



Competency

In this project, you will demonstrate your mastery of the following competency:

- Utilize various programming languages to develop secure, efficient code

Scenario

You are doing a fantastic job at Chada Tech in your new role as a junior developer, and you exceeded expectations in your last assignment for Airgead Banking. Since your team is impressed with your work, they have given you another, more complex assignment. Some of the code for this project has already been completed by a senior developer on your team. Because this work will require you to use both C++ and Python, the senior developer has given you the code to begin linking between C++ and Python. Your task is to build an item-tracking program for the Corner Grocer, which should incorporate all of their requested functionality.



The Corner Grocer needs a program that analyzes the text records they generate throughout the day. These records list items purchased in chronological order from the time the store opens to the time it closes. They are interested in rearranging their produce section and need to know how often items are purchased so they can create the most effective layout for their customers. The program that the Corner Grocer is asking you to create should address the following three requirements for a given text-based input file that contains a list of purchased items for a single day:

1. Produce a list of all items purchased in a given day along with the number of times each item was purchased.
2. Produce a number representing how many times a specific item was purchased in a given day.
3. Produce a text-based histogram listing all items purchased in a given day, along with a representation of the number of times each item was purchased.

As you complete this work, your manager at Chada Tech is interested to see your thought process regarding how you use the different programming languages, C++ and Python. To help explain your rationale, you will also complete a written explanation of your code's design and functionality.

Directions

One of Python's strengths is its ability to search through text and process large amounts of data, so that programming language will be used to manage internal functions of the program you create. Alternatively, C++ will be used to interface with users who are interested in using the prototype tracking program.

Grocery Tracking Program

Begin with a Visual Studio project file that has been set up correctly to work with both C++ and Python, as you have done in a previous module. Remember to be sure you are working in Release mode, rather than Debug mode. Then add the CS210_Starter_CPP_Code and CS210_Starter_PY_Code files, linked in the Supporting Materials section, to their appropriate tabs within the project file so that C++ and Python will be able to effectively communicate with one another. After you have begun to code, you will also wish to access the CS210_Project_Three_Input_File, linked in the Supporting Materials section, to check the functionality and output of your work.

As you work, continue checking your code's syntax to ensure your code will run. Note that when you compile your code, you will be able to tell if this is successful overall because it will produce an error message for any issues regarding syntax. Some common syntax errors are missing a semicolon, calling a function that does not exist, not closing an open bracket, or using double quotes and not closing them in a string, among others.

1. Use C++ to develop a menu display that asks users what they would like to do. Include options for each of the three requirements outlined in the scenario and number them 1, 2, and 3. You should also include an option 4 to exit the program.

All of your code needs to effectively validate user input.

2. **Create code to determine the number of times each individual item appears.** Here you will be addressing the first requirement from the scenario to produce a list of all items purchased in a given day along with the number of times each item was purchased. Note that each grocery item is represented by a word in the input file. Reference the following to help guide how you can break down the coding work.
 - Write C++ code for when a user selects option 1 from the menu. Select and apply a C++ function to call the appropriate Python function, which will display the number of times each item (or word) appears.
 - Write Python code to calculate the frequency of every word that appears from the input file. It is recommended that you build off the code you have already been given for this work.
 - Use Python to display the final result of items and their corresponding numeric value on the screen.
3. **Create code to determine the frequency of a specific item.** Here you will be addressing the second requirement from the scenario to produce a number representing how many times a specific item was purchased in a given day. Remember an item is represented by a word and its frequency is the number of times that word appears in the input file. Reference the following to help guide how you can break down the coding work.
 - a. Use C++ to validate user input for option 2 in the menu. Prompt a user to input the item, or word, they wish to look for. Write a C++ function to take the user's input and pass it to Python.
 - b. Write Python code to return the frequency of a specific word. It will be useful to build off the code you just wrote to address the first requirement. You can use the logic you wrote but modify it to return just one value; this should be a fairly simple change (about one line). Next, instead of displaying the result on the screen from Python, return a numeric value for the frequency of the specific word to C++.
 - c. Write a C++ function to display the value returned from Python. Remember, this should be displayed on the screen in C++. We recommend reviewing the C++ functions that have already been provided to you for this work.
4. **Create code to graphically display a data file as a text-based histogram.** Here you will be addressing the third requirement from the scenario: to produce a text-based histogram listing all items purchased in a given day, along with a representation of the number of times each item was purchased. Reference the following to help guide how you can break down the coding work:
 - a. Use C++ to validate user input for option 3 in the menu. Then have C++ prompt Python to execute its relevant function.
 - b. Write a Python function that reads an input file (CS210_Project_Three_Input_File, which is linked in the Supporting Materials section) and then creates a file, which contains the words and their frequencies. The file that you create should be named *frequency.dat*, which needs to be specified in your C++ code and in your Python code. Note that you may wish to refer to work you completed in a previous assignment where you practiced reading and writing to a file. Some of your code from that work may be useful to reuse or manipulate here. The frequency.dat file should include every item (represented by a word) paired with the number of times that item appears in the input file. For example, the file might read as follows:


```
Potatoes 4
Pumpkins 5
Onions 3
```
 - c. Write C++ code to read the frequency.dat file and display a histogram. Loop through the newly created file and read the name and frequency on each row. Then print the name, followed by asterisks or another special character to represent the numeric amount. The number of asterisks should equal the frequency read from the file. For example, if the file includes 4 potatoes, 5 pumpkins, and 3 onions then your text-based histogram may appear as represented below. However, you can alter the appearance or color of the histogram in any way you choose.


```
Potatoes ****
Pumpkins *****
Onions ***
```

5. **Apply industry standard best practices such as in-line comments and appropriate naming conventions to enhance readability and maintainability.** Remember that you must demonstrate industry standard best practices in all your code to ensure clarity, consistency, and efficiency. This includes the following:
 - a. Using input validation and error handling to anticipate, detect, and respond to run-time and user errors (for example, make sure you have option 4 on your menu so users can exit the program)
 - b. Inserting in-line comments to denote your changes and briefly describe the functionality of the code
 - c. Using appropriate variable, parameter, and other naming conventions throughout your code

Programming Languages Explanation

Consider the coding work you have completed for the grocery-tracking program. You will now take the time to think more deeply regarding how you were able to combine two different programming languages, C++ and Python, to create a complete program. The following should be completed as a written explanation.

1. **Explain the benefits and drawbacks of using C++ in a coding project.** Think about the user-focused portion of the grocery-tracking program you completed using C++. What control does this give you over the user interface? How does it allow you to use colors or formatting effectively?
2. **Explain the benefits and drawbacks of using Python in a coding project.** Think about the analysis portions of the grocery-tracking program you completed using Python. How does Python allow you to deal with regular expressions? How is Python able to work through large amounts of data? What makes it efficient for this process?
3. **Discuss when two or more coding languages can effectively be combined in a project.** Think about how C++ and Python's different functions were able to support one another in the overall grocery-tracking program. How do the two function well together? What is another scenario where you may wish to use both? Then, consider what would happen if you added in a third language or switched Python or C++ for something else. In past courses, you have worked with Java as a possible example. What could another language add that would be unique or interesting? Could it help you do something more effectively or efficiently in the grocery-tracking program?

What to Submit

To complete this project, you must submit the following:

Grocery-Tracking Program

Submit your completed work as a ZIP file, including all Visual Studio project files that are required to run the program. Reference the Visual Studio Export Tutorial, linked in the Supporting Materials section, for guidance on how to download the necessary ZIP folder.

Programming Languages Explanation

Submit your completed explanation as a one-page Microsoft Word document with 12-point Times New Roman font, double spacing, and one-inch margins. Any sources should be cited according to APA style.

Supporting Materials

The following resource(s) may help support your work on the project:

[CS210 Project Three Input File](#)

Use this TXT input file to test your code as you work. It contains a list of different produce items that were purchased on a sample day, beginning in the morning when the Corner Grocer opened and continuing until the evening when the store closed.

[CS210 Starter CPP Code](#)

Add this CPP code to the CPP tab in your Visual Studio project file. You will need these functions to be able to communicate with the Python portion of this project's code.

[CS210 Starter PY Code](#)

Add this PY code to the PY tab in your Visual Studio project file. You will need these functions to be able to communicate with the C++ portion of this project's code.

[Visual Studio Export Tutorial](#)

This guide will walk you through how to download all of your work from Visual Studio as a ZIP folder.

Project Three Rubric

Criteria	Exemplary (100%)	Proficient (85%)	Needs Improvement (55%)	Not Evident (0%)	Value
Code: Overall Frequency	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Creates code to determine the number of times each individual item appears	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include validating user input, writing Python code to determine the frequency of all words, or writing C++ code to call the Python function	Does not attempt criterion	18
Code: Individual Frequency	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Creates code to determine the frequency of a specific item	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include validating user input, writing Python code to determine the frequency of a specific word, or writing C++ code to display a value returned from Python	Does not attempt criterion	18
Code: Histogram	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Creates code to graphically display a data file as a text-based histogram	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include validating user input, writing Python code to create a file, or writing C++ code to read a file and display a text-based	Does not attempt criterion	18

			histogram		
Industry Standard Best Practices	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Applies industry standard best practices such as in-line comments and appropriate naming conventions to enhance readability and maintainability	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include using proper naming conventions or in-line comments	Does not attempt criterion	8
Explanation: C++	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Explains the benefits and drawbacks of using C++ in a coding project	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include discussing a user interface or reviewing C++ challenges	Does not attempt criterion	10
Explanation: Python	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Explains the benefits and drawbacks of using Python in a coding project	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include discussing efficient analysis of data or reviewing Python challenges	Does not attempt criterion	10
Explanation: Multiple Languages	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Discusses when two or more coding languages can effectively be combined in a project	Shows progress toward proficiency, but with errors or omissions; areas for improvement may include discussing efficient tasks different languages can accomplish or examples of where they might be used	Does not attempt criterion	14
Articulation of Response	Exceeds proficiency in an exceptionally clear, insightful, sophisticated, or creative manner	Clearly conveys meaning with correct grammar, sentence structure, and spelling,	Shows progress toward proficiency, but with errors in grammar, sentence structure, and	Submission has critical errors in grammar, sentence structure, and spelling, preventing	4

		demonstrating an understanding of audience and purpose	spelling, negatively impacting readability	understanding of ideas	
Total:					100%