Mitchel Dauk

11/24/2024

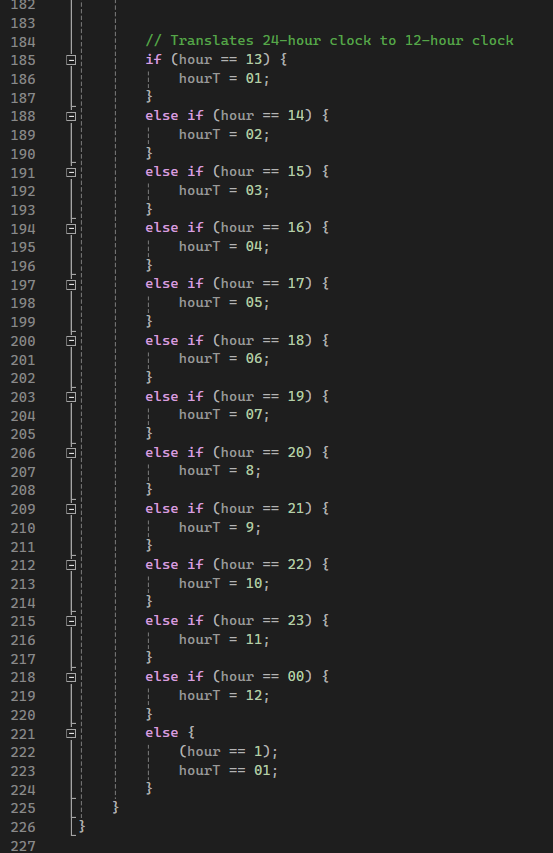
CS-499-18323

Computer Science Capstone

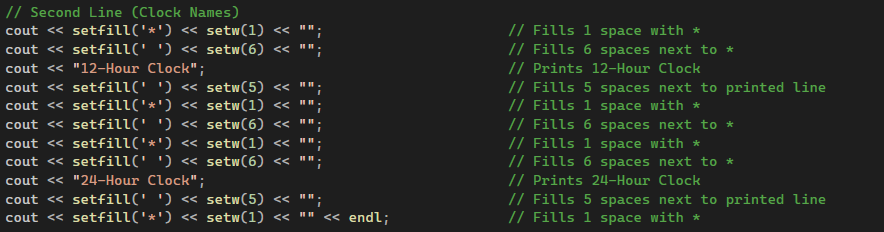
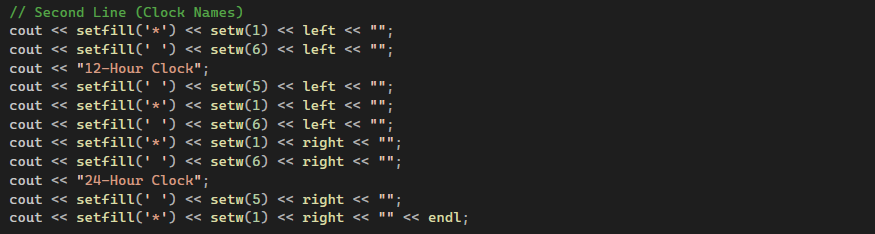
Southern New Hampshire University

Enhancement Two: Algorithms and Data Structure

This artifact is a digital clock that displays a random time. The user can choose to add an hour, minute, second, or exit the application. This was created earlier this year, in February. There were many algorithmic flaws that essentially bog down the program. Even though this program isn’t very big, on a larger scale, it will save a large amount of time when loading.

I selected this item because there was redundant code and some of the redundant code allowed the program to behave in an unnecessary way. This artifact shows that even the smallest changes can make a large impact when applied to a bigger scale.

First, there was a flaw calculating the random time. I initially started with a time that I chose. By adding one line of code to initialize the random time, I was able to print it correctly. There was a translation of code that I was able to delete completely by adding random numbers. The right picture shows 42 lines of code that were eliminated.

There also was redundant code while making the outline of the printed message. I originally had the characters pushed to the right or left. Looking back at my code, they were already defaulted to the left, so it made no sense to give them a location. Even the right characters followed the left side by default. The bottom pictures not only reduce the code, but also makes it look much cleaner.

I also eliminated white space. Essentially, a program with thousands of lines of white space will take more time and cause the programmer more time to review and fix.

Lastly, I added in-depth notes. Each line now has an explanation without referring to similar code displayed above. This saves time in the future without referring back to previous notes.

I believe I met the course outcomes because I took 242 lines of code and turned it into 205 lines of code. In the revised outcome, I also added many notes, so the amount of actual coded line is even less than 205. Not only does the clock work faster, it works correctly. The code also flows better and looks easier to read.

My outcome-coverage plan can be updated, once again, by adding another course outcome. I have now completed four of the five steps and added:

3. Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution while managing the trade-offs involved in design choices.

This artifact was more than I expected. When I first made this program, nearly nine months ago, I thought it was a masterpiece. I was impressed with my ability to code and thought the program ran well. Upon revisiting the artifact, I was amazed at how many flaws there were. Some code I originally thought was needed, turned out to be useless. I imagined a program with thousands of lines of code that didn’t serve a purpose, so it bogged down how efficient it ran. This can impact a number of problems in the workplace when it comes to more complex programs.

I learned that there are easier ways to program how a random number is initiated. Instead of explaining each step of the process and what to do if each number is displayed, there are better algorithms that will count for all numbers. If this pertained to a clock with thousands of numbers, the lines of code used would be off the charts and cause hours of coding for a programmer. I also learned that if there is a default setting, such as the characters and their placement in the printed section, that there is no need to initialize them into a spot. It’s like having an extra, unneeded coordinator that takes up more space.

I didn’t face many challenges other than figuring out what some of the code meant. This is the main reason I added code to all parts of the program. It actually took me longer to figure out what I was specifically doing in certain circumstances, which would have helped with better documentation. Overall, I feel like I have dramatically improved this artifact and used different algorithms to do so.