

**LAPORAN PRAKTIKUM
SISTEM OPERASI RD
MODUL 7**

Oleh :

Muhammad Yusuf (122140193)



Program Studi Teknik Informatika

Institut Teknologi Sumatera

2024

Daftar Isi

Daftar Isi	2
1. Dasar Teori.....	3
2. Ulasan dan Hasil Jawaban.....	4
3. Kesimpulan dan Saran.....	9

1. Dasar Teori

Shell adalah program yang bertindak sebagai penerjemah perintah antara pengguna dan sistem operasi, khususnya kernel. Ini memberikan prompt sebagai antarmuka pengguna di mana pengguna dapat memasukkan perintah internal atau eksternal untuk dieksekusi. Pemrograman Shell melibatkan pengelompokan perintah untuk melakukan tugas tertentu. Kelebihan pemrograman Shell di Linux adalah kemampuannya untuk melakukan proses I/O, mengelola kondisi, pengulangan, dan membuat fungsi layaknya bahasa pemrograman. Dasar-dasar pemrograman Shell mencakup konsep variabel, eksekusi Shell, input pengguna, operator perbandingan, logika kondisi, pengulangan, dan penggunaan array.

Variabel dalam Shell digunakan untuk menyimpan data sementara dalam memori komputer. Shell Execution adalah pelaksanaan perintah Linux di dalam skrip. Input pengguna memungkinkan pengguna untuk memberikan input dalam skrip. Operator perbandingan digunakan untuk membandingkan nilai atau variabel. Logika kondisi, seperti if-then-else, digunakan untuk mengatur tindakan yang diambil berdasarkan kondisi yang diberikan. Pengulangan dalam Shell bisa dilakukan menggunakan for, while, atau until. Array dalam Shell didefinisikan menggunakan tanda kurung dan dapat digunakan untuk menyimpan sejumlah nilai dalam satu variabel.

2. Ulasan dan Hasil Jawaban

TUGAS

1. Buatlah sebuah program yang dapat menerima inputan sebanyak N kemudian tentukan :
 - a. Nilai Maksimum
 - b. Nilai Minimum
2. Buatlah sebuah program kalkulator sederhana yang dapat melakukan penjumlahan, pengurangan, perkalian dan pembagian!
3. Buatlah sebuah program yang dapat menerima inputan sebanyak N. Kemudian tentukan jumlah bilangan ganjil dan genap!

Jawaban No. 1

```
Vbox@yusuf:~$ cat minmax.sh
echo "Masukkan jumlah N:"
read N

max=0
min=0

for ((i=1; i<=$N; i++))
do
    echo "Masukkan angka ke-$i:"
    read angka

    if [ $i -eq 1 ]; then
        max=$angka
        min=$angka
    else
        if [ $angka -gt $max ]; then
            max=$angka
        fi

        if [ $angka -lt $min ]; then
            min=$angka
        fi
    fi
done

echo "Nilai Maksimum: $max"
echo "Nilai Minimum: $min"
```

```
Vbox@yusuf:~$ nano minmax.sh
Vbox@yusuf:~$ chmod +x minmax.sh
Vbox@yusuf:~$ ./minmax.sh
Masukkan jumlah N:
5
Masukkan angka ke-1:
10
Masukkan angka ke-2:
90
Masukkan angka ke-3:
20
Masukkan angka ke-4:
100
Masukkan angka ke-5:
1
Nilai Maksimum: 100
Nilai Minimum: 1
```

Jawaban No. 2

```
Vbox@yusuf:~$ cat kalku.sh
echo "simple calculator"
sum=0
i="y"
echo "enter first number"
read n1
echo "enter second number"
read n2
while [ $i = "y" ]
do
echo "1.Addition"
echo "2.Subtraction"
echo "3.Multiplication"
echo "4.Division"
echo "Enter choice"
read ch
case $ch in
1)sum=$(echo " $n1 + $n2" | bc -l)
echo "Addition is =" $sum;;
2)sum=$(echo "$n1 - $n2" | bc -l)
echo "Sub is =" $sum;;
3)sum=$(echo "$n1 * $n2" | bc -l)
echo "Mul is =" $sum;;
4)sum=$(echo "$n1 / $n2" | bc -l)
echo "div is =" $sum;;
*)echo "invalid choice"
esac
echo "Do you want to continue"
read i
if [ $i != "y" ]
then
exit
fi
done
```

Referensi From Internet

```
Vbox@yusuf:~$ nano kalku.sh
Vbox@yusuf:~$ chmod +x kalku.sh
Vbox@yusuf:~$ ./kalku.sh
```

```
simple calculator
enter first number
10
enter second number
20
1.Addition
2.Subtraction
3.Multiplication
4.Division
Enter choice
1
Addition is = 30
Do you want to continue
y
1.Addition
2.Subtraction
3.Multiplication
4.Division
Enter choice
2
Sub is = -10
```

```
Do you want to continue
y
1.Addition
2.Subtraction
3.Multiplication
4.Division
Enter choice
3
Mul is = 200
Do you want to continue
y
1.Addition
2.Subtraction
3.Multiplication
4.Division
Enter choice
4
div is = .50000000000000000000
Do you want to continue
n
```

Jawaban No. 3

```
Vbox@yusuf:~$ cat ganjilgenap.sh
echo "Masukkan jumlah N:"
read N

jumlah_ganjil=0
jumlah_genap=0

for ((i=1; i<=$N; i++))
do
    echo "Masukkan angka ke-$i:"
    read angka

    if [ $((angka % 2)) -eq 0 ]; then
        ((jumlah_genap++))
    else
        ((jumlah_ganjil++))
    fi
done

echo "Jumlah Bilangan Ganjil: $jumlah_ganjil"
echo "Jumlah Bilangan Genap: $jumlah_genap"
```

```
Vbox@yusuf:~$ nano ganjilgenap.sh
Vbox@yusuf:~$ chmod +x ganjilgenap.sh
Vbox@yusuf:~$ ./ganjilgenap.sh
Masukkan jumlah N:
4
Masukkan angka ke-1:
10
Masukkan angka ke-2:
8
Masukkan angka ke-3:
9
Masukkan angka ke-4:
1
Jumlah Bilangan Ganjil: 2
Jumlah Bilangan Genap: 2
```


3. Kesimpulan dan Saran

Setelah melakukan praktikum sistem operasi dan menulis kode Shell, dapat disimpulkan bahwa Shell merupakan antarmuka yang memungkinkan interaksi antara pengguna dan sistem operasi melalui perintah-perintah. Dalam pemrograman Shell, kita dapat mengelompokkan perintah-perintah untuk menjalankan tugas-tugas tertentu, seperti pengolahan data, pengambilan input dari pengguna, dan pengambilan keputusan berdasarkan kondisi tertentu. Variabel digunakan untuk menyimpan data sementara, sementara eksekusi Shell memungkinkan perintah Linux dieksekusi dalam skrip. Operator perbandingan dan logika kondisi digunakan untuk pengambilan keputusan, sedangkan pengulangan memungkinkan eksekusi berulang untuk serangkaian perintah. Penggunaan array juga berguna untuk menyimpan sejumlah nilai dalam satu variabel.

Melalui praktikum ini, pemahaman tentang pemrograman Shell menjadi lebih baik, termasuk konsep-konsep dasarnya seperti variabel, eksekusi perintah, input pengguna, operator perbandingan, logika kondisi, pengulangan, dan penggunaan array. Kemampuan ini sangat berguna dalam pengembangan skrip-skrip otomatisasi, pengolahan data, dan pengaturan tugas-tugas sistem yang kompleks dalam lingkungan Linux atau Unix.