|  |  |
| --- | --- |
| Project NumberCSC134 Project | Project 2 |
| Project Name | Morse Code Project |
| Project Filename | Firstname\_Lastname\_Project2 (example: *Jessica\_Smith\_Project2*) |
| Chapter Review | Chapter 4Use only the concepts covered in Chapters 1 - 4 to complete this project. Failure to do so may result in a 0 for the assignment. Students are expected to complete the practice programs before starting the project. |
| Points | 50 |
| Assistance | Instructors and teaching assistants have been available to assist with all practice work. Students are now expected to complete this project without assistance from others (this includes receiving assistance from individuals inside or outside of CPCC). Students should consider projects as non-proctored exams. Please review the academic integrity policy on your syllabus.Please note: students can continue to receive assistance with practice work up to 5:00 p.m. on the due date (review the late period on the syllabus). |
| Project Description | |  |  | | --- | --- | | **Letter** | **Encoding** | | S, s | **. . .** | | T, t | **-** | | U, u | **. . -** | | V, v | **. . . -** | | W, w | **. - -** | | X, x | **- . . -** | | Y, y | **- . - -** | | Z, z | **- - . .** |   For this part of the project, you will add input validation to the customer payment and begin adding the functionality to convert words into morse code. Use the tables below to translate letters into morse code.   |  |  | | --- | --- | | **Letter** | **Encoding** | | A, a | **. -** | | B, b | **- …** | | C, c | **- . - .** | | D, d | **- . .** | | E, e | **.** | | F, f | **. . - .** | | G, g | **- - .** | | H, h | **. . . .** | | I, i | **. .** |  |  |  | | --- | --- | | **Letter** | **Encoding** | | J, j | **. - - -** | | K, k | **- . -** | | L, l | **. - . .** | | M, m | **- -** | | N, n | **- .** | | O, o | **- - -** | | P, p | **. - - .** | | Q, q | **- - . -** | | R, r | **. - .** |   **Instructions**   1. Modify your Project 1 code to display the number of dollars, quarters, dimes, nickels, and pennies that the customer should receive as change **only if** the payment amount is greater than the amount owed. Otherwise, display a descriptive error message. 2. Add a menu to offer the user a choice of processing a telegram bill or translating a message into morse code. The menu should be the first thing displayed when your program begins executing. The menu format should match the one in the **Sample Output** on page 2. 3. Validate the user’s menu choice. If the user enters any number other than 1 or 2, a descriptive error message should display. See **Sample Output** below for format. 4. If the user chooses the translate to morse code option, your program (for now) should prompt the user for a single letter and then display the cooresponding morse code. If the user’s input can not be translated into morse code, then a descriptive error message should display. Your program should translate both upper and lower case letters. 5. Save your **.cpp** file using the Firstname\_Lastname\_Project2 naming format. Upload in Moodle. Verify that your file actually uploaded. 6. **Bonus Pts (5pts)**. After completing all instructions above, add code to validate the user’s input for the letter. Your program should only translate the user’s input into morse code if it contains exactly one character. If the input contains more than one character, then your program should display a descriptive error message. You must follow all these instructions to receive full credit. |

# Sample Output

Welcome to Western Union Telegraph Company

1 – Process Telegram Bill

2 – Translate to Morse Code

Enter your choice: 1

Enter the name of the customer: Larry Smith

Enter street address: 122 Main Street

Enter city: Charlotte

Enter state: NC

Enter zip code: 23499

Enter the number of words sent: 157

Larry Smith

122 Main Street

Charlotte, NC 23499

Amount Owed: $47.50

Enter the amount received from customer: 5000

Denomination Number

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

Dollars 2

Quarters 2

Dimes 0

Nickels 0

Pennies 0

Welcome to Western Union Telegraph Company

1 – Process Telegram Bill

2 – Translate to Morse Code

Enter your choice: 2

Enter a single letter: A

The letter A translates to **. -**

Welcome to Western Union Telegraph Company

1 – Process Telegram Bill

2 – Translate to Morse Code

Enter your choice: 2

Enter a single letter: h

The letter h translates to **. . . .**

Welcome to Western Union Telegraph Company

1 – Process Telegram Bill

2 – Translate to Morse Code

Enter your choice: 8

8 is not a valid choice.

# Grading Score Card: 50

|  |  |  |
| --- | --- | --- |
| **OBJECTIVE** | **DESCRIPTION** | **POINTS** |
| Program is Menu Driven | Menu is the first thing displayed when the program runs. Menu contains all the required options and the format matches the one in the **Sample Output**. Menu choices are properly processed and validated. | 10 |
| Required Value(s) Correctly Calculated | Program displays correct output for all input values. | 20 |
| Algorithms are efficient. Code is clean, simple and easy to understand. | 5 |
| User Input Validated | User input for payment amount and letter are properly validated. | 10 |
| Documentation and Programming Style | * Multi-line comment present at the beginning of the program that contains: your **Name**, **Date** and **Purpose** which fully describes what your program does. * Adequate descriptive comments used throughout program. There must be at least four comments to receive credit for this step. * Proper indentation and spacing used throughout program. * Proper naming conventions used throughout program. | 5 |
| **TOTAL** |  | **50** |
| **Bonus Points** | Additional code was added to validate the user’s input for the letter. The program only translates the user’s input into morse code if it contains exactly one character. If the input contains more than one character, the program displays a descriptive error message. | **+5** |
| **Point Deduction** | Program does not compile without errors. | **-15** |