CSE4510 Operating Systems Lab

Shell Scripts Recap & Practice

Salman Shamil

United International University (UIU)

Summer 2025

- **Recap Lab 01–04**
  - Any questions / confusions
- **Shell Scripting**
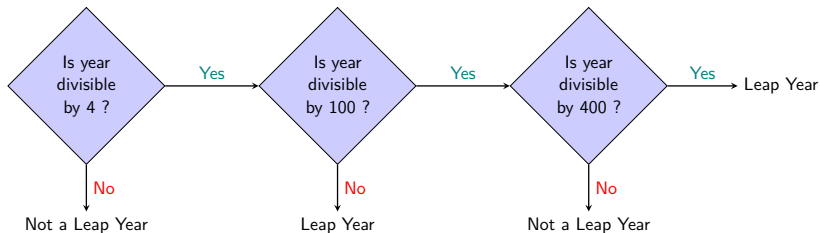 - Practice

Shell Script: Summary

- Navigation
- Basic file operations
- Search files and contents
    - use of grep and find
- Programming in shell
    - functions & variables
    - conditional statements
    - loops

Practice Problem # 01

1. Use grep to print all lines from sample_1.txt that contain a magic string. A magic string is any five-character alphanumeric word composed exclusively of the letters {e, n, r, v} and digits 0 through 4. The search should be case-insensitive. For each match, print the matching line number along with one line after the matching line.

2. Write a single command that finds all .log files in the sample_directory that were modified within the last 7 days, and for each file, prints the number of lines containing the word ERROR (case-insensitive).

3. Update your command in subtask 2 so that it considers only the first 10 lines of each file.
   - Hint 1: You can use sh -c '<your_command>' within -exec
   - Hint 2: sh -c can also access arguments with $1, $2, ...

Practice Problem # 02

1. Write a shell script that takes three input integers (values are given in non-decreasing order, *i.e.*, $a \leq b \leq c$) and determines if those three values can be three sides of a right triangle (one angle is a $90°$). [Hint: Use Pythagorean theorem]

2. Write a bash script that takes three integer inputs and prints the largest value among them.

3. Write a bash script that takes a year as input from command line and determines if the given year is a leap year or not.

Practice Problem # 03

Let's go back to the sample_directory. Write a bash script that loops over all .log files in sample_directory and its subdirectories. For each .log file, counts how many lines contain the word ERROR and WARNING (case-insensitive), and prints a summary line per file as below:

```
sample_directory/app1.log: ERROR=10, WARNING=5
sample_directory/app2.log: ERROR=0, WARNING=0
...
```

At the end, prints the total number of ERROR lines and WARNING lines across all .log files.

```
Total ERROR lines: 17
Total WARNING lines: 6
```