

**Gebze Technical University**  
**Department Of Computer Engineering**  
**CSE 312 /CSE 504 Spring 2020**  
**Operating Systems**  
**Makeup Project**  
**Due Date: 26.07.2020**  
**No Late Submission!**

Your makeup project will be similar to HW1, so this project will not depend on any other projects. You will extend the original SPIM OS with some systems calls, you will also write a number of assembly programs.

The operating system that comes with SPIM is very primitive. It only supports a few trivial system calls, such as reading and printing, which are defined in `/CPU/syscall.h`.

First, download and study the SPIM package very carefully. Then, you will write and test the following MIPS assembly files.

1. `sort.asm` : Reads from standard input positive integers (negative integer indicates the end of list), then sorts the integers in increasing order and prints the sorted integers to standard output.
2. `random.asm` : produces 100 random positive integers and prints them to the standard output. The last integer is a negative number.
3. `search.asm` : Reads from standard input an integer, then reads a number of positive integers. Makes a search on the list. prints the result to the standard output.
4. `type.asm`: works like UNIX `type` command. Prints the contents of a file to the standard output. At the end prints -1.
5. `ls.asm`: very simplified `ls` command of UNIX. prints the contents of the directory to the standard output. Note that there is only one directory in our system.
6. `Shell.asm`: You will write a shell program that will be very similar to `tcsh`. Note that current SPIM OS does not support many things such as file processing (read, write, create) and multiprocessing. You will have to add many system calls to the new OS. You will change the SPIM source code in two places: `/CPU/syscall.h` and `/CPU/syscall.cpp`. Do not modify any other SPIM code other than these files. Your shell will include the following features
  - a. Standard piping such as `random | sort | search or type a.txt | sort`
  - b. Standard redirection such as `sort > res.txt < input.txt`
  - c. Back ground tasks such as `type a.txt &`
  - d. Standard screen and keyboard read/write.

Below are the instructions for homework submission

1. Download and Install Vmware Player from official site.
2. Download and install our virtual machine from [https://drive.google.com/open?id=1YppX3lNkyTsHV\\_lvA4w9TomNCUkpLeEg](https://drive.google.com/open?id=1YppX3lNkyTsHV_lvA4w9TomNCUkpLeEg)
3. Your submission includes only the files below

- (20 points) ProjectReport.pdf: your detailed project report that includes
  - How you implemented piping, redirection, standard input and output, screen input and output
  - How you added systems calls to the new OS. For each system call, include information about the functionality and the parameters
  - How you handled the programs running in parallel
  - How you handled the file creation, opening, closing, etc.
  - Any other points that you like to mention
- (40 points) shell.asm
- (5 points each) sort.asm, random.asm, search.asm, ls.asm, type.asm
- (20 points) syscall.cpp and syscall.h