Intelligent Systems Engineering Department
The National School of Artificial Intelligence
Semester 2 — 2024/2025

AI224: OPERATING SYSTEMS

Lab 1

Description. This lab aims to introduce you to Unix and to open-source GNU/Linux operating systems. You will learn the difference between Unix and GNU/Linux and start using one of many GNU/Linux distributions (e.g., Debian, Ubuntu, etc).

Task 1

In this first task, the instructor will use some slides to

1.1. Briefly go over the history of Unix and how GNU/Linux came into existence.

1.2. Present the different ways you can run a GNU/Linux OS on a PC.

Task 2

In this second task, you will start using your favorite GNU/Linux distribution.

2.1. Take 5 minutes to explore and browse over the system.

2.2. Open a terminal (Ctrl+Alt+T) and start running the following commands:

1. `uname`
2. `whoami`
3. `hostname`
4. `pwd`
5. `ls`
6. `cd`
7. `man`
8. `whatis`
9. `which`
10. `touch`
11. `nano`
12. `cat`
13. `head`
14. `tail`
15. `mv`
16. `rm`
17. `cmp`
18. `diff`
19. `grep`
20. `mkdir`
21. `rmdir`
22. `chmod`
23. `df`
24. `du`
25. `ps`
26. `top`
27. `kill`
28. `ping`
29. `wget`
30. `ifconfig`
31. `find`
32. `history`
33. `clear`
34. `su`
35. `passwd`
36. `sudo`
37. `useradd`
38. `userdel`
39. `zip/unzip`
40. `tar/untar`

Task 3

In this task, you will learn how to write and run your first “Hello, World!” C program on a GNU/Linux distribution through the command line interface. For this, you will need a text editor, e.g., nano, a C compiler (GNU C Compiler), and the `chmod` command. Follow the below steps:

- ☞ Create a new file to write your source code (e.g., Prog.c).
- ☞ Write your “Hello, World!” code using the C programming language.
- ☞ Use `gcc` to compile Prog.c and generate the executable (e.g., Prog.exe).
- ☞ Add execute rights to the generated file using `chmod` command.
- ☞ Run your program using the command `./prog.exe`.

Task 4

If the lab session is not over, you can try to work on this task. Otherwise, you can work on it at home. The instructor will check your work during the next session.

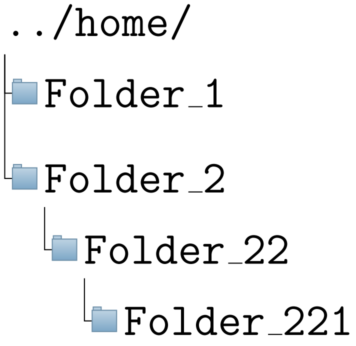
4.1. Write a loop-free C program that asks the user to guess an integer value between 0 and 100. The program gives the user 10 tries to guess the right value (e.g., 21).

4.4. Write a C program that performs matrix multiplication.

4.5. You may want to learn other GNU/Linux commands, try these ones:

1. `date`
2. `who`
3. `chgrp`
4. `chown`
5. `more`
6. `cp`
7. `id`
8. `finger`
9. `lscpu`
10. `pstree`
11. `iwconfig`
12. `netstat`
13. `ssh`
14. `echo`
15. `apt`
16. `reboot`
17. `shutdown`
18. `systemctl`
19. `sort`
20. `adduser`

4.6. Use GNU/Linux commands to create the following directory tree:


../home/
Folder_1
Folder_2
Folder_22
Folder_221