

Term Project Assignment

Instructor: Prof. Seokin Hong (seokin@knu.ac.kr)

Due: 11:59pm December 8, 2019

1. Term Project Description

Your task is to design and implement a Linux program of your choosing. Your project may be graphical, text-based, or even file-based. It may be a game, or a math or science application, or an editor (such as vi editor), or a shell program, or anything else you choose. This is a wide-open assignment, and you are expected to be creative in your approach to it. You can use any type of Linux machine (e.g., desktop computer or micro computer such as Raspberry Pi). If you want to use the Raspberry Pi, we can provide a Raspberry Pi board for this term project. Unfortunately, however, we couldn't give you any technical support for that.

Each of your team will be assigned a mentor TA for your term project. Your mentor will provide most of support and guidance for your term project. Also, your mentor TA will be grading your term project.

This term project is very open-ended, but there are a few restrictions:

- Your program should be executable on Linux system.
- Your program should use the system calls that are covered in this course.
- Source code should be maintained on a public **GitHub repository**. You can use the GitHub for the version control and the collaboration. We will use the GitHub to check the contribution of each of your members.

2. Project Ideas

Here's a list of example of programs that you can get the project ideas.

1. tetris-bsd : A text-based tetris game.
2. Nudoku : terminal-based Soduku game for Linux.
3. Pacm4conesole : a free terminal clone of the famous Pac-man game.
4. nInvaders : a terminal version of popular GUI Space invaders game.
5. WeeChat : a IRC (Internet Relay Chat) program
6. Irssi : another IRC (Internet Relay Chat) program
7. talk : a chat program that allows two users to type messages to each other
8. lftp : command line ftp client.
9. ftpserver
10. Rsync : utility that provides fast incremental file transfer. Frequently used for data backup
11. Vim : text editor
12. Nano : feature-rich CLI text editor
13. Alpine : email client
14. htop : interactive process viewer
15. Midnight commander : a visual file manager
16. Etc
 - a. Chat application with 2 users or multiple users
 - b. File transfer application
 - c. Online Pong game
 - d. Shell
17. Anything you are interested in ...

3. Team building

- a. Team size : 1~3 students
- b. **Name** your team ☺
- c. Assign a **role** for each member (e.g., team leader, scheduler)
- d. Discuss efficient ways to collaborate each other
- e. Clearly **define the tasks** of each member. All members should implement some parts of your program.
- f. Instructor and TAs will help you to build your team if you have any trouble finding your team members.

4. How to evaluate individual student

- a. **Group performance (80%)**
 - i. **Presentation (60%)**
 1. **Peer-assessment (30%)** : all students except your team members will evaluate the final output of your project.
 2. **Instructor evaluation (30%)** : Instructor and TAs will also evaluate your final output.
 - ii. **Proposal and Final Report (20%)** : The report will be used to evaluate the design and implementation of your program. Thus, in the report, you need to describe overall architecture of the program and how each key feature of the program is implemented (e.g., used library functions or system calls).
- b. **Individual performance (20%)**
 - i. **Self-assessment (15%)** : Each team member documents and evaluates his/her own contributions to the team.
 - ii. **Instructor evaluation (5%)** : Instructor will assess the individual performance with the contribution history of the project on GitHub.

5. Schedule

Here is a guide for the term project schedule

- a. **Week 9 (Oct. 28 – Nov. 3)**
 - i. Build your team.
- b. **Week 10 (Nov. 4—Nov. 10)**
 - i. **Nov. 4 :** Your mentor will be determined. TA will provide a short tutorial about the GitHub. If your team needs any hardware equipments (e.g., Raspberry Pi) for your project, ask the instructor to borrow them.
 - ii. **Oct. 4 —Nov. 10 :** You can discuss the **project idea** and schedule with your mentor. Determine the role of each member. Start to **define the features** and to design the architecture of your program. **Allocate each task** to your team members.
- c. **Week 11 (Nov. 11 —Nov. 17)**
 - i. **Nov. 11 :** Submit a **project proposal** (1 page) which includes the **project topic, team name, team members, the role of each member, and brief schedule**.
- d. **Week 12 (Nov. 18 —Nov. 24)**
 - i. **Nov. 14 :** Meet your mentor to report your progress and to discuss any issues.
- e. **Week 13 (Nov. 25 —Dec.1)**
 - i. **Nov. 21 :** Meet your Mentor to report your progress and to discuss any issues.
- f. **Week 14 (Dec. 2 —Dec. 8)**
 - i. **Dec. 8 :** Deliver your final result of the term-project. Upload the final version of your program source code, presentation file, final report, and demo video to the lms system. If the video file size is too large, you can upload the demo video to Youtube. In that case, please let me know the link to the video.
- g. **Week 15 (Dec. 9 —Dec. 13)**
 - i. **Dec. 9 :** Project presentation.

Late Day Policy

Late submissions are not accepted

Plagiarism

No plagiarism will be tolerated. If the assignment is to be worked on your own, please respect it. If the instructor determines that there are substantial similarities exceeding the likelihood of such an event, he will call the two (or more) students to explain them and possibly to take an immediate test (or assignment, at the discretion of the instructor) to determine the student's abilities related to the offending work.