**LAPORAN PRAKTIKUM**

**SISTEM OPERASI RB**

**MODUL 5**

**Oleh :**

**Indah Mutiara (12140158)**



**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 11**](#_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:

* Files
* Keyboard (input)
* Layar (output)
* 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

Melakukan percobaan Output ke layar (standar output) dengan input dari sistem (kernel).

1. Soal 2

Melakukan Output ke layar (standar output), input dari keyboard (standar input).

1. Soal 3

Melakukan Input nama direktori, output tidak ada (membuat direktori baru), bila terkadi error maka tampilan error pada layar (standard error).

Percobaan 2

1. Soal 4

Melakukan Pembelokkan standar output.

1. Soal 5

Melakukan Pembelokkan standar input.

1. Soal 6

Melakukan Pembelokkan standar error untuk disimpan pada file.

1. Soal 7

Melakukan Pembelokkan standar output ke dalam file.

1. Soal 8

Melakukan Penambahan output ke file yang sudah ada.

1. Soal 9

Menampilkan file 1 dan menampilkan input keyboard seta menampilkan file 2.

1. Soal 10

Melakukan output ke layar dari input keyboard dengan menggunakan pembatas.

Percobaan 3

1. Soal 11

Melakukan Pipeline untuk membuat eksekusi proses dengan melewati data langsung ke data lainnya.

1. Soal 12

Melakukan Filtering dengan Pipeline untuk mengkombinasikan utilitas sistem untuk membuat fungsi kompleks.

1. Soal 13

Buatlah sebuah file txt yang berisikan daftar mata kuliah dengan menggunakan pembelokkan.

1. Soal 14

Tampilkan daftar tersebut dengan menerapkan pembelokkan.

1. Soal 15

Silahkan Kumpulkan Latihan Percobaan yang telah dilakukan Sesuai dengan Arahan Asisten Praktikum.

# **Hasil dan Jawaban**

Soal 1 Melakukan percobaan Output ke layar (standar output) dengan input dari sistem (kernel).

|  |
| --- |

Penjelasan :

Untuk menyelesaikan langkah awal, tindakan yang akan dilakukan adalah menjalankan perintah "ps" dengan tujuan untuk memeriksa daftar proses yang sedang aktif di sistem operasi. Setelah menjalankan perintah tersebut, kita dapat mengamati adanya proses "bash" dan "ps" yang sedang berjalan.

Soal 2 Melakukan Output ke layar (standar output), input dari keyboard (standar input).

|  |
| --- |

Penjelasan :

Tahap kedua akan diselesaikan menggunakan perintah "cat". Perintah ini akan menerima input dari keyboard dan menampilkan outputnya di terminal, seperti yang terlihat pada gambar di atas.

Soal 3 Melakukan Input nama direktori, output tidak ada (membuat direktori baru), bila terkadi error maka tampilan error pada layar (standard error).

|  |
| --- |

Penjelasan:

Untuk menyelesaikan latihan nomor 3, akan digunakan perintah "mkdir MyData" untuk memeriksa kemungkinan adanya kesalahan (error) pada sistem. Setelah menjalankan perintah tersebut, memang benar bahwa direktori/file dengan nama MyData tidak dapat ditemukan dalam direktori kerja (working directory) sistem.

Soal 4 Melakukan Pembelokkan standar output.

|  |
| --- |

Penjelasan :

Pada langkah berikutnya, perintah yang akan digunakan untuk menyelesaikan latihan ini adalah perintah "cat 1> MyFile.txt" yang bertujuan untuk mengisi teks ke dalam file MyFile.txt. Setelah selesai mengisi teks sesuai instruksi latihan, gunakan perintah "control z" untuk menyimpan teks tersebut.

Soal 5 Melakukan Pembelokkan standar input.

|  |
| --- |

Penjelasan:

Pada tahap ini, akan digunakan perintah "cat 0< MyFile.txt" untuk membaca konten dari file MyFile.txt.

Soal 6 Melakukan Pembelokkan standar error untuk disimpan pada file.

|  |
| --- |

Penjelasan :

Kemudian, untuk menyelesaikan latihan ini, akan digunakan perintah pembelokkan standar error "mkdir MyData 2> Error\_MyData.txt". Perintah ini akan menyimpan pesan error yang terjadi saat menjalankan perintah "mkdir MyData" ke dalam file Error\_MyData.txt.

Soal 7 Melakukan Pembelokkan standar output ke dalam file.

|  |
| --- |

Penjelasan :

Untuk menyelesaikan tahap ini, dibutuhkan dua perintah. Perintah pertama adalah "echo NIM:121140158 > Identitas.txt" yang akan digunakan untuk menyimpan identitas NIM ke dalam file Identitas.txt. Perintah kedua adalah "cat 0< Identitas.txt" yang akan dijalankan untuk membaca isi dari file Identitas.txt.

Soal 8 Melakukan Penambahan output ke file yang sudah ada.

|  |
| --- |

Penjelasan :

Untuk menyelesaikan tahap ini, diperlukan perintah "echo Nama: Indah Mutiara >> Identitas.txt" untuk menyimpan identitas nama praktikan ke dalam file Identitas.txt. Setelah berhasil menjalankan perintah tersebut, perintah "cat 0< Identitas.txt" dapat digunakan untuk melihat isi perubahan pada file Identitas.txt.

Soal 9 Menampilkan file 1 dan menampilkan input keyboard seta menampilkan file 2.

|  |
| --- |

Penjelasan :

Pada langkah ini, perintah "cat 0< MyFile2.txt" akan menghasilkan output kesalahan standar yang menunjukkan bahwa file tersebut tidak dapat ditemukan dalam direktori kerja sistem. Setelah itu, perintah "cat 1> MyFile2.txt" akan dieksekusi untuk membuat file MyFile2.txt beserta isi teksnya. Setelah berhasil membuat file MyFile2.txt, perintah "cat MyFile.txt - MyFile2.txt" akan dijalankan untuk menggabungkan dan menampilkan isi kedua file tersebut.

Soal 10 Melakukan output ke layar dari input keyboard dengan menggunakan pembatas.

|  |
| --- |

Penjelasan :

Akan digunakan perintah "cat" untuk melakukan input dengan menggunakan pembatas (seperti yang terlihat pada gambar di atas).

Soal 11 Melakukan Pipeline untuk membuat eksekusi proses dengan melewati data langsung ke data lainnya.

|  |  |
| --- | --- |

Penjelasan :

Pada tahap ini, akan digunakan perintah pipeline "ls -la" untuk menampilkan semua file yang ada dalam direktori kerja sistem. Selanjutnya, perintah "ls -la | grep txt" akan dieksekusi untuk memfilter dan menampilkan hanya file-file dengan ekstensi txt.

Soal 12 Melakukan Filtering dengan Pipeline untuk mengkombinasikan utilitas sistem untuk membuat fungsi kompleks.

|  |
| --- |

Penjelasan :

Untuk menyelesaikan tahap ini, akan digunakan perintah "cat 1> Animal.txt" untuk membuat file txt baru dengan isi sesuai yang ditampilkan dalam gambar di atas. Setelah itu, akan dilakukan filtering menggunakan pipeline dengan perintah "cat Animal.txt | sort" untuk mengurutkan isi file Animal.txt. Selanjutnya, akan dijalankan perintah "cat Animal.txt | grep dog | grep -v cat" untuk menampilkan baris-baris yang mengandung kata "dog" dan tidak mengandung kata "cat".

Soal 13 Buatlah sebuah file txt yang berisikan daftar mata kuliah dengan menggunakan pembelokkan.

|  |
| --- |

Penjelasan :

Pada tahap ini, akan dilakukan eksekusi perintah "cat 1> MataKuliah.txt" untuk membuat sebuah file txt yang berisi daftar mata kuliah yang diambil oleh praktikan pada semester ini. Setelah berhasil menciptakan file txt tersebut dengan perintah pertama, akan dijalankan perintah "cat 0< MataKuliah.txt" untuk melihat isi dari file MataKuliah.txt.

Soal 14 Tampilkan daftar tersebut dengan menerapkan pembelokkan.

|  |
| --- |

Penjelasan :

Pada tahap ini, akan dijalankan perintah "cat MataKuliah.txt | sort" untuk mengurutkan isi file txt MataKuliah. Setelah berhasil menjalankan perintah pertama, perintah pembelokkan berikutnya yang digunakan adalah "cat MataKuliah.txt | grep PBO" untuk menampilkan baris yang mencakup kata "PBO" dalam file MataKuliah.txt. Perintah terakhir yang dijalankan adalah perintah pembelokkan "cat MataKuliah.txt | grep -v dan | grep Sistem" untuk menampilkan baris yang tidak mengandung kata "dan" dan mencakup kata "Sistem".

# **Kesimpulan dan Saran**

Dalam praktikum modul 5, akan dipelajari bagaimana pengguna dapat melakukan proses Input/Output melalui terminal menggunakan perintah-perintah seperti "cat", "cat 0<", dan "cat 1>". Selain itu, modul ini juga akan memperkenalkan konsep File descriptor yang melibatkan perintah pembelokkan dan filtering. Kedua konsep ini akan membantu pengguna dalam menampilkan isi tertentu dari file yang ada di sistem operasi.

Sebagai saran, disarankan untuk mencoba perintah-perintah yang dipelajari dalam modul ini dengan menggunakan file berjenis selain "txt".