// =======================

// Included: Lab #6 Program #1, Program #2

// =======================

// Program #1

// =======================

// Christian Falucho

// CMPR 121

// =======================

#include <iostream>

#include <fstream>

#include <iomanip>

using namespace std;

/\*

===========================================================

=== MAIN FUNCTION BEGINS ===

===========================================================

\*/

int main(){

fstream dataFile;

double number = 17.816392;

dataFile.open("values.txt", ios::out);

dataFile << number << endl;

dataFile << setprecision(2) << showpoint << fixed;

dataFile << number;

cout << "Data has been written to file.\n";

return 0;

}/\*

===========================================================

=== MAIN FUNCTION ENDS ===

===========================================================

\*/

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/

A screenshot of a computer

AI-generated content may be incorrect.

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/

// =======================

// Included: Lab #6 Program #1, Program #2

// =======================

// Program #2

// =======================

// Christian Falucho

// CMPR 121

// =======================

#include <iostream>

#include <fstream>

using namespace std;

/\*

===========================================================

=== MAIN FUNCTION BEGINS ===

===========================================================

\*/

int main(){

const int SIZE = 10;

int numbers[SIZE] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

fstream file;

// Open the file and write to the file in binary

// Close the file once done

file.open("test.bin", ios::out | ios::binary);

cout << "Writing the data to the file.\n";

file.write(reinterpret\_cast<char \*>(numbers), SIZE);

file.close();

// Open the file and read the binary contents of the file

// Display contents on terminal

// Close the file

cout << "Now reading the data back into memory.\n";

file.open("text.bin", ios::in | ios::binary);

file.read(reinterpret\_cast<char \*>(numbers), SIZE);

for (int i = 0; i < SIZE; i++)

{

cout << numbers[i] << " ";

}

cout << endl;

file.close();

return 0;

}

/\*

===========================================================

=== MAIN FUNCTION ENDS ===

===========================================================

\*/

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/

A black background with white text

AI-generated content may be incorrect.

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/