**LAPORAN PRAKTIKUM**

**SISTEM OPERASI RD**

**MODUL 3**

**Oleh :**

**Muhammad Yusuf (122140193)**



**Program Studi Teknik Informatika**

**Institut Teknologi Sumatera**

**2024**

# **Daftar Isi**

[**Daftar Isi** 2](#_Toc164803214)

[**1.** **Dasar Teori** 3](#_Toc164803215)

[**2.** **Ulasan Hasil & Jawaban** 4](#_Toc164803216)

[**3.** **Kesimpulan dan Saran** 6](#_Toc164803217)

# 

# **Dasar Teori**

Tulisan di atas membahas mengenai System Call, yang merupakan metode bagi program komputer untuk meminta layanan dari kernel sistem operasi di mana program tersebut dijalankan. Berikut adalah rangkumannya:

System Call (Panggilan Sistem)

1. Definisi: System call adalah metode program komputer untuk meminta layanan dari kernel sistem operasi.
2. Antarmuka: System call menyediakan antarmuka yang terdefinisi antara program pengguna dan sistem operasi melalui API.
3. Fungsi Utama:
4. Interface: Program membuat permintaan dengan memanggil fungsi tertentu, dan sistem operasi menjalankan layanan yang diminta.
5. Protection: System call digunakan untuk mengakses operasi istimewa yang tidak tersedia untuk program pengguna normal, dengan hak istimewa untuk melindungi sistem dari akses berbahaya atau tidak sah.
6. Kernel: Saat system call dibuat, program dialihkan dari mode pengguna ke mode kernel untuk mengakses sumber daya sistem.
7. Context Switching: System call memerlukan pengalihan konteks yang dapat menimbulkan biaya tambahan.
8. Error Handling: System call dapat mengembalikan kode kesalahan yang harus ditangani program dengan tepat.
9. Synchronization: System call digunakan untuk menyinkronkan akses ke sumber daya bersama.
10. Jenis System Call:
11. Process Control: Mengendalikan proses dalam sistem operasi.
12. File Management: Mengelola file dan direktori.
13. Device Management: Mengelola perangkat keras.
14. Information Maintenance: Mengelola informasi sistem.
15. Communication System: Mengelola komunikasi antar proses dan jaringan.
16. Memory Management: Mengelola memori dalam sistem operasi.

**System Call READ**

System call read digunakan untuk membaca data dari suatu file descriptor atau input standar. Data yang dibaca akan disimpan di dalam buffer yang disediakan oleh pengguna.

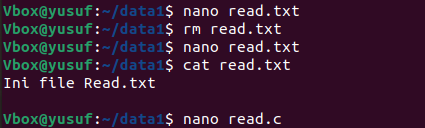
**System Call Exec**

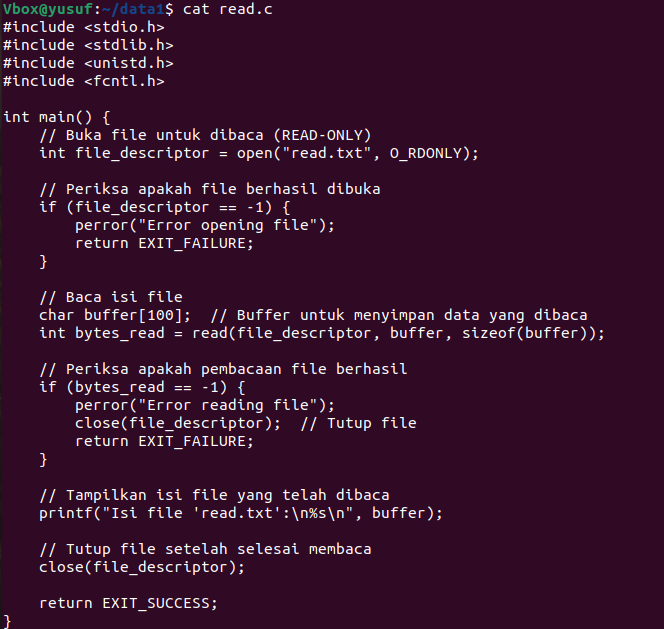
System call EXEC digunakan untuk menggantikan proses yang sedang berjalan dengan proses baru yang dieksekusi. Proses baru ini akan mengambil alih proses yang sebelumnya sedang berjalan.

# **Ulasan Hasil & Jawaban**

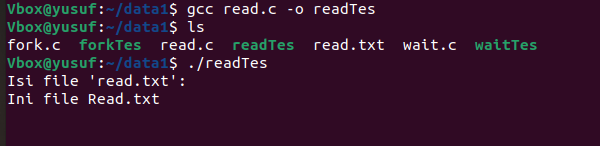
**Percobaan 1 READ**

1. Buat file dengan ekstensi C dengan nama ”read.c” dan file dengan code untuk mempraktikan contoh system call jenis READ.



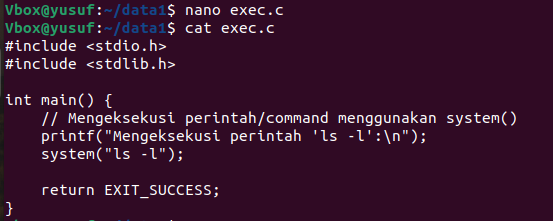


1. Jalankan perintah ”gcc read.c -o readTes” dan run perintah ”./readTes”.

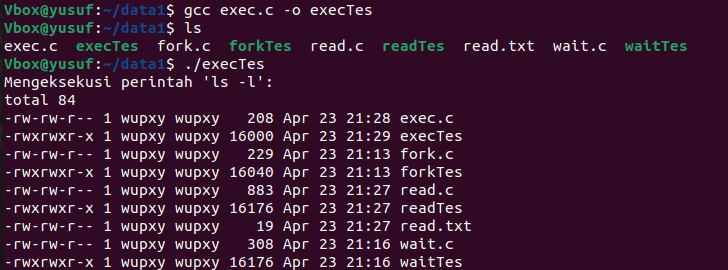
****

**Perobaan 2 EXEC**

1. Buat file dengan ekstensi C dengan nama ”exec.c” dan file dengan code untuk mempraktikan contoh system call jenis EXEC.



1. Jalankan perintah ”gcc exec.c -o exec” dan run perintah ”./exec”.

****

# **Kesimpulan dan Saran**

Setelah melakukan praktikum yang fokus pada System Call READ dan EXEC, dapat disimpulkan bahwa kedua System Call ini memiliki peran yang penting dalam interaksi antara program pengguna dan kernel sistem operasi. System Call READ memungkinkan program untuk membaca data dari file atau perangkat input tertentu, sedangkan System Call EXEC digunakan untuk menjalankan program baru di dalam proses yang sedang berjalan.