Overview

- ***System health check tool: The system health check tool is designed to provide a Health Report for a Linux system it performs checks on some aspects of the system and resources, such as disk space, memory usage, running services and the recent system updates. The tool aims to assist system user in maintaining optimal system performance, identifying potential issues, and ensuring that the system is up-to-date with the latest security patches.***
- ***Backup Tool: The Backup Tool is designed to provide a user friendly interface for performing backups of user specified directories***

System health check Functions:

- **check_disk_space:**
  - Purpose: Monitors disk space usage and recommends actions if a disk is almost full.
  - Customization: Allows the user to set a custom disk usage threshold.
  - Functionality: Provides summaries of disk space usage for each file in the system.
- **check_memory_usage:**
  - Purpose: Checks and categorizes memory usage status into different levels (low, moderate, high) based on the usage parentage.
  - Customization: Allows the user to set a custom memory usage threshold.
  - Functionality: Lists top memory-consuming processes if usage exceeds the threshold, and suggests to kill the unnecessary ones.
- **check_running_services:**
  - Purpose: Lists currently running services and recommends updating them.
  - Functionality: Displays the running services using the command **systemctl** and prompts the user for updates.
- **update_and_patch_services:**
  - Purpose: Checks for available updates and upgrades system services.
  - Functionality: Utilizes **apt-get** command for updates and provides a visual progress bar using **PV**.
- **check_system_updates:**
  - Purpose: Displays recent system updates and recommends updating the entire system.
  - Functionality: Retrieves and displays updates from the **apt** history log.

- **log_message:**
    - Purpose: Logs messages, including timestamps, to a log file (**system_health_log.txt**).
    - Functionality: Ensures a record of health check results is maintained.

Backup Tool Functions:

- welcome_message:
    - Purpose: Displays a welcome message and brief instructions about the tool.
    - Functionality: Informs users about the tool's capabilities, default settings, and purpose.
- log_message:
    - Purpose: Logs messages, including timestamps, to a log file (backup_log_${current_datetime}.log).
    - Functionality: Maintains a record of backup-related events and errors.
- perform_backup:
    - Purpose: Starts the backup of the specified directory.
    - Functionality: Creates a compressed pack up file using **tar** command, uses the PV command to visualize progress bar and logs backup details.
- change_backup_directory:
    - Purpose: Allows users to customize the directory where the backup file will be saved.
    - Functionality: Provides options to change the directory, create a new one if it doesn't exist, or continue with the default directory.

- Commands used:
    - (**df -h**) will display the disk space usage in a human-readable way.
    - (**Here string "<<<"**) will you take the output of another command and input it into another command as a string (it has been used in a loop).
    - (**Suffix Pattern Matching "%" Operator**) this operator is used for pattern removal, I used it to remove the percentage sign in my script.
    - (**Systemctl list-units --type=service --state=running**) this command is used to list the units (services, sockets, etc.) from the system d and filter them paste on their type service and the state running.
    - **(Pipe Viewer "pv")** this command will visualize the progress in the pipeline, and it has been used it to visualize the updates**.**
    - **(sudo)** will run the command with the super user privileges.
    - **(grep)** will search for patterns within the profile text.
    - **(head)** will display the first num of lines in a file.
    - **(awk)** is a text processing tool, (the -F option) specific files the field separator

- **(tar)** is a tool for compressing files into a single achieve.
- **The regular expression has been used multiple times in the script.**

Performance Analysis:

- Readability:
  - The script is well-organized into modular functions, enhancing readability, extensibility and reusability.
- User Interaction:
  - Provides clear and informative prompts, allowing users to customize thresholds, make informed decisions and sit the wanted directories with no trouble.
- Logging:
  - Log messages are time stamped, providing a detailed history of health checks and events and errors to a dedicated log file.
- Optimizations:
  - Implements optimizations like visual progress bar (pv) in different places.

Conclusion: I gained a lot of experience in this assignment in how to use Linux terminal bash shell. And I really liked the tools and the commands and the idea of combining more than one command to output the wanted outcome.

Future Work: make a man page for the tool, learn how to make a stress test to test the tool at its edge cases, send reports throw email from time to time without interactions from the user.