**CSC 1101 – Problem Solving and Programming Laboratory**

**Lab 6 – Omar Faruk**

**25 points – Due September 28, 11pm**

**a)** Save this document with your name and the homework number somewhere in the file name.

**b)** Paste your code and screenshots into the document.

**c)** Submit this document and your .cpp file(s) to the Canvas item where you downloaded this document. Do not submit a zip file but individually attach your files.

**1) [12 points]** You've been hired by *Cash Closet* to write a C++ console application that converts a US dollar value to another currency. Visit [www.oanda.com/currency/converter/](https://www.oanda.com/currency/converter/), choose a currency, and note the exchange rate. Prompt for and get from the user a real-number value in US dollars. Calculate its real-number equivalent in your chosen currency. Use formatted output manipulators (setw, left/right) to print the following rows:

● Value in US dollars

● Conversion rate (per US dollar)

● Value in chosen currency

And two columns:

● A left-justified label.

● A right-justified value.

Define constants for the conversion rate and column widths. Format all real numbers to two decimal places. The output should look like this:

Welcome to Cash Closet

----------------------

Enter a value (US dollars): 45.12

US dollars: 45.12

Conversion rate (per US dollar): 1.33

Canadian dollars: 60.05

End of Cash Closet

Do not use this sample input for the final run that is pasted below. Run the program three times with different values for US dollars but the same chosen currency. What are the converted values?

|  |  |  |
| --- | --- | --- |
| Run | US dollars | Converted value |
| 1 | 25 | 2082.07 |
| 2 | 750 | 62462.25 |
| 3 | 1500 | 124924.50 |

*[your program code here]\**

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

//

// Title: Currency Converter

// Course: CSC 1101

// Lab Number: Lab 06-01

// Author: Omar Faruk

// Date: 09/27/2020

// Description:

// Currency converter from USD to BD Taka or BDT.

// Using table formatting and Oanda for currency rate after user input.

//

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

#include <cstdlib> // For several general-purpose functions

#include <fstream> // For file handling

#include <iomanip> // For formatted output

#include <iostream> // For cin, cout, and system

#include <string> // For string data type

using namespace std; // So "std::cout" may be abbreviated to "cout"

int main()

{

// Setting Decimal Points

cout << fixed << setprecision(2);

// Declare variables

double usDollar;

double currencyEquation;

double totalBdt;

// Constants

const double bdt = 83.2830;

const double usdNum = 1;

const int COLFMT1 = 30;

const int COLFMT2 = 15;

// Show application header

cout << "Cash Converter" << endl;

cout << "--------------------------" << endl << endl;

// Write to screen

cout << "Hello" << endl;

cout << "Welcome to the Currency Converter!" << endl;

// cout << "Enter a number: ";

cout << "\nEnter a number in US Dollar:";

//User Input

cin >> usDollar;

// Calculation

currencyEquation = (bdt/usdNum);

totalBdt = (usDollar \* currencyEquation);

// Adding spacing

cout << endl;

//Table Formatting

cout << setw(COLFMT1) << left << "US Dollars:";

cout << setw(COLFMT2) << right << usDollar << endl;

cout << setw(COLFMT1) << left << "Conversion rate (per USD):";

cout << setw(COLFMT2) << right << currencyEquation << endl;

cout << setw(COLFMT1) << left << "Bangaldesh Taka:";

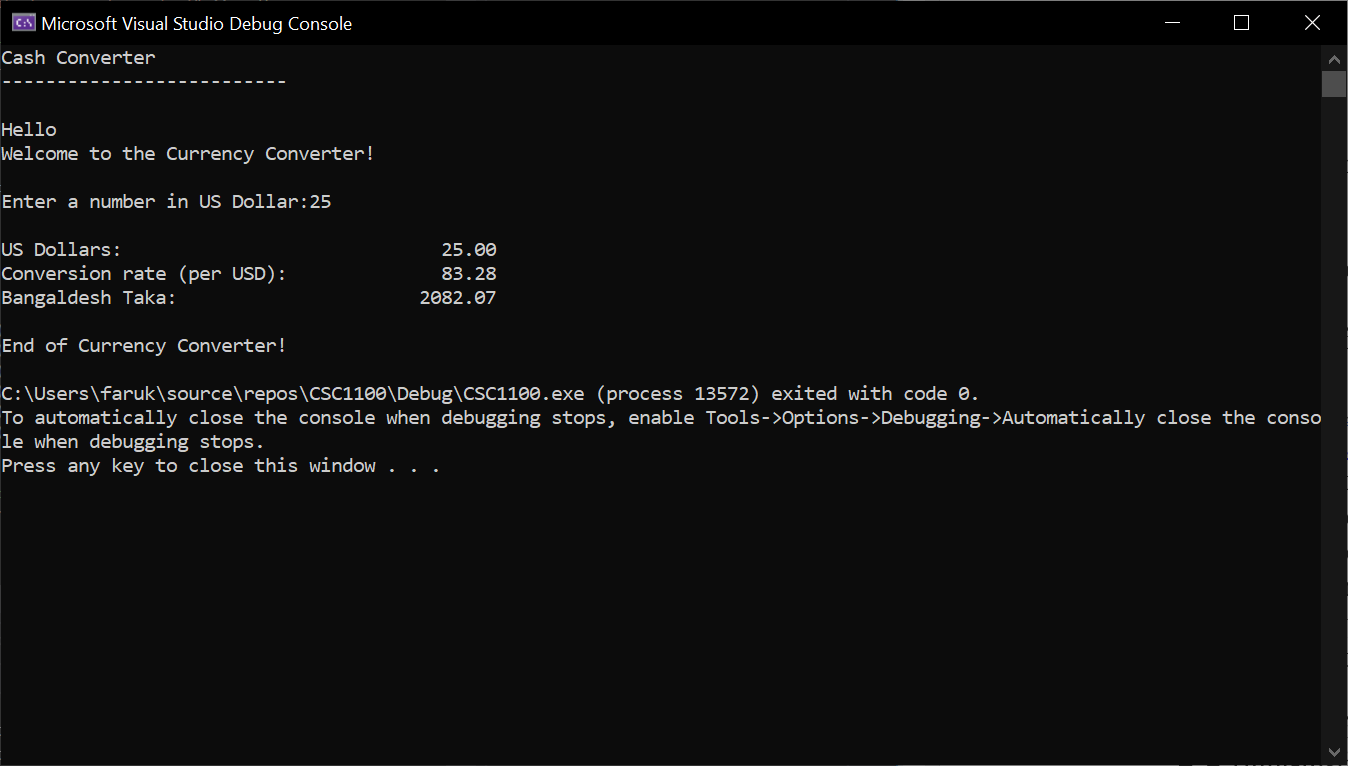
cout << setw(COLFMT2) << right << totalBdt << endl;

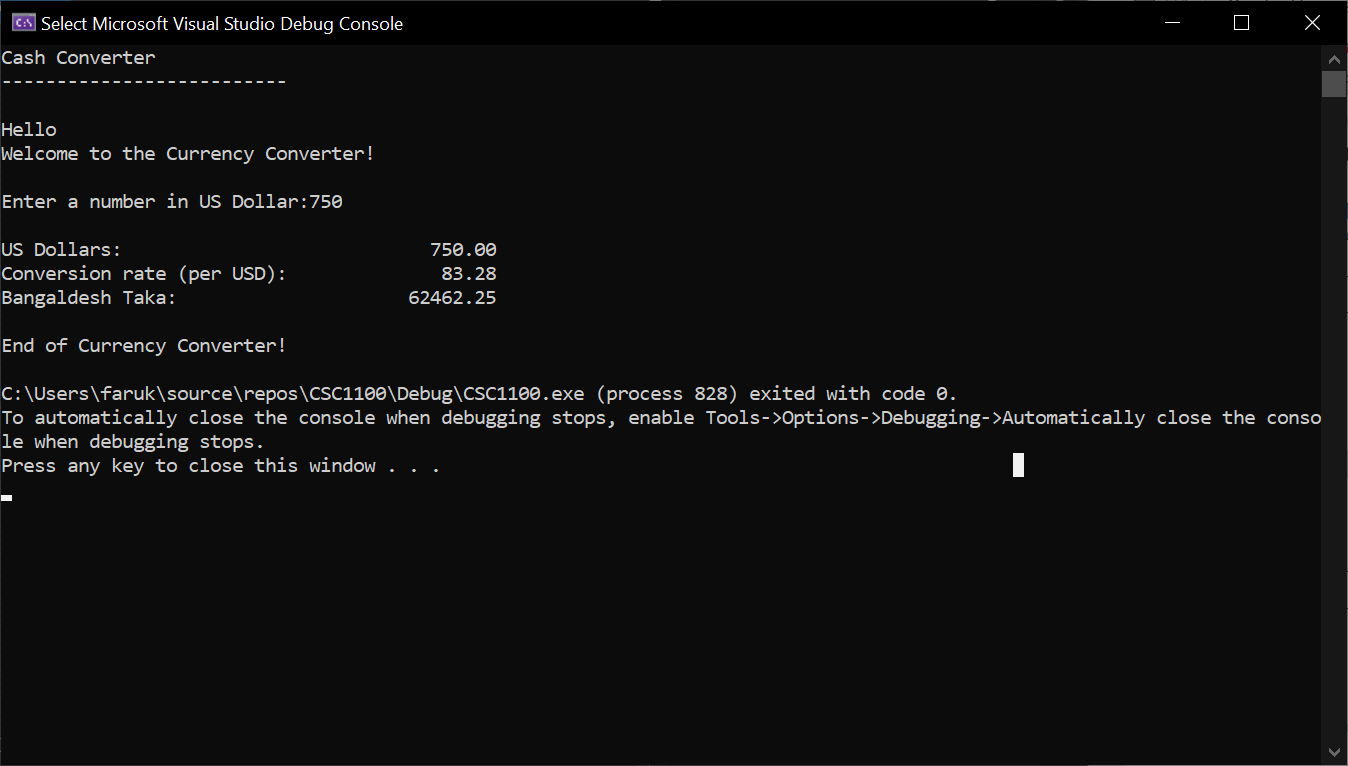
// Show application close

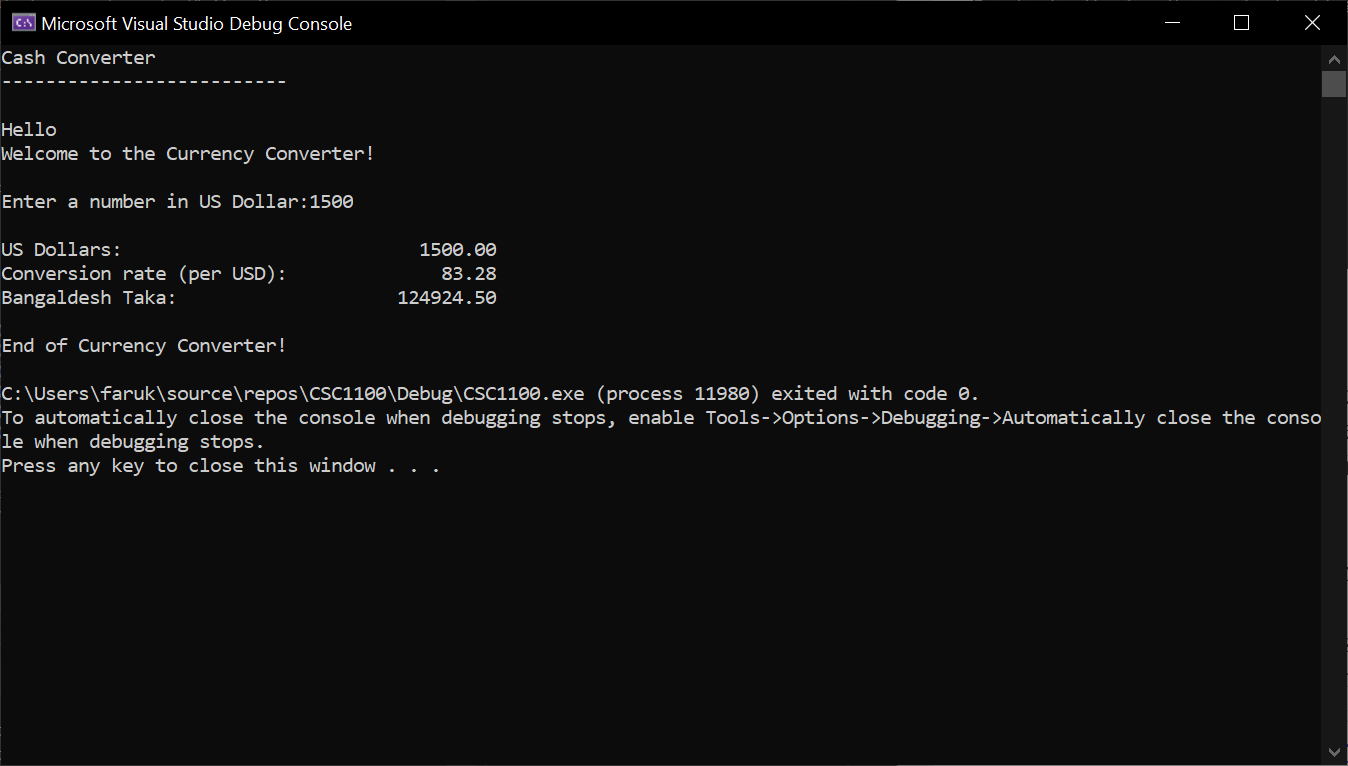
cout << "\nEnd of Currency Converter!" << endl;

}

*[your program output for one run here]\*\**







**2) [13 points]** You've been hired by *Frigid Feet* to write a C++ console application that calculates and displays the Celsius and Fahrenheit temperatures at a given altitude. The further you are from the surface of the Earth, the colder its gets. Here is the formula:

air-temperature = ground-temperature - (altitude \* 6.5 degrees per kilometer)

Prompt for and get from the user real numbers for the ground temperature in Celsius and an altitude in kilometers. Calculate the air temperature and show in both Celsius and Fahrenheit units. See <https://www.albireo.ch/temperatureconverter/formula.html> for the Celsius-to-Fahrenheit conversion. Use formatted output manipulators (setw, left/right) to print the following rows:

● Temperature on ground (°C)

● Altitude (km)

● Celsius temperature

● Fahrenheit temperature

And three columns:

● A left-justified label.

● A right-justified value.

● Left-justified units (no manipulators needed for this column).

Define constants for the column widths. Format all real numbers to two decimal places. Include the degree symbol in the prompt and output:

● Windows users – declare a constant for the degree symbol:

const char DEGREE\_SYMBOL = (char) 167

● Mac OS users – copy the degree symbol above to your code.

The output should look like this:

Welcome to Frigid Feet

----------------------

Enter ground temperature (ºC): 22

Enter altitude (km): 7.4

Ground temperature: 22.00 ºC

Altitude: 7.40 km

Air temperature: -26.10 ºC

Air temperature: -14.98 ºF

End of Frigid Feet

Do not use this sample input for the final run that is pasted below. Run the program five times with different values for the two integers. What are the resulting air temperatures?

|  |  |  |  |
| --- | --- | --- | --- |
| Run | Ground temp (°C) | Altitude (km) | Air temp (°C) |
| 1 | 11 | 7.8 | -39.70 |
| 2 | 15 | 5.7 | -22.05 |
| 3 | 20 | 10 | -45.00 |
| 4 | 27 | 4 | 1.00 |
| 5 | 30 | 1.5 | 20.25 |

*[your program code here]\**

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

//

// Title: Temperate at Altitude

// Course: CSC 1101

// Lab Number: Lab 06-02

// Author: Omar Faruk

// Date: 09/27/2020

// Description:

// Calculate the temperate at a certain altitude

// using user input, calculations, table formatting, and char.

//

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

#include <cstdlib> // For several general-purpose functions

#include <fstream> // For file handling

#include <iomanip> // For formatted output

#include <iostream> // For cin, cout, and system

#include <string> // For string data type

using namespace std; // So "std::cout" may be abbreviated to "cout"

int main()

{

//Setting 2 decimal points

cout << fixed << setprecision(2);

// Constant

const char DEGREE\_SYMBOL = (char)167;

const int COLFMT1 = 20;

const int COLFMT2 = 5;

// Declare variables

int groundTemperature;

double altitude;

double airTemperature;

double fahrenheit;

// Show application header

cout << "Welcome to Frigid Feet!" << endl;

cout << "--------------------------" << endl << endl;

// User Input

cout << "\nEnter ground temperature" << DEGREE\_SYMBOL << "C" << ":";

cin >> groundTemperature; // Ground temperature in Celsius

cout << "Enter altitude (km):";

cin >> altitude; // Altitude in Kilometers

// Calculation

airTemperature = groundTemperature - (altitude \* 6.5); // air temp

fahrenheit = (airTemperature \* 9 / 5) + 32; // Celsius to Fahrenheit

// New line for spacing

cout << endl;

//Table Formatting

cout << setw(COLFMT1) << left << "Ground temperature:";

cout << setw(COLFMT2) << right << groundTemperature << " ";

cout << left << DEGREE\_SYMBOL << "C" << endl;

cout << setw(COLFMT1) << left << "Altitude:";

cout << setw(COLFMT2) << right << altitude << " ";

cout << left << "Km" << endl;

cout << setw(COLFMT1) << left << "Air temperature:";

cout << setw(COLFMT2) << right << airTemperature << " ";

cout << left << DEGREE\_SYMBOL << "C" << endl;

cout << setw(COLFMT1) << left << "Air temperature:";

cout << setw(COLFMT2) << right << fahrenheit << " ";

cout << left << DEGREE\_SYMBOL << "F" << endl;

// Show application close

cout << "\nEnd of Frigid Feet!" << endl;

}

*[your program output for one run here]\*\**

