This C++ program reads data from a CSV file, processes it, and writes the result to another CSV file. Let's go through the code step by step:

Headers and Struct Definition:

cpp

Copy code

#include <iostream>

#include <fstream>

#include <sstream>

#include <map>

using namespace std;

// A struct to represent cell data in the CSV file

struct CellData

{

int count;

int tWeight;

int dWeight;

};

The program includes necessary header files (iostream, fstream, sstream, map).

It defines a struct CellData to represent the data for each cell in the CSV file, containing the count, T-weight, and D-weight.

Function to Read CSV Data:

cpp

Copy code

void readCSVData(const string &filename, map<string, map<string, CellData>> &cellData,

map<string, int> &columnTotal, map<string, int> &rowTotal)

{

// Function to read data from the input CSV file into provided data structures

fstream file(filename, ios::in);

if (!file.is\_open())

{

cout << "Couldn't open file: " << filename << endl;

return;

}

// ... (code to read and process CSV data)

}

readCSVData is a function that takes a filename and three maps (cellData, columnTotal, rowTotal) as references.

It opens the input CSV file, reads its data, and populates the provided data structures.

Function to Write CSV Result:

cpp

Copy code

void writeCSVResult(const string &filename, const map<string, map<string, CellData>> &cellData,

const map<string, int> &columnTotal, const map<string, int> &rowTotal)

{

// Function to write the result to an output CSV file

ofstream outputFile(filename, ios::out);

// ... (code to write CSV result)

}

writeCSVResult is a function that takes a filename and three maps (cellData, columnTotal, rowTotal) as references.

It opens an output CSV file and writes the processed data to it.

Main Function:

cpp

Copy code

int main()

{

// ... (code to declare and initialize data structures)

readCSVData("Input\_File.csv", cellData, columnTotal, rowTotal);

writeCSVResult("Output\_File.csv", cellData, columnTotal, rowTotal);

cout << endl << "Check 'Output\_File.csv' for the result" << endl;

return 0;

}

The main function initializes maps (cellData, columnTotal, rowTotal), calls readCSVData to read data from "Input\_File.csv," and then calls writeCSVResult to write the result to "Output\_File.csv."

It prints a message indicating where to check the result.

Overall, this program is designed to perform calculations on CSV data, specifically counting, T-weight, and D-weight calculations, and then writing the result to another CSV file. The cell data, column totals, and row totals are stored in corresponding maps for further processing.