Final Project: Measurement Converter GUI Development Project – 100 Points for this assignment

Due: Tuesday 4/30 by midnight

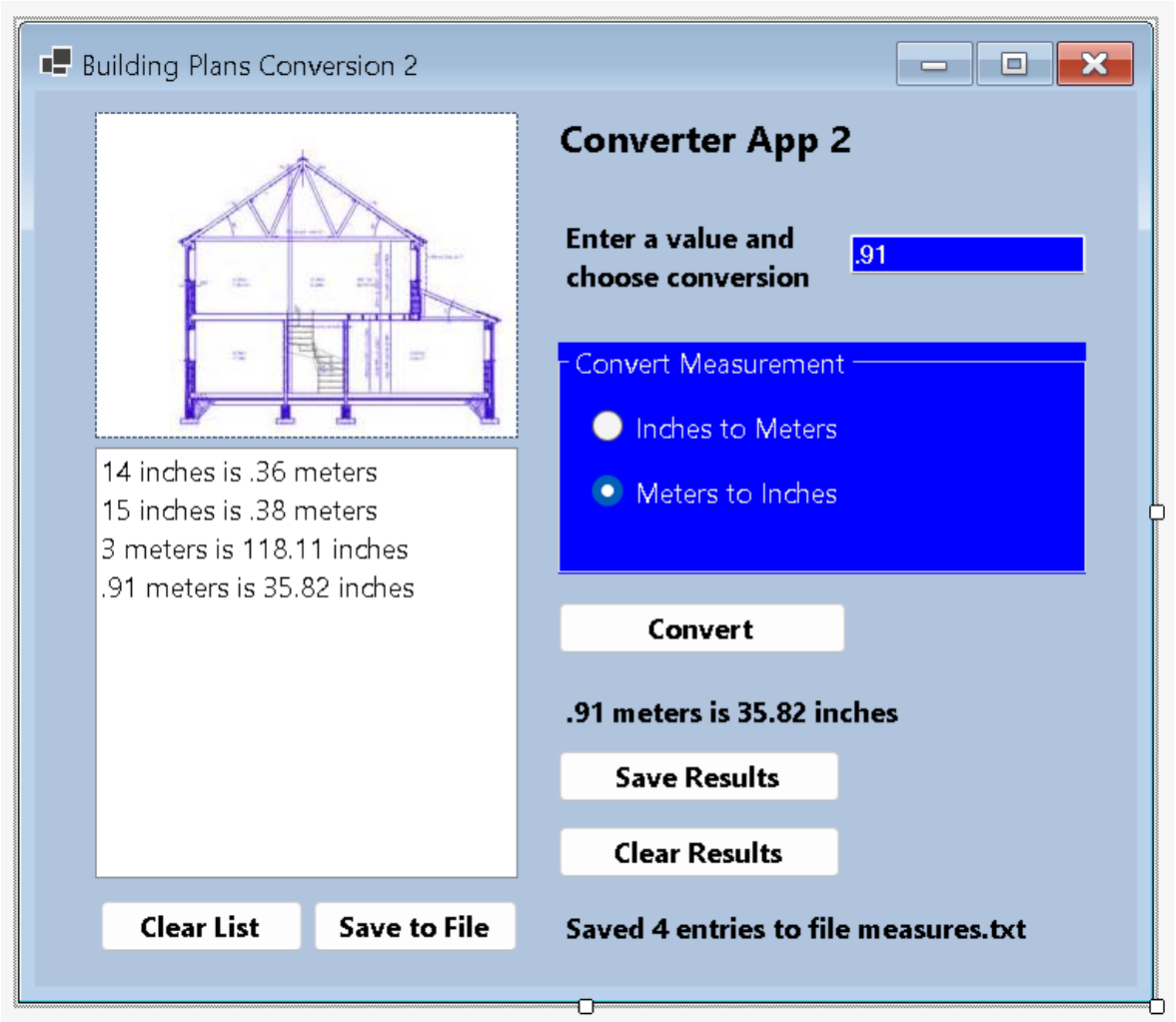
Project Overview

The Final Project is intended to show your mastery of all the approaches we have used in the class. Like the midterm project it is drawn from, this project has three parts: a standalone functional GUI-based app, a demonstration video, and a README markdown file.

GUI Development Details

You may choose whether to develop your application in VS/VB or in Python/Tkinter. Tkinter users can optionally use PAGE to help with layout/development. This project revisits the code from the midterm to add functionality. You can decide whether to copy and extend your midterm code for this project or to start from scratch.

You will provide a design for the following GUI:



This is the UI pictured after the user has entered a numeric value, clicked the Convert button, and has saved four results to the listbox shown.

Your design does not have to use the exact blue tones used for colors here; you can make your own design choices, but please notice that the design has to be readable, the font large, the font color a good contrast to the back color.

The house image (a .PNG file) is in a PictureBox (VB) or Frame (Tkinter). You can find the image attached to the assignment in CougarVIEW.

Use a RadioButton Group and two RadioButtons for the unit choices.

Clear Results button: Input textbox and display label for the conversion are cleared. The input textbox gets focus. The RadioButton check property is reset to "Inches to Meters". (This is the same set of requirements for when the form is first loaded.)

Convert button: Use the value in the textbox to perform the calculation, and show a label with the results as shown above. Note that the input can be either an integer or floating point value.

Use if statements to determine if the value entered in the textbox is a number, so it can be safely converted. Then determine that the value is positive. Based on the user choice in the radiobuttons, perform the conversion.

Here is a resource showing you all about conversions. There are many formulas, determine the correct math for both conversions. Display all calculations rounded to three decimal points.

Use a Message Box to display the following errors: If the converted value is negative OR If the value entered is not numeric

New elements (added to Midterm functionality):

Save Results button: Add the content of the last successful conversion string into the listbox under the picture.

Clear List button: Remove all entries from the listbox under the picture. This listbox should be empty when the program begins. The label under the Clear Results button should also be cleared (at startup or when Clear List is pressed).

Save To File button: Write all the lines in the listbox to a text file called measures.txt. If there is any data in measures.txt, it should be overwritten whenever Save To File is used. The bottom label under the Clear Results button should be updated to indicate the number of records written to the file.

Test Cycle:

- With no entry in the value field, press Convert
- Enter ABC and press Convert
- Enter -12 and press Convert
- Enter 14 (with Inches to Meters selected), press Convert, press Save Results
- Press Clear Results
- Enter 15 (with Inches to Meters selected), press Convert, press Save Results
- Enter 3 (with Meters to Inches selected), press Convert, press Save Results
- Enter .91 (with Meters to Inches selected), press Convert, press Save Results
- Press Save to File (information should be save to file, label updates)
- Press Clear List (listbox and save file label are cleared)

Project Deliverables

First Deliverable: Create a working GUI that performs the computations as laid out in the above discussion and ensure the form works accordingly. All code should be well documented. Use naming conventions for controls and variables appropriate to VB or Python/Tkinter. All project elements should be provided in your GitHub repo.

Second Deliverable: Create a video demonstration (likely 5 to 10 minutes) of using your application with the test cycle entries shown above. Open the measures.txt file to show its contents. Finally, open up your code in your development environment and discuss the key elements, including how you do the conversion, input checks, and file I/O. You can use any video capture approach you like. I'm a fan of Zoom, but you can use any video conference tool: Teams, Google Meet, etc. You could also use one of the following screen recorders:

- https://www.ispringsolu□ons.com/ispring-free-cam (free)
- https://www.ezvid.com/download (free)
- https://www.techsmith.com/video-editor.html (free 30-day trial)
- https://www.flashbackrecorder.com/ (free 30-day trial)

Please provide the captured video file in your GitHub repo, or alternatively, a link to the video accessible online. See me if you have any issues with the recording.

Third Deliverable: Create a README Markdown file for the Midterm Project, stored in your GitHub repo, including Project name and student name, development tools used, and any issues you encountered in development (none is a possible answer).

Project Delivery and Rubric

Submit your VB project files or Python files for your application, along with your recorded demonstration video or video link, and your README in a GitHub repo. Provide the GitHub repo link as a comment or as content in a text file.

Project grading rubric:

- 5 points – README as requested above
- 40 points – A thorough recorded video demonstration, as outlined above, showing test inputs and your discussion of the code
- 20 points – All objects requested represented in code and visually on GUI
- 20 points – GUI should run as requested in development environment (VB or Python) – all events should occur as requested
- 15 points – Cleanly formatted code with comments. This should include a header block containing student, project, and class names. Comments for functions/methods/classes, comments for key actions or any statements that may not be obvious in their function. Use descriptive variable names.

Note: Always cite what you write! If you get code or content from somewhere you must include at least a URL or other source identification. You must understand all the code you turn in. It is plagiarism (academic dishonesty) to use code or content, in part or in whole, written by other people without proper attribution. Failure to do so will result in a 0 on the assignment and may result in an academic misconduct report.

See Bruce for questions.