**LAPORAN PRAKTIKUM**

**SISTEM OPERASI RB**

**MODUL 5**

**Oleh :**

**Indah Mutiara (121140158)**



**Program Studi Teknik Informatika**

**Institut Teknologi Sumatera**

**2023**

# **Daftar Isi**

[**Daftar Isi 2**](#_ajo0u2o5viv0)

[**1. Dasar Teori 3**](#_83n9bu57nsc)

[**2. Ulasan 4**](#_68mmq9rf5dx5)

[**3. Hasil dan Jawaban 5**](#_65bygpwizg8j)

[**4. Kesimpulan dan Saran 9**](#_tyzeyjhy25fz)

# 

# **Dasar Teori**

Dalam praktikum modul ini, materi yang akan dipelajari mencakup konsep input/output dan deskriptor file. Pada dasarnya, input/output adalah suatu proses yang membutuhkan masukan dan menghasilkan keluaran ke sistem atau pengguna. Sementara itu, file descriptor digunakan dalam sistem operasi Linux untuk memfasilitasi komunikasi dengan file.

1. Proses Input/Output

Dalam sistem operasi, terdapat situasi di mana suatu proses memerlukan masukan dan menghasilkan keluaran sebagai hasil dari proses tersebut. Dalam konteks sistem operasi Linux, perintah atau command yang diberikan melalui Shell dapat dieksekusi sebagai sebuah program yang disebut proses. Setiap kali perintah dijalankan, kernel Linux akan membuat suatu proses dan memberikan nomor identifikasi yang disebut PID (Process Identify). Input/output dapat bervariasi dan mencakup beberapa jenis berikut ini :

* Keyboard (input)
* Layar (output)
* Files
* Struktur data kernel
* Peralatan I/O lainnya (misalnya network)

1. File Descriptor

File descriptor pada sistem operasi Linux memiliki peran penting dalam berinteraksi dengan file yang direpresentasikan sebagai angka (dalam bentuk bilangan bulat). Sesuai dengan standar file descriptor, terdapat tiga proses yang terbentuk, yaitu:

* 0 = keyboard (standard input)
* 1 = layar (standard output)
* 2 = layar (standard error)

Selain itu, file descriptor juga memiliki 3 perintah yang dapat digunakan untuk berinteraksi/berkomunikasi dengan file, yaitu sebagai berikut :

* Perintah “sort”

Perintah ini digunakan untuk mengurutkan masukan berdasarkan urutan nomor ASCII dari karakter.

* Perintah “cut”

Perintah ini digunakan untuk melakukan “cut” atau memotong kolom tertentu dari suatu baris masukan.

* Perintah “uniq”

Perintah ini digunakan untuk menghilangkan baris baris berurutan yang mengalami duplikasi.

# **Ulasan**

Berdasarkan soal ……. *dijelaskan* :

Percobaan 1

1. Soal 1

Bagaimana perbedaan antara command more dan less, dan implementasikan pada terminal.

1. Soal 2

Buatlah sebuah file yang berisikan daftar nama daerah dengan menggunakan file descriptor dan redirection dengan mengimplementasikan penggunaan standar input, standar output dan standar error.

1. Soal 3

Tampilkan isi file tersebut dengan menggunakan implementasi dari pipeline dan filter yaitu perintah.

* Grep
* Wc
* Uniq
* Sort
* Head
* Tail
* Cut
* Paste
* Awk
* Sed

# **Hasil dan Jawaban**

Soal 1 Bagaimana perbedaan antara command more dan less, dan implementasikan pada terminal.

| 3.1.1 | 3.1.2 |
| --- | --- |
| 3.1.3 | 3.1.4 |

Penjelasan :

Dalam rangka menuntaskan tugas soal 1, akan dicoba perintah untuk membandingkan perbedaan antara command more dan less. Salah satu contoh perintah yang akan diuji adalah "more tinggiBadanKelas.txt". Fungsi perintah ini adalah untuk membaca atau menampilkan isi dari file tinggiBadanKelas.txt yang telah dibuat sebelumnya. Informasi ini dapat ditemukan pada gambar 3.1.1.

Untuk melihat perintah selanjutnya, perhatikan gambar-gambar 3.1.2, 3.1.3, dan 3.1.4, di mana akan dijalankan perintah "less tinggiBadanKelas.txt". Berbeda dengan perintah "more", perintah ini akan menampilkan isi dari file "tinggiBadanKelas.txt" dalam sebuah jendela baru. Selain itu, perintah "less" juga dapat menerima input tambahan seperti "/60", yang dapat ditemukan pada gambar 3.1.3. Setelah input "/60" diberikan, terminal akan menampilkan data yang mengandung kata "60" dalam file txt tersebut.

Soal 2 Buatlah sebuah file yang berisikan daftar nama daerah dengan menggunakan file descriptor dan redirection dengan mengimplementasikan penggunaan standar input, standar output dan standar error.

| 3.2.1. | 3.2.2 |
| --- | --- |
| 3.2.3 | |

Penjelasan :

Dalam langkah ini, langkah pertama yang akan dijalankan adalah perintah "ls game\_popular.txt 2> error.txt", yang terlihat dalam gambar 2.3.1. Karena file game\_popular.txt belum ada, akan muncul kesalahan "no such file or directory" yang akan disimpan di file "error.txt". Selanjutnya, perintah "cat 0< error.txt" akan digunakan untuk membaca isi dari error.txt yang berisi pesan kesalahan dari perintah sebelumnya.

Dalam gambar 3.2.2, terlihat bahwa akan dijalankan perintah "cat 1> game\_popular.txt" untuk membuat dan mengisi file teks "game\_popular". Isi dari file teks ini berisi daftar permainan video yang sering dimainkan oleh para pemain saat ini. Setelah berhasil menjalankan perintah tersebut, gambar 3.2.3 menunjukkan bahwa perintah "cat 0< game\_popular.txt" akan digunakan untuk membaca isi dari file teks "game\_popular".

Soal 3 Tampilkan isi file tersebut dengan menggunakan implementasi dari pipeline dan filter yaitu perintah.

* Grep
* Wc
* Uniq
* Sort
* Head
* Tail
* Cut
* Paste
* Awk
* Sed

| 3.3.1 | 3.3.2 |
| --- | --- |
| 3.3.3 | 3.3.4 |
| 3.3.5 | 3.3.6 |
| 3.3.7 | 3.3.8 |
| 3.3.9 | 3.3.10 |

Penjelasan:

Pada gambar 3.3.1, akan dijalankan perintah "cat game\_popular.txt | grep of" yang akan menampilkan baris-baris yang mengandung kata "of". Setelah menjalankan perintah tersebut, dapat dilihat bahwa ada 6 permainan video yang memiliki kata "of" pada nama gamenya.

Pada gambar 3.3.2, perlu dilakukan perintah "cat game\_popular.txt | wc" untuk menampilkan informasi seperti jumlah kata, jumlah baris, dan jumlah karakter yang terdapat dalam file teks tersebut.

Pada gambar 3.3.3, perintah yang diperlukan untuk menyelesaikan langkah ini adalah "cat game\_popular.txt | uniq" yang akan menampilkan isi dari file teks tanpa menampilkan baris duplikat atau baris yang sama. Namun, karena tidak ada permainan video yang memiliki nama yang sama, hasil output dari perintah ini akan sama dengan isi file teks awalnya.

Pada gambar 3.3.4, perintah yang akan dijalankan adalah perintah “cat game\_popular.txt | sort” untuk mengurut isi dari barisan-barisan (secara abjad dari a-z) yang terdapat pada file txt “game\_popular”.

Pada gambar 3.3.5, diperlukan perintah "cat game\_popular.txt | head -5" untuk menampilkan lima baris teratas dari file teks "game\_popular". Setelah menjalankan perintah tersebut, dapat dilihat bahwa lima baris teratas dari isi file teks "game\_popular" ditampilkan di terminal.

Pada gambar 3.3.6, perintah yang akan dijalankan adalah "cat game\_popular.txt | tail -3" yang akan menampilkan tiga baris terakhir dari isi file teks "game\_popular". Setelah menjalankan perintah ini, perhatikan bahwa hanya tiga baris terakhir yang ditampilkan di terminal.

Pada gambar 3.3.7, perintah yang akan dijalankan adalah "cat game\_popular.txt | cut -b 1-4" untuk menampilkan empat karakter pertama dari setiap baris dalam file teks "game\_popular.txt".

Pada gambar 3.3.8, perintah yang diperlukan adalah "paste -d "," tinggiBadanKelas.txt game\_popular.txt" untuk menggabungkan baris dari file teks "tinggiBadanKelas" dan file teks "game\_popular". Setelah menjalankan perintah ini, baris dari kedua file teks akan digabungkan seperti yang terlihat pada gambar 3.3.8.

Pada gambar 3.3.9, hal yang sama terjadi seperti perintah pada gambar 3.3.8, di mana perintah "cat game\_popular.txt | awk '{print NR,$0}'" akan dijalankan untuk menambahkan nomor baris pada awal setiap baris dalam file teks "game\_popular". Hasil dari perintah ini hampir sama dengan perintah pada gambar 3.3.8 dan dapat dilihat pada gambar 3.3.9.

Pada gambar 3.3.10, diperlukan perintah "cat nama\_daerah.txt | sed 's/Of/The/ '" untuk mengganti kata "Of" menjadi "The" dalam setiap baris. Hasil dari perintah ini dapat dilihat pada gambar 3.3.10.

# **Kesimpulan dan Saran**

Pada praktikum modul 5, dapat dipelajari bagaimana pengguna dapat melakukan proses Input/Output melalui terminal dengan bantuan perintah seperti “cat”, “cat 0<”, dan “cat 1>”. Selain itu, File descriptor juga dapat dipelajari pada modul ini untuk mempelajari perintah pembelokkan dan filtering. Kedua hal ini dapat membantu pengguna dalam menampilkan isi-isi tertentu dari suatu file yang ada pada dalam sistem operasi.

Saran yang dapat diberikan adalah melakukan perintah-perintah yang dipelajari pada modul ini namun file yang akan digunakan adalah file dengan tipe file selain “txt”.