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

**Lab 6 – [your name]**

**25 points – Due February 8, 11pm**

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

**b)** Type/paste your answers into the document.

**c)** Submit the following documents to the Canvas assignment link where you downloaded this document:

✓ This document.

✓ Your .cpp files renamed to .txt.

Submit the documents separately, not as one .zip file.

**1) [13 points]** You've been hired by *Tea Tempests* to write a C++ console application that determines how long it takes for something left on the kitchen counter to assume room temperature. Prompt for and get from the user the item left out, and the current room and item starting temperatures. Assuming a cooling/warming rate of 2°C per minute, calculate how long it takes for the item to reach room temperature. Use formatted output manipulators (setw, left/right) to print the following rows:

● Item

● Room temperature (°C)

● Item starting temperature (°C)

● Time to room temperature (seconds)

And three columns:

● A left-justified label.

● A right-justified value.

● A left-justified units.

Define constants for the cooling/warming rate and column widths. Format all real numbers to one decimal place. Include the degree symbol in the prompts and output:

● Windows users – you may use:

(char) 167

● macOS users – you may copy and paste the degree symbol to your code.

In the calculation for cooling/warming time, use the absolute value function to get a positive time:

abs(…)

The output should look like this:

Welcome to Tea Tempests

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

Enter the item you left on the counter: Milk

Enter the current room temperature (ºC): 20

Enter the current item temperature (ºC): 4

Item: Milk

Room temperature: 20.0 ºC

Item temperature: 4.0 ºC

Cooling/warming time: 480.0 seconds

End of Tea Tempests

Do not use this sample input for the final run that is pasted below. Run the program three times with different values for item and room and item temperatures.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Run | Item | Room temp (ºC) | Item temp (ºC) | Cooling/warming time (seconds) |
| 1 | laptop | 22 | 31 | 270 |
| 2 | Water bottle | 19 | 16 | 90 |
| 3 | Book | 27 | 19 | 240 |

**//==========================================================**

**//**

**// Title: Lab 06**

**// Course: CSC 1101**

**// Lab Number: 06**

**// Author: Rory Lange**

**// Date: 2/5/21**

**// Description:**

**// Create an app that tells when an item reaches**

**// room temp.**

**//**

**//==========================================================**

**#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()**

**{**

**// Declare variables**

**string item;**

**char degree = 167;**

**int room;**

**int itemTemp;**

**int tempDiff;**

**const int tempIncrease = 2;**

**const int width = 25;**

**const int widthr = 20;**

**// Show application header**

**cout << "Item Room Temperature Calculator" << endl;**

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

**//collect data**

**cout << setw(width) << left << "Name of item: ";**

**cout << setw(widthr) << right;**

**getline(cin, item);**

**cout << setw(width) << left << "Room Temperature (C): ";**

**cin >> setw(widthr) >> right >> room;**

**cout << setw(width) << left << "Item Temperature (C): ";**

**cin >> itemTemp;**

**cout << endl;**

**//do math**

**if (itemTemp > room) {**

**tempDiff = itemTemp - room;**

**}**

**else {**

**tempDiff = room - itemTemp;**

**}**

**float time = (float) tempDiff / tempIncrease \* 60;**

**//report results**

**cout << setw(width) << left << "Nam of Item: "**

**<< setw(widthr) << right << item << endl;**

**cout << setw(width) << left << "Room Temperature: "**

**<< setw(widthr) << right << room << " " << left << degree << "C" << endl;**

**cout << setw(width) << left << "Item Temperature: "**

**<< setw(widthr) << right << itemTemp << " " << left << degree << "C" << endl;**

**cout << setw(width) << left << "Cooling/Warming time: "**

**<< setw(widthr) << right << time << left << " Seconds" << endl;**

**// Show application close**

**cout << "\nEnd of my Application" << endl;**

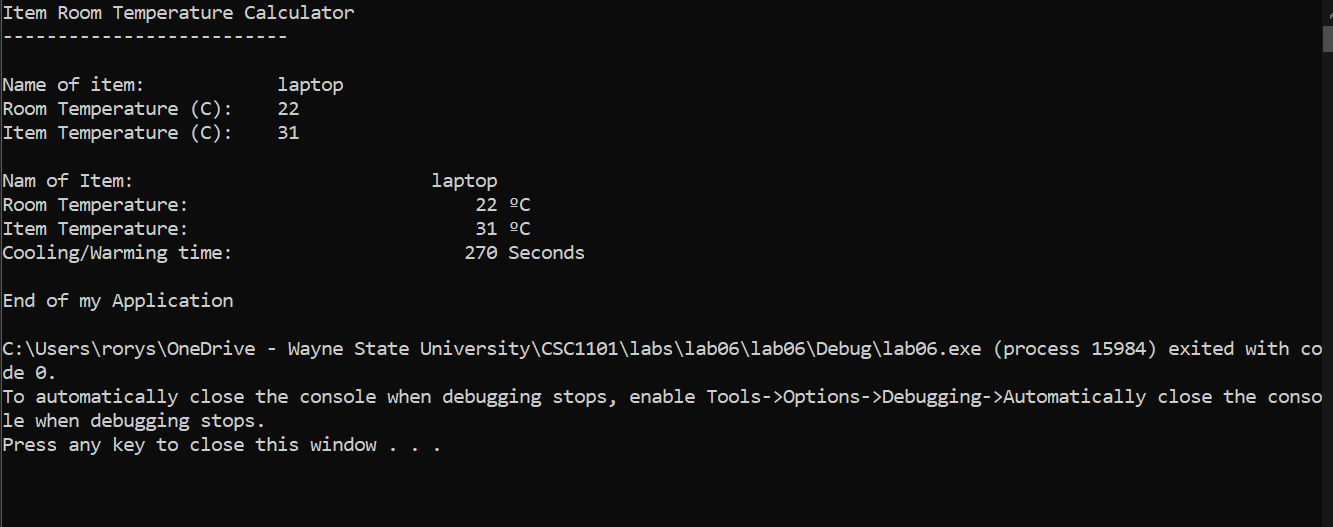
**}**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

**

**2) [12 points]** You've been hired by *Planet Pals* to write a C++ console application that calculates and displays the distance from Earth to another planet in parsecs. Prompt the user for a planet and the distance to the planet from Earth in astronomical units (AUs). See [theplanets.org/distances-between-planets/](https://theplanets.org/distances-between-planets/) for AU distances. There are 206264.8 AUs per parsec. Calculate the distance to the planet in parsecs. Use formatted output manipulators (setw, left/right) to print the following rows:

● Planet

● Distance to planet in AUs

● Distance to planet in parsecs

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 ten decimal places. The output should look like this:

Welcome to Planet Pals

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

Enter the planet to travel to from Earth: Mars

Distance to planet (AUs): .52

Planet: Mars

Distance to (AUs): 0.5200000000

Distance to (parsecs): 0.0000025210

End of Planet Pals

Do not use this sample input for the final run that is pasted below. Run the program three times with different values for planet and AUs.

|  |  |  |  |
| --- | --- | --- | --- |
| Run | Planet | AUs | Parsecs |
| 1 | Mercury | .61 | .0000029574 |
| 2 | Neptune | 29.09 | .0001410323 |
| 3 | Uranus | 18.21 | .0000882846 |

**//==========================================================**

**//**

**// Title:      Lab 06**

**// Course:     CSC 1101**

**// Lab Number: 06**

**// Author:     Rory Lange**

**// Date:       2/5/21**

**// Description:**

**//   Create an app that tells when an item reaches**

**// room temp.**

**//**

**//==========================================================**

**#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() {**

**//declare variables**

**const float conversion = 206264.8;**

**const int w = 25;**

**const int w1 = 20;**

**string planet;**

**double AU;**

**double PSC;**

**//header**

**cout << "PLANET PALS DISTANCE CONVERSIONS" << endl;**

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

**//user input**

**cout << setw(w) << left << "Enter planet to travel to: ";**

**getline(cin, planet);**

**cout << setw(w) << left << "Distance to planet (AU): ";**

**cin >> AU;**

**cout << endl;**

**//math**

**PSC = AU / conversion;**

**//report results**

**cout << fixed << setprecision(10);**

**cout << setw(w) << left << "Planet: "**

**<< right << setw(w1) << planet << endl;**

**cout << setw(w) << left << "Distance to (AU): "**

**<< right << setw(w1) << AU << endl;**

**cout << setw(w) << left << "Distance to (Parsec): "**

**<< right << setw(w1) << PSC << endl;**

**//ending**

**cout << "\nEnd of application.";**

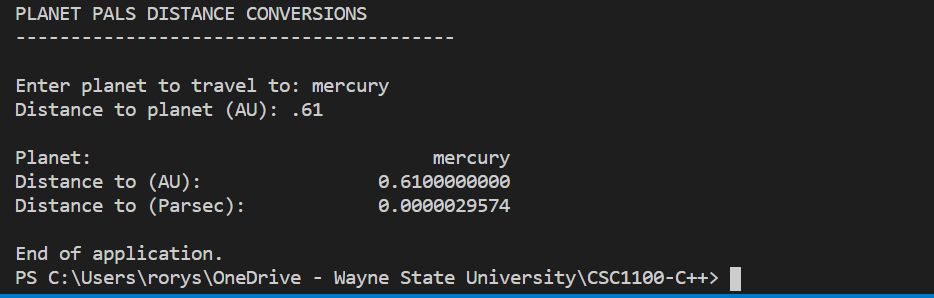
**}**

**If possible, format your code like this:**

**Font “Courier New”**

**Font size “9”**

**Bold**

**

**\* Copying-and-pasting C++ code to a Word document**

**macOS**

1) From within the C++ program, press **command-A** and press **command-C**.

2) From within the Word document, press **command-V**.

**Windows**

1) From within the C++ program, press **CTRL-A** and press **CTRL-C**.

2) From within the Word document, press **CTRL-V**.

**\*\* Copying-and-pasting C++ console application output to a Word document**

**macOS**

1) From the C++ console, press **shift-command-4-space**.

2) From within the Word document, **command-V**.

**Windows**

1) From the C++ console, press **ALT-PrintScreen**.

2) From within the Word document, press **CTRL-V**.