

CSC 230 Assignment 4

Due Sunday, December 15, 2019 at 11:55pm

Late submissions will not be graded.

Overview

This assignment covers low level C programming on the AVR architecture. The goal of this assignment is to Implement a Collatz sequence explorer using the C programming language. You are already familiar with the specifics, which are almost identical to Assignment 3 with just a few differences:

- The implementation must be written in C (instead of assembly).
- You must use three separate ISRs to handle (1) the button input, (2) the blinking of the cursor, and (3) the advancement of the Collatz sequence (and its count) to the next value.
- The SELECT button is associated with "speed memory". When pressed, it swaps the current speed with the one in memory. Initially, both the current speed and the speed memory are set to 0. When the SELECT button is pressed for the first time, the algorithm must remember the current speed and set the new speed to 0. When the SELECT button is pressed again, the algorithm (again) must remember the current speed and set the new speed to the previously remembered value.

Consider the following example. After starting the program, the user changes the speed, let's say to 5, and then presses the select button. At this point, your program would set the current speed to 0 and the remembered speed to 5. Your program would also immediately update the timer settings so that the Collatz sequence would stop advancing. Then the user changes the "new" current speed from 0 to 7, while the speed memory still contains 5. When the user presses select again, your program would set the current speed to 5 and the remembered speed to 7 and immediately update the timer settings so that the Collatz sequence now advances at speed 7.

Your code should be readable and, at minimum, contain header comments for each function (except main) describing the purpose of the function (and its arguments and return value, if applicable). You should also ensure that your source files contain header comments with your name and student number.

Consult the Assignment 3 specification for the remaining details on the expected functionality of the program.

Grading guidelines.

This assignment is worth 6% of your total grade, which is divided evenly among the following items:

- Credits screen (approximately one second).
- Cursor is blinking, which is done via ISR.
- Button checking is done via ISR.
- Buttons Left and Right move the cursor around.
- Buttons Up and Down change the number and select the new Collatz value on asterisk.
- Holding down a button does not result in multiple presses.
- Correct Collatz values (on LCD).
- Correct Collatz count (on LCD).
- The Collatz update speeds up and slows down as per specifications.
- Zero speed results in a complete stop.
- Select button remembers previous speed, swaps speeds correctly.
- The program is stable (no random behavior, garbage on screen, etc.).

Submission

Submit via conneX all .c and .h files needed to compile your solution (including all of the necessary LCD library files, even if you did not modify them). Your code must compile, upload and run correctly on the ATmega2560 boards in ECS 249 using the toolchain and methodology described in Lab 9. The buttons must follow the LCD shield version 1.1, where UP = 0x0C3, DOWN = 0x17C, LEFT = 0x22B, and SELECT = 0x316.

You are permitted to delete and resubmit your assignment as many times as you would like before the due date, but no submissions or re-submissions will be accepted after the due date has passed. You will receive a mark of zero if you have not officially submitted your assignment (and received a confirmation email) before the due date. Ensure that each submitted file contains a comment with your name and student number.

Ensure that all code files needed to compile, upload and run your program in ECS 249 are submitted. Only the files that you submit through conneX will be marked. The best way to make sure your submission is correct is to download it from conneX after submitting and test it. You are not permitted to revise your submission after the due date, and late submissions will not be accepted, so you should ensure that you have submitted the correct version of your code before the due date. conneX will allow you to change your submission before the due date if you notice a mistake. After submitting your assignment, conneX will automatically send you a confirmation email. **If you do not receive such**

an email, you did not submit the assignment. If you have problems with the submission process, send an email to the instructor **before** the due date.

The solution is worth 6% of your final grade and **must be your individual work**. You may discuss the assignment with your fellow students, but you must write your own code from scratch. Sharing code in any way (or receiving shared code), either electronically or over the shoulder of another student, will be considered plagiarism, even if the code is modified after being shared.

Late submissions will not be graded.