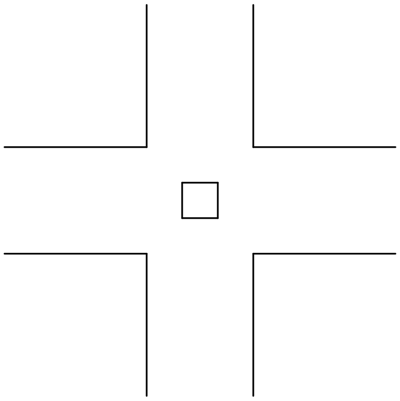
Background



The purpose of this assignment is twofold: (1) to gain some experience writing software that interacts with the world outside of the computer in a real-time fashion, and (2) to prepare a first version of the software that will later on be used to control a real-world device, a traffic light. The sketch at the right represents a traffic light at the intersection of two streets, one running north-south and the other running east-west. Each of the four faces of the light has three lamps (red, yellow, and green). The light operates on a very simple cycle. It provides the same amount of time for north-south traffic as for east-west traffic. That amount of time is adjustable, but an example would be 55 seconds green and 5 seconds yellow for a total cycle time of 60 seconds.

In lab next week you will modify your program to control a simulated traffic light device in a hands-on lab exercise using the FEH Proteus controller. The Proteus is a small controller based on a 32 bit, 100 MHz Freescale K60 ARM microcontroller.

Problem Statement

Write a program in C/C++ to simulate the logic for the operation of the traffic light. Your program needs to run in real time, which means that it needs to make use of available system timing functions to wait for a prescribed amount of time to elapse. Then after the waiting period is over, it must check the current status of the lights, determine what change(s) to make, and display the next light configuration.

Instructions

Represent

- Create a flowchart, algorithm, or pseudo code for solving the problem.

Plan

- Create a file named **APP_C31_1.cpp** .
- Outline the steps your program will take by adding comment statements to your file based on the flowchart, algorithm, or pseudo code.

Implement

- Write a complete C/C++ program **APP_C31_1.cpp** to perform the following tasks:
  - Prompt the user for the total time to run the program in seconds
  - Prompt the user for the length of the "green" cycle and the length of the "yellow" cycle, in seconds
  - Display the status of the N/S and E/W traffic lights in real-time
  - Stop when the specified total time has elapsed
- To figure out how much time has passed, call `time(NULL)` to store the time, and then later, call `time(NULL)` again and compare the two times. You may use this same function or the `sleep()` function to pause your program for a definite amount of time. Both of these functions are included within the `<unistd.h>` header file. See the class 31 slide deck for examples on how to use these functions.

- For testing purposes you may wish to use a relatively short cycle time for the traffic light, say 4 seconds green plus 2 seconds yellow (and thus, the red cycle time is a total of 6 seconds). You should terminate the program only after at least 30 seconds have elapsed and at least two full cycles have been completed.

Evaluate

- Using a watch/stopwatch/phone with seconds resolution, verify the sequence and timing of your simulated traffic light.

Document

- Create a single PDF that includes your flowchart/algorithm/pseudocode, your program, and sample output from running your program.
- Submit the PDF to Carmen according to the DAL.