

# OS ASSIGNMENT BY GROUP-71

## GROUP MEMBERS CONTRIBUTION:

- 1) Ayush Singhal: Implemented signal handling and segmentation fault part.
- 2) Sanjeet Kumar Patel: Implemented segmentation fault and memory allocation part.

This assignment was a combined effort from both of us where we both have contributed to our best.

## SUMMARY OF SIMPLE SCHEDULER IMPLEMENTATION

- The code constitutes a simple ELF loader, capable of loading an ELF executable file, handling page faults, and managing memory allocation.
- It defines structures to hold ELF header (`Elf32_Ehdr`) and program header (`Elf32_Phdr`) information and sets up variables to track page faults, page allocations, and fragmentation.
- The main function (`load_and_run_elf`) opens the ELF file specified as a command-line argument, reads the ELF header and program header table, and retrieves the entry point for the ELF program.
- It defines a custom signal handler (`signal_handler`) for `SIGSEGV` (segmentation fault) to manage page faults. Upon a page fault, the handler identifies the segment causing the fault and allocates memory, utilizing `mmap`.
- The allocation strategy involves checking for available space in the segment where the fault occurred and then using `mmap` to allocate a new page with read, write, and execute permissions. It reads the required content from the ELF file and maps it to the allocated memory.
- After the ELF program finishes execution, it displays the result, total page faults, total page allocations, total internal fragmentation in bytes, and total internal fragmentation in kilobytes.
- The `main` function validates the command-line arguments and sets up the custom signal handler for `SIGSEGV` before calling the `load_and_run_elf` function.

## Key points:

- ELF file loading and execution.
- Custom signal handler for handling page faults (`SIGSEGV`).
- Memory allocation using `mmap` upon a page fault.

- Tracking and reporting of page faults, page allocations, and internal fragmentation after the ELF program execution.

**GITHUB LINK:**

[LINK](#)