CS342 Operating Systems - Spring 2023
Homework #1

Assigned: Feb 1, 2022.
Due date: Feb 10, 2022, 23:59. Document version: 1.0

- *Submit through **Moodle**. Make sure you start submitting one day before the deadline. You can overwrite your submission as many times as you wish. Late submissions will not be accepted (no excuse; no email will be accepted).*
- *Like all assignments, this assignment is for learning and improving your skills. If you do, you learn.*

Assignment

Please perform the following tasks and write a report at the end.

1. Read the first 2 chapters of the textbook this week.

2. Install Linux as soon as possible. Install the following Linux distribution and release on your own computer: **Ubuntu Desktop 64-bit 22.04** (latest long term supported distribution). It is essential that you install this distribution and release so that you will not have problems like "*was working on my machine*". You can install Linux on bare hardware, i.e., on a partition of your hard-disk. In this case, make sure you first **backup** all your important data so that you will not lose your data in case your computer does not boot up after installation.

   You can also install Linux in a **virtual machine** created in your computer. For this, you first need to install a virtualization software, VMware or VirtualBox (or some other virtualization software), on your computer. VirtualBox is free to use.

   You can help each other in installing Linux.

   You can download Ubuntu 22.04 LTS from:
   `https://ubuntu.com/download/desktop`

   Write briefly about your installation choices and experiences in your report. After installing Linux, start Linux and learn basic Linux usage. There are lots of guides and tutorials in Internet teaching basic Linux usage. You can benefit from them. In your report, write down the names of 10 Linux commands that you learned.

3. Find out and write down the location (pathname) where the kernel executable resides in the *default* local directory tree (starting with "/ ") of your Linux installation. Find out the version of your running kernel by

using the "`uname -r`" command. Write the version number in your report.

4. Download the source code of the Linux kernel (from **`kernel.org`**, for example). Download the version that is close to the version of your running kernel. After opening the tar package, change into the root directory of the downloaded kernel source code (it is in the directory where you downloaded the tar package), and in your report write the names of the subdirectories you see there.

5. In the source code of the kernel, find out the definition of **the system call table** (for 64 bit architecture). Write the pathname where you found it. Then, examine the table. Find out the system call names corresponding to system call numbers 0, 1, 2, 3, 4, 5, 6, 39, 120, and 150.

6. Use the **`strace`** command of Linux to trace the system calls made by some simple programs like cp, ls, etc. Use the manual page of `strace` to learn more about it (type `man strace`). Include sample output in your report. The "man" command provides help pages about Linux commands, system calls, and C library functions.

7. Use the **`time`** command to measure the time required to execute some programs like `cp`, `ls`, etc. It reports different times: real, user and sys. What are they? Write those values for different program executions.

8. Learn C Programming [1, 2]. Write a simple C program that implements a *doubly* linked list of integers, kept in sorted order (ascending). Generate and insert 10000 random integers into the list. Make sure you use pointers and `malloc()`. In your program, measure the total time it takes to insert 10000 numbers, using the **`gettimeofday()`** system call. This system call gives the current time in microseconds granularity.

   Write a simple **Makefile** to compile your program. A Makefile is a set of directives and commands specified in a file to compile a project. The following can be a starting point for your Makefile content. Be careful about TAB characters.

```
all: list
list: list.c
   gcc –Wall –g -o list list.c
clean:
   rm -fr list list.o *~
```

   This program is useful for you to warm up with C and set up your Linux environment to develop C programs. **Make sure you do it yourself**. Otherwise, it will be very difficult to do the projects. You will develop your programs in C and Linux. You will use the gcc compiler. Include the source code of your program in your report.

Submission

Submit a pdf file as your report which will include the information required for each question above.

Your report (pdf) should include your program C code and Makefile listing (even though we will not compile and run your program). Put your report into a directory named with your Student Id, and tar and gzip the directory. For example a student with ID 21404312 will create a directory named "21404312" and will put the report there. Then he/she will tar the directory (package the directory) as follows:

```
tar cvf 21404312.tar 21404312
```

Then he will gzip the tar file as follows:

```
gzip 21404312.tar
```

In this way he will obtain a file called 21404312.tar.gz. Then he will upload this file in Moodle.

Late submission will not be accepted (no exception). A late submission will get 0 automatically (you will not be able to argue it). Make sure you make a submission one day before the deadline. You can then overwrite it.

References

[1] The C Programming Language. B. Kernighan and D. Ritchie. Second Edition. Prentice Hall. 1998. **A must have C book; very useful.**
[2] Any Book on C.

Tips and Clarifications

- Make sure you learn a debugger like **gdb**, xxgdb, or the debugger of the IDE (integrated development environment) that you are using to develop programs (for example Eclipse IDE). Learn how you analyze core (memory) image dumped when a memory error occurs. For core image to be dumped you may need to set a core limit in the bash configuration file.
- There are a lot of documents and pages in Internet about how to develop, compile, run, and debug C programs in Linux OS. You can benefit from them.