (arr[i] > second && arr[i] < first) { 13 second = arr[i]; 14 } 15 } 16 return second; 17 } 18 int main() { 19 int arr[] = {12, 35, 1, 10, 34, 1}; 20 int n = sizeof(arr) / sizeof(arr[0]); 21 22 int result = findSecondLargest(arr, n); 23 if (result != INT_MIN) { 24 cout << "The second largest element is: " << result << endl; 25 } else { 26 cout << "No second largest element found." << endl; 27 } }</pre>		<pre>/tmp/RU1xVLJLKn.o The second largest element is: 34 === Code Execution Successful ===</pre>

Problem 21: Second Smallest Number

Write a program to find the second smallest number in an array of integers.

main.cpp	Run	Output
<pre>3 using namespace std; 4 5 int findSecondSmallest(int arr[], int n) { 6 int first = INT_MAX, second = INT_MAX; 7 8 for (int i = 0; i < n; ++i) { 9 if (arr[i] < first) { 10 second = first; 11 first = arr[i]; 12 } else if (arr[i] < second && arr[i] > first) { 13 second = arr[i]; 14 } 15 } 16 return second; 17 } 18 int main() { 19 int arr[] = {12, 35, 1, 10, 34, 1}; 20 int n = sizeof(arr) / sizeof(arr[0]); 21 int result = findSecondSmallest(arr, n); 22 if (result != INT_MAX) { 23 cout << "The second smallest element is: " << result << endl; 24 } else { 25 cout << "No second smallest element found." << endl; 26 } 27 return 0; }</pre>		<pre>/tmp/L1LCVsR2pr.o The second smallest element is: 10 === Code Execution Successful ===</pre>

Problem 22: Second Largest in a Sorted Array

Given a sorted array, write a program to find the second largest number without iterating through the entire array.

main.cpp	Output
<pre>1 #include <iostream> 2 using namespace std; 3 4 int findSecondLargestSorted(int arr[], int n) { 5 if (n < 2) return -1; 6 int largest = arr[n - 1]; 7 for (int i = n - 2; i >= 0; --i) { 8 if (arr[i] < largest) { 9 return arr[i]; 10 } 11 } 12 return -1; 13 } 14 int main() { 15 int arr[] = {1, 2, 3, 4, 5, 5}; 16 int n = sizeof(arr) / sizeof(arr[0]); 17 int result = findSecondLargestSorted(arr, n); 18 if (result != -1) { 19 cout << "The second largest element is: " << result << endl; 20 } else { 21 cout << "No second largest element found." << endl; 22 } 23 return 0; }</pre>	<pre>/tmp/DqHFQLpUCZ.o The second largest element is: 4 === Code Execution Successful ===</pre>

Problem 23: Remove Duplicates and Find Second Largest

Write a program that removes duplicate numbers from an array and then finds the second largest number in the resulting array.

<pre>int removeDuplicates(int arr[], int n) { if (n == 0 n == 1) return n; int temp[n]; int j = 0; for (int i = 0; i < n - 1; ++i) { if (arr[i] != arr[i + 1]) { temp[j++] = arr[i]; } } temp[j++] = arr[n - 1]; for (int i = 0; i < j; ++i) { arr[i] = temp[i]; } return j; } int findSecondLargestAfterRemovingDuplicates(int arr[], int n) { sort(arr, arr + n); int newSize = removeDuplicates(arr, n); if (newSize < 2) return -1; return arr[newSize - 2]; } int main() { int arr[] = {1, 2, 3, 4, 5, 5, 3}; int n = sizeof(arr) / sizeof(arr[0]); int result = findSecondLargestAfterRemovingDuplicates(arr, n); if (result != -1) { cout << "The second largest element after: " << result ; } }</pre>	<pre>Output: The second largest element after: 4</pre>
--	--

Problem 24: Second Largest Using a Single Pass

Write an algorithm to find the second largest number in an array in a single pass (without sorting the array).

main.cpp	Output
<pre>4 int findSecondLargestSinglePass(int arr[], int n) { 5 int first = INT_MIN, second = INT_MIN; 6 for (int i = 0; i < n; ++i) { 7 if (arr[i] > first) { 8 second = first; 9 first = arr[i]; 10 } else if (arr[i] > second && arr[i] < first) { 11 second = arr[i]; 12 } 13 } 14 return second; 15 } 16 int main() { 17 int arr[] = {10, 5, 10, 15, 5, 20}; 18 int n = sizeof(arr) / sizeof(arr[0]); 19 int result = findSecondLargestSinglePass(arr, n); 20 if (result != INT_MIN) { 21 cout << "The second largest element is: " << result << endl; 22 } else { 23 cout << "No second largest element found." << endl; 24 } 25 return 0; }</pre>	<pre>/tmp/93ywHdZZfV.o The second largest element is: 15 === Code Execution Successful ===</pre>

Problem 25: Second Largest in a 2D Array

Write a program to find the second largest number in a 2D array of integers.

main.cpp	Output
<pre>5 int findSecondLargest2D(int arr[][3], int rows, int cols) { 6 int first = INT_MIN, second = INT_MIN; 7 8 for (int i = 0; i < rows; ++i) { 9 for (int j = 0; j < cols; ++j) { 10 if (arr[i][j] > first) { 11 second = first; 12 first = arr[i][j]; 13 } else if (arr[i][j] > second && arr[i][j] < first) { 14 second = arr[i][j]; 15 } 16 } 17 } 18 return second; 19 } 20 int main() { 21 int arr[3][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}; 22 int rows = 3, cols = 3; 23 int result = findSecondLargest2D(arr, rows, cols); 24 if (result != INT_MIN) { 25 cout << "The second largest element in the 2D array is: " << result << endl; 26 } 27 }</pre>	<pre>/tmp/y0NKLZatQ2.o The second largest element in the 2D array is: 8 === Code Execution Successful ===</pre>

```
24     } else {
25         cout << "The second largest element in the 2D array is: " << result <<
            endl;
26     }
27     } else {
28         cout << "No second largest element found." << endl;
29     }
30     return 0;
}
```