

MODUL PRAKTIKUM SISTEM OPERASI LINUX

MINGGU 14 – DASAR SHELL PROGRAMMING

A. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa mampu:

1. Menjelaskan konsep shell, jenis-jenis shell, dan peran shell dalam sistem operasi Linux.
2. Membedakan shell interaktif, shell script, dan subshell.
3. Menulis dan menjalankan shell script sederhana menggunakan Bash.
4. Menggunakan variabel (user-defined dan environment) secara benar.
5. Menggunakan input/output, redirection, dan pipeline dalam script.
6. Menggunakan operator aritmatika, relasional, dan logika.
7. Mengimplementasikan percabangan dasar (if, elif, else, case) dan perulangan dasar (for, while, until).
8. Memahami konsep dasar exit status dan pengaruhnya terhadap alur script.

B. DASAR TEORI SINGKAT

1. Peran Shell dalam Sistem Operasi

Shell adalah antarmuka berbasis teks yang menghubungkan user dengan kernel. Shell menerima perintah dari user, menerjemahkannya menjadi system call, lalu menampilkan hasilnya.

Fungsi utama shell:

- Command-line interface (CLI)
- Scripting environment untuk otomatisasi

Perintah Linux yang telah digunakan pada modul-modul sebelumnya (ps, ls, df, top, chmod, useradd) dijalankan melalui shell.

2. Jenis-Jenis Shell di Linux

Contoh shell:

- Bourne Shell (sh)
- Bash (Bourne Again Shell)
- C Shell (csh)

- Korn Shell (ksh)
- TENEX C Shell (tcsh)

Bash adalah shell default di banyak distribusi Linux karena kaya fitur dan dokumentasi.

3. Mode Kerja Bash: Interaktif vs Script

- Mode interaktif: user mengetik perintah satu per satu di terminal.
- Mode script: perintah ditulis dalam file .sh dan dijalankan sekaligus.

Contoh penggunaan script:

- Backup otomatis
- Pembersihan log
- Pemrosesan data dan laporan

4. Shebang dan Environment

Baris pertama script:

```
#!/bin/bash
```

disebut shebang dan memberi tahu kernel bahwa file akan dijalankan oleh interpreter /bin/bash.

Environment adalah kumpulan variabel global (PATH, HOME, USER, SHELL, LANG) yang memengaruhi perilaku program dan script.

5. Variabel di Bash

Karakteristik variabel:

- Tidak perlu deklarasi tipe.
- Nilai disimpan sebagai string; interpretasi angka dilakukan dalam konteks aritmatika.

Jenis variabel:

1. User-defined variables

Contoh:

```
nama="Budi"
```

2. Environment variables

Contoh: PATH, HOME, USER, SHELL.

3. Special variables

- \$0 : nama script
- \$1, \$2, ... : argumen script

- \$# : jumlah argumen
- \$@ : semua argumen
- \$\$: PID script
- \$? : exit status perintah terakhir

6. Operator dan Ekspresi

Kategori operator:

- Aritmatika: +, -, *, /, %
- Relasional angka: -eq, -ne, -gt, -lt, -ge, -le
- Relasional string: =, !=, -z, -n
- File: -f, -d, -e, -r, -w, -x
- Logika: &&, ||, !

Penulisan ekspresi:

- test EXPRESSION
- [EXPRESSION]
- [[EXPRESSION]] (lebih modern dan fleksibel)

7. Struktur Kontrol Dasar

Bash mendukung:

- Percabangan: if, elif, else, case
- Perulangan: for, while, until
- Perintah pendukung: break, continue

Struktur kontrol memungkinkan script mengambil keputusan dan mengulang operasi sesuai kondisi.

8. Exit Status

Setiap perintah mengembalikan exit status 0–255:

- 0 : sukses
- non-0 : error

Contoh:

```
ls /tmp
echo $?
```

```
ls /folder_tidak_ada  
echo $?
```

Exit status digunakan untuk menentukan apakah script melanjutkan atau berhenti.

C. SOFTWARE YANG DIBUTUHKAN

1. Sistem operasi Linux (Debian/Ubuntu).
2. Bash Shell.
3. Editor teks: nano, vim, atau gedit.
4. Terminal.

D. DAFTAR PERINTAH YANG DIGUNAKAN

echo, read, printf, bash, chmod, export, env, test, [], [[]], \$((...)).

E. PRAKTIKUM 1 – MEMBUAT SCRIPT PERTAMA

Langkah 1 – Cek Versi Bash

```
bash --version
```

Langkah 2 – Script helloworld.sh

```
01 #!/bin/bash  
02 echo "Halo Dunia!"  
03 echo "Praktikum Shell Programming"
```

Langkah 3 – Jadikan Executable

```
chmod +x helloworld.sh
```

Langkah 4 – Menjalankan Script

```
./helloworld.sh
```

atau:

```
bash helloworld.sh
```

F. PRAKTIKUM 2 – VARIABEL DAN OPERASI DASAR

Script variabel.sh

```
01 #!/bin/bash
02 nama="Mahasiswa"
03 nim="2024xxxx"
04 ipk=3.75
05
06 echo "Nama: $nama"
07 echo "NIM : $nim"
08 echo "IPK : $ipk"
```

Script aritmatika.sh

```
01 #!/bin/bash
02 a=20
03 b=5
04
05 echo "a = $a, b = $b"
06 echo "Tambah : $((a + b))"
07 echo "Kurang : $((a - b))"
08 echo "Kali    : $((a * b))"
09 echo "Bagi    : $((a / b))"
10 echo "Sisa    : $((a % b))"
```

G. PRAKTIKUM 3 – INPUT DAN OUTPUT

Script input.sh

```
01 #!/bin/bash
02 read -p "Masukkan nama Anda: " nama
03 read -p "Masukkan NIM Anda : " nim
04 echo "Halo $nama, NIM: $nim"
```

Script inputpass.sh

```

01 #!/bin/bash
02 read -p "Username: " user
03 read -sp "Password: " pass
04 echo ""
05 echo "Login sebagai: $user"

```

Script format.sh

```

01 #!/bin/bash
02 printf "%-15s %-10s %-5s\n" "Nama" "NIM" "Nilai"
03 printf "%-15s %-10s %-5d\n" "Budi" "2024001" 90
04 printf "%-15s %-10s %-5d\n" "Siti" "2024002" 85

```

H. PRAKTIKUM 4 – PERCABANGAN

Script if1.sh – If Sederhana

```

01 #!/bin/bash
02 read -p "Masukkan nilai: " nilai
03
04 if [ $nilai -ge 60 ]; then
05     echo "LULUS"
06 fi

```

Script if2.sh – If-Else

```

01 #!/bin/bash
02 read -p "Masukkan nilai: " nilai
03
04 if [ $nilai -ge 60 ]; then
05     echo "LULUS"
06 else
07     echo "TIDAK LULUS"
08 fi

```

Script grade.sh – Penentuan Grade

```

01 #!/bin/bash
02 read -p "Masukkan nilai (0-100): " nilai
03
04 if [ $nilai -ge 85 ]; then
05     echo "Grade: A"
06 elif [ $nilai -ge 75 ]; then
07     echo "Grade: B"
08 elif [ $nilai -ge 65 ]; then
09     echo "Grade: C"
10 elif [ $nilai -ge 55 ]; then
11     echo "Grade: D"
12 else
13     echo "Grade: E (Tidak Lulus)"
14 fi

```

Script cekfile.sh – Pemeriksaan File

```

01 #!/bin/bash
02 read -p "Masukkan nama file: " file
03
04 if [ -f "$file" ]; then
05     echo "File ada"
06 elif [ -d "$file" ]; then
07     echo "$file adalah direktori"
08 else
09     echo "File atau direktori tidak ditemukan"
10 fi

```

I. PRAKTIKUM 5 – PERULANGAN

Script for1.sh

```

01 #!/bin/bash

```

```
02 for hari in Senin Selasa Rabu Kamis Jumat
03 do
04     echo "Hari: $hari"
05 done
```

Script while1.sh

```
01 #!/bin/bash
02 i=1
03 while [ $i -le 5 ]
04 do
05     echo "Nilai i = $i"
06     i=$((i+1))
07 done
```

Script until1.sh

```
01 #!/bin/bash
02 i=1
03 until [ $i -gt 5 ]
04 do
05     echo "Until i = $i"
06     i=$((i+1))
07 done
```

Script breakcon.sh

```
01 #!/bin/bash
02 for i in {1..10}
03 do
04     if [ $i -eq 3 ]; then
05         echo "Melewati angka 3"
06         continue
07     fi
08
```

```
09      if [ $i -eq 7 ]; then
10          echo "Berhenti di angka 7"
11          break
12      fi
13
14      echo "i = $i"
15 done
```

J. BEST PRACTICES PENULISAN BASH SCRIPT (LEVEL DASAR)

1. Selalu tulis shebang `#!/bin/bash` pada baris pertama.
2. Gunakan nama file dan variabel yang jelas dan konsisten.
3. Tambahkan komentar seperlunya untuk blok logika penting.
4. Hindari hardcoding path penting; gunakan variabel konfigurasi di awal script.
5. Periksa keberadaan file/direktori dengan `-f`, `-d`, `-e` sebelum digunakan.
6. Selalu gunakan kutip ganda `"..."` untuk variabel, terutama yang mengandung spasi.
7. Baca kembali script sebelum dijalankan; pastikan tanda kurung dan kutip seimbang.

K. ANALISIS HASIL PERCOBAAN

1. Apa perbedaan shell interaktif dan shell script?
2. Mengapa shebang diperlukan?
3. Mengapa tidak boleh ada spasi di sekitar tanda `=` pada variabel?
4. Apa fungsi environment variable?
5. Mengapa exit status penting dalam alur script?

L. TUGAS MANDIRI

1. Buat script yang menerima dua angka sebagai argumen dan menampilkan hasil operasi aritmatika (tambah, kurang, kali, bagi, sisa).
2. Buat script yang menghitung rata-rata tiga nilai dan menentukan grade.
3. Buat script yang menampilkan tabel perkalian 1–10.

4. Buat script yang memeriksa apakah suatu bilangan genap atau ganjil.
5. Buat script yang menampilkan informasi sistem: username, hostname, tanggal, direktori aktif, versi kernel.

M. REFERENSI

1. W3Schools – Bash Tutorial (intro, variables, conditions, loops, functions).
2. W3Schools.io – Bash Shell Scripting Tutorials and Examples.
3. Advanced Bash-Scripting Guide – The Linux Documentation Project.
4. Modul Sistem Operasi sebelumnya: System Call, Process, File System, User Management.