Assignment #1 Part 1 - Set up your Linux environment and Basic Operating system Access

This is an individual assignment. However, you are encouraged to help (and seek help from) your peers (except sharing code, of course). Everything you do in this warm-up assignment is at the user-level (outside of the OS kernel). Your code must be compiled for Linux, and our reference machines will be the ones in the CSIL lab(Make sure your code compiles and runs correctly on these machines).

Part I: Observing the OS through the /proc file system /proc contains info on running processes in the system

The OS is a program that uses various data structures. Like all programs in execution, you can determine the performance and other behavior of the OS by inspecting its state - the values stored in its data structures. In this part of the assignment, we study some aspects of the organization and behavior of a Linux system by observing values of kernel data structures exposed through the /proc virtual file system.

The /proc virtual file system:
Linux uses the /proc file system to collect information from kernel data structures. The /proc implementation provided with Linux can read many different kernel data structures. If you cd to /proc on a Linux machine, you will see a number of files and directories at that location. Files in this directory subtree each corresponds to some kernel data structure. The subdirectories with numeric names contain virtual files with information about the process whose process ID is the same as the directory name. Files in /proc can be read like ordinary ASCII files. You can open each file and read it using library routines such as fgets() or fscanf(). The proc (5) manual page explains the virtual files and their content available through the /proc file system.

Requirements in detail:
In this part, you are asked to write a program to report the behavior of the Linux kernel. The program should print the following values on stdout:

Processor type
Kernel version
The amount of memory configured into this computer
Amount of time since the system was last booted

Your compiled executable **must be called proc_parse** and running it without any parameter should print out information required line by line.

Administrative policies

A note on the programming language:
C/C++ is the only choice for this assignment and all later programming assignments. We are not alone in this. Most existing operating system kernels (Linux and other UNIX variants) themselves are written in C; the remaining parts are written in assembly language. Higher-level languages (Java, Perl, ...), while possible, are less desirable because C allows more flexible and direct control of system resources.

Turn-in:

You are asked to electronically turn in your source files and a makefile. Attach a README file describing the executable names, special compiling instructions, or anything else special you want to let us know. The README file should be in plain text format. Instructions for electronic turn-in can be found on the class web page. You should indicate clearly in your README what tests you conducted.

Grading guideline:

The /proc/ parser is worth 20% of assignment 1.

Late turn-in policy:

Late turn-ins will be accepted for up to three days, with 10% penalty for each late day. No turn-ins more than three days late will be accepted.