

## Programming Assignments Overview – TOS

**Objective:** This semester your programming assignments will result in you writing a real operating system that will run on any PC. Your final project will be to use that operating system to write an application that will control a model train layout.

**TOS** – Train/Teachable/Tiny Operating System is an educational operating system running on a PC. TOS is written in C (99%) and x86 assembly (1%). All the files and Makefiles are provided for you. In addition test cases for each assignment are provided.

Your goal for the 5 assignments will be to implement the core functions of the TOS operating system and the train application. These assignments will provide you with a basic understanding of key components in an operating system and the issues facing operating system authors.

Assignments will be posted at: <http://www.my-sfsu.com/csc415/>

**Deliverables:** All assignments with source code deliverables shall be submitted by email to Robert Bierman at [bierman@sfsu.edu](mailto:bierman@sfsu.edu). You will attach the modified source code files to your email along with a one to three paragraph "journal" of your opinion and any issues regarding the assignment. The journal will not be graded but provides developmental information for future semesters. COMMENT your code! You do not need to comment any of the code given to you, but you should comment your code so others can easily understand what you are doing.

**Grading:** All assignments will be graded out of 10 points. 4 points for comments, 3 points for your code, and 3 points for functionality (i.e., it passes the test cases).

The following summarizes the assignments:

Assignment	Avail.	Due	Deliverable	Description
0	Now	9/5	None	The purpose of this assignment is to walk you through the installation of the Bochs PC emulator and make sure you are comfortable using it. You won't be writing any code for this assignment.
1	9/5	9/19	Source code	In this assignment you will install the core of TOS and implement some basic I/O functions.
2	9/19	10/10	Source code	Queues, dispatching, cooperative multi-tasking, and inter-process communication
3	10/17	10/31	Source code	Initializing interrupts and preemptive multi-tasking
4	10/31	11/16	Source code	Using interrupts and implementing serial communications
5	11/16	12/12	Demonstration	Write the application to control the train