

MODUL 3
SISTEM OPERASI (SO)
“PERULANGAN DAN SELEKSI”



TGL PRAKTIKUM	: 8 November 2012
NAMA	: ACHMAD FAJAR NORULLAH
NRP	: 11.04.111.00075
KELAS	: C
DOSEN PENGAMPU	: FAIKUL UMAM, S.Kom.

Disetujui :
...../...../...../Bangkalan

(TONI WIJAYA)
09.04.111.000.64



LABORATORIUM SISTEM INFORMASI

LABORATORIUM SISTEM INFORMASI
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS TRUNOJOYO

2011

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada modul ini terdapat pengulangan yang mana pengulangan merupakan sebuah perintah yang dapat mengulang suatu aksi sampai memenuhi kondisi yang terdapat di pengulangan tersebut. Selain itu terdapat juga pengkondisian dimana didalam suatu permasalahan terdapat dua atau lebih kondisi dan jika memenuhi kondisi misal kondisi A maka aksi pada kondisi A lah yang akan dilakukan.

1.2 Tujuan

Tujuan mahasiswa mengikuti pratikum ini adalah :

- a. Praktikan bisa mengerti, memahami dan menerapkan tentang fungsi pengulangan dan seleksi kondisi pada Linux.
- b. Pratikan mengetahui dan bisa memahami semua tata cara penulisan dan pendeskripsian pengulangan dan seleksi kondisi pada sistem operasi Linux.

BAB II

DASAR TEORI

Perulangan dan Seleksi kondisi ada dalam setiap pemrograman tak terkecuali dalam shell Linux. Perulangan sendiri merupakan pengulangan aksi sampai memenuhi kondisi dari aksi tersebut sedangkan penyeleksian kondisi adalah keadaan dimana suatu program akan mengeksekusi aksi yang terdapat pada kondisi yang terpenuhi oleh masalah.

2.1 Perulangan

1. Perulangan For

Perulangan for digunakan untuk melakukan pekerjaan berulang sebanyak daftar yang disediakan, contoh :

```
#!/bin/sh
# Nama file skrip: for1.sh for i in 1 2 3 4 5
do
```

```
    echo "Looping ... number $i" done
```

dari skrip diatas jika dijalankan akan keluar output berupa :

```
Looping ..... number 1
Looping ..... number 2
Looping ..... number 3
Looping ..... number 4
Looping ..... number 5
```

2. Perulangan While

Perulangan while dapat digunakan untuk melakukan pekerjaan berulang yang jumlah perulangannya tidak pasti tetapi bergantung pada suatu kondisi yang harus dipenuhi. Perhatikan skrip berikut :

```
#!/bin/sh
# Nama file skrip: while.sh
INPUT_STRING=hello
while [ "$INPUT_STRING" != "bye" ]
do
    echo "Ketikkan sesuatu (bye untuk keluar)"
    read INPUT_STRING
    echo "Anda mengetikkan: $INPUT_STRING"
done
```

2.2 Seleksi Kondisi

1. Test

Test digunakan secara virtual oleh setiap skrip shell yang ditulis. Test memang tidak sering dipanggil secara langsung. Test sering dipanggil sebagai [. [adalah link simbolik ke perintah test, membuat program shell lebih nyaman dibaca. Coba tulis perintah di bawah ini dan perhatikan luarannya:

```
$ type [  
[ is a shell builtin  
$ which [  
/usr/bin/[  
$ ls -l /usr/bin/[  
lrwxrwxrwx 1 root root 4 Mar 27 2000 /usr/bin/[ -> test
```

Test sering dipanggil secara tak langsung melalui pernyataan if dan while. Sintaks untuk if...then...else... adalah

```
if [ ... ]  
then  
    # if-code  
else  
    # else-code  
fi
```

2. Case

Pernyataan case menghemat pernyataan if .. then .. else banyak baris.

Sintaksnya sangat sederhana:

```
#!/bin/sh  
# Nama file skrip: talk.sh  
echo "Please talk to me ..." while :  
do  
    read INPUT_STRING  
    case $INPUT_STRING in hello)  
        echo "Hello yourself!" ;;  
        bye)  
            echo "See you again!" break  
        ;;  
        *)  
            echo "Sorry, I don't understand" ;;  
    esac  
done  
echo  
echo "That's all folks!"  
Coba jalankan kode di atas dan perhatikan bagaimana kerjanya...  
$ ./talk.sh  
Please talk to me ... hello  
Hello yourself!  
What do you think of politics? Sorry, I don't understand bye  
See you again!
```

That's all folks!

Baris case sendiri selalu berformat sama. Dimulai dengan memeriksa nilai suatu variabel kondisi, pada contoh di atas adalah variabel `INPUT_STRING`. Pilihan-pilihan kemudian didaftarkan dan diikuti oleh suatu kurung tutup seperti `hello)` dan `bye)`. Ini berarti bahwa jika `INPUT_STRING` cocok dengan `hello` maka bagian kode itu dieksekusi, sampai dengan dua titik-koma. Jika `INPUT_STRING` cocok dengan `bye` maka pesan “goodbye” dicetak dan keluar dari perulangan. Jika anda ingin keluar dari skrip maka dapat menggunakan perintah `exit`, bukan `break`. Pilihan ketiga, di sini `*`), merupakan kondisi *catch-all* default; tidak harus tetapi sering berguna untuk tujuan debugging bahkan jika kita mengetahui nilai apa yang akan dimiliki variabel dalam test.

BAB III

TUGAS PENDAHULUAN

Pertanyaan :

1. Apa manfaat dari karakter escape? Bagaimana menampilkan karakter ^, % dan ~?
2. Anda sudah mempelajari perulangan dan seleksi kondisi pada bahasa. Apa kegunaan keduanya? Buat sebuah program dalam bahasa C yang melibatkan kedua fitur ini, misalnya program tebak angka!
3. Apa yang dikeluarkan oleh perintah berikut:

```
$ ls -ld {usr,usr/local}/{bin,sbin,lib}
```

Jawaban :

1. karakter escape berfungsi untuk menyatakan atau memunculkan suatu karakter khusus, untuk menampilkan tanda ^ dan % cukup ditulis saja sedangkan untuk menampilkan karakter ~ dengan mengetik \~ karena jika mengetik ~ akan menampilkan folder user.
2. fungsi perulangan digunakan untuk melakukan perulangan atau looping sampai syarat yang ditentukan dipenuhi atau mencapai kondisi yang benar benar diinginkan pada syarat, sedangkan percabangan adalah suatu keadaan dimana terdapat pengkondisian dan jika kondisi tersebut dipenuhi maka akan dilakukan aksi yang terdapat pada kondisi tersebut sedangkan jika tidak maka aksi tidak akan berjalan.

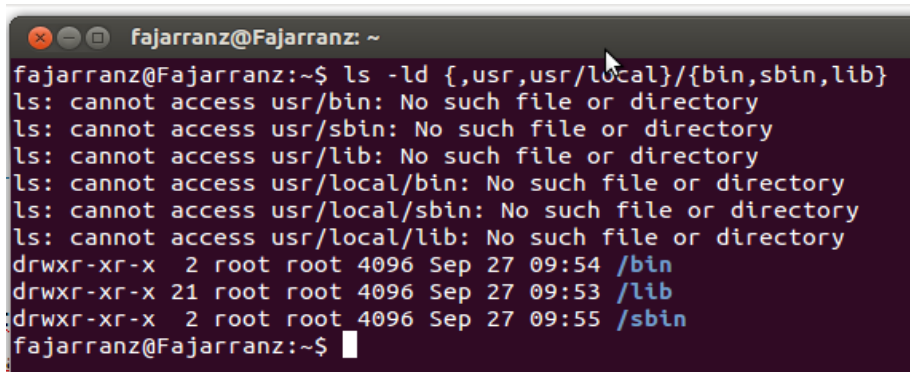
```
#include
#include
#include
typedef enum{false = 0, true = 1}boolean;
int RandomIsi(int jarak);
main() {
int angka, Jawaban;
boolean ketemu;
printf("Game Tebak Angka!\n"); /*Judul Game*/
ketemu=false;
Jawaban=RandomIsi(100); /*buat angka acak antara 1-100*/
while(!ketemu) //Pengulangan
{ printf("Masukan angka tebakan = ");
scanf("%d", &angka); /*Membaca Jawaban angka dari user*/
```

```

if(angka==Jawaban) // Penyeleksian Kondisi Untuk Mengetahui kebenaran
jawaban dari user
{ ketemu=true; //jika sama berarti jawaban benar
}else
{ if(angka { printf("Terlalu kecil\n"); //jika lebih kecil
}else
{ printf("Terlalu besar\n"); //jika lebih besar
}
}
}
if(ketemu)
{ printf("Tebakan anda benar!\n"); }
}
/*fungsi membuat angka acak*/
int RandomIsi(int jarak)
{ int RandomAngka; srand(time(NULL)); /*memakai fungsi time agar hasil angka
acak berbeda-beda*/
RandomAngka=(rand()%jarak)+1;
return RandomAngka; }

```

3.



```

fajarranz@Fajarranz: ~
fajarranz@Fajarranz:~$ ls -ld {,usr,usr/local}/{bin,sbin,lib}
ls: cannot access usr/bin: No such file or directory
ls: cannot access usr/sbin: No such file or directory
ls: cannot access usr/lib: No such file or directory
ls: cannot access usr/local/bin: No such file or directory
ls: cannot access usr/local/sbin: No such file or directory
ls: cannot access usr/local/lib: No such file or directory
drwxr-xr-x  2 root root 4096 Sep 27 09:54 /bin
drwxr-xr-x 21 root root 4096 Sep 27 09:53 /lib
drwxr-xr-x  2 root root 4096 Sep 27 09:55 /sbin
fajarranz@Fajarranz:~$

```

BAB IV

TUGAS PRAKTIKUM

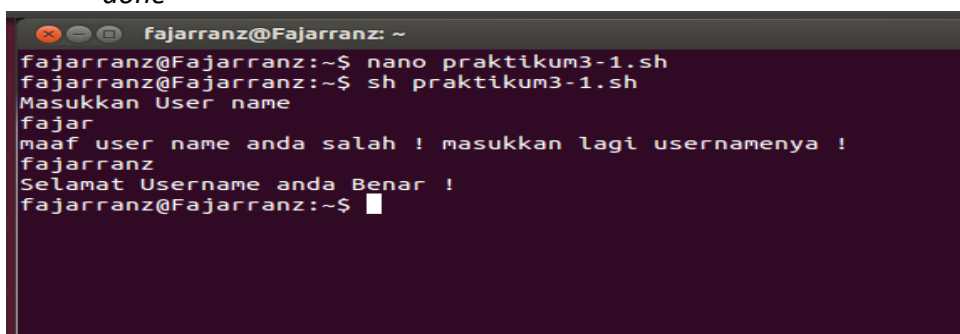
4.1 Tugas Praktikum

1. Buat sebuah skrip untuk memeriksa keshahihan username dari sistem Linux! Pengguna memberikan inputan dari keyboard (gunakan perintah read).
2. Buat sebuah program untuk menampilkan daftar semua pengguna yang ada di dalam sistem Linux mencakup Username, Nama lengkap dan Home directorynya!
3. Selesaikan program tebak angka pada tugas pendahuluan dengan menggunakan perintah case...esac!

Jawaban:

1.

```
#!/bin/sh
echo "Masukkan User name"
while :
do
read INPUT_STRING
case $INPUT_STRING in `whoami`)
echo "Selamat Username anda Benar !"
break
;;
*)
echo "maaf user name anda salah ! masukkan lagi usernamenya !"
;;
esac
done
```



```
fajarranz@Fajarranz: ~
fajarranz@Fajarranz:~$ nano praktikum3-1.sh
fajarranz@Fajarranz:~$ sh praktikum3-1.sh
Masukkan User name
fajar
maaf user name anda salah ! masukkan lagi usernamenya !
fajarranz
Selamat Username anda Benar !
fajarranz@Fajarranz:~$
```

2.

```
#!/bin/sh
echo "Menampilkan nama dan direktori dari user"
finger -l
```



```
3. #!/bin/sh
echo "Tebak Angka"
echo "Ketikkan Angka Tebakanmu"
while :
do
read INPUT_STRING
case $INPUT_STRING in 6)
echo "Selamat Tebakan anda Benar !"
break
;;
*)
echo "maaf Tebakan anda salah ! coba lagi ! "
;;
esac
done
```

```
fajarranz@Fajarranz:~$ sh praktikum3-2.sh
Tebak Angka
Ketikkan Angka Tebakanmu
2
maaf Tebakan anda salah ! coba lagi !
27
maaf Tebakan anda salah ! coba lagi !
87
maaf Tebakan anda salah ! coba lagi !
25
maaf Tebakan anda salah ! coba lagi !
78
maaf Tebakan anda salah ! coba lagi !
27
maaf Tebakan anda salah ! coba lagi !
6
Selamat Tebakan anda Benar !
fajarranz@Fajarranz:~$
```

BAB V

TUGAS TAMBAHAN

1. Membuat program untuk menampilkan Luas segitiga,persegi,persegi panjang dan lingkaran dengan menggunakan case.

jawaban :

```
#!/bin/sh
echo "Masukkan Menu Pilihan"
echo "1. Luas Segitiga"
echo "2. Luas Persegi"
echo "3. Luas Persegi Panjang"
echo "4. Luas Lingkaran"
echo "5. Keluar"
while :
do
read INPUT_STRING
case $INPUT_STRING in
1)
echo "Luas Segitiga"
echo "alas : "
read alas
echo "masukkan tinggi"
read tinggi
luas=`echo 0.5*$alas*$tinggi | bc`
echo "Luasnya adalah = $luas "
echo "masukkan menu lagi"
;;
2)
echo "Luas Persegi"
echo "sisi : "
read sisi
luas=`echo $sisi*$sisi | bc`
echo "Luasnya adalah = $luas "
echo "masukkan menu lagi"
;;
3)
echo "Luas Persegi Panjang"
echo "panjang : "
read panjang
echo "lebar : "
read lebar
luas=`echo $panjang*$lebar | bc`
```

```

echo "Luasnya adalah = $luas "
echo "masukkan menu lagi"
;;

4)
echo "luas lingkaran"
echo "masukkan jari jari :"
read jarijari
luas=`echo 3.14*$jarijari*$jarijari | bc`
echo "Luasnya adalah = $luas "
echo "masukkan menu lagi"
;;

5)
echo "terimakasih !"
break
;;

*)
echo "maaf menu salah ! Coba Lagi !"
esac
done

```

```

fajarranz@Fajarranz: ~
Masukkan Menu Pilihan
1. Luas Segitiga
2. Luas Persegi
3. Luas Persegi Panjang
4. Luas Lingkaran
5. Keluar
6
maaf menu salah ! Coba Lagi !
1
Luas Segitiga
alas :
2
masukkan tinggi
5
Luasnya adalah = 5.0
masukkan menu lagi
2
Luas Persegi
sisi :
4
Luasnya adalah = 16
masukkan menu lagi
3
Luas Persegi Panjang
panjang :
5
lebar :
6
Luasnya adalah = 30
masukkan menu lagi
4
Luas lingkaran
masukkan jari jari :
5
Luasnya adalah = 78.50
masukkan menu lagi
5
terimakasih !
fajarranz@Fajarranz:~$

```

2. Membuat program konversi Desimal ke biner

jawaban :

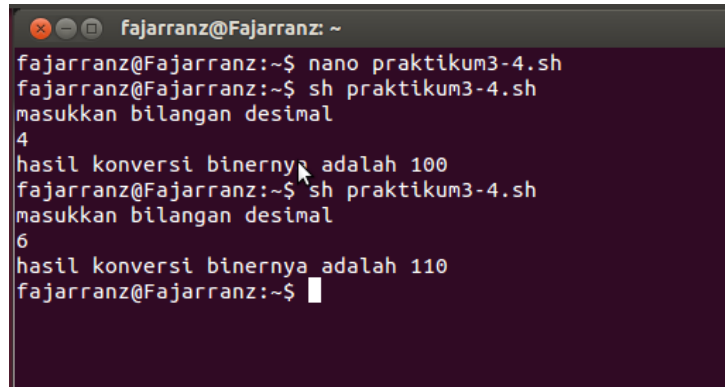
```
#!/bin/bash
```

```
echo "masukkan bilangan desimal"
```

```
read desimal
```

```
biner=`echo "obase=2; $desimal" | bc`
```

```
echo "hasil konversi binernya adalah $biner"
```

A screenshot of a terminal window with a dark background. The window title is 'fajarranz@Fajarranz: ~'. The user has executed the following commands: 'nano praktikum3-4.sh', 'sh praktikum3-4.sh', and 'sh praktikum3-4.sh' again. The script prompts for a decimal number. In the first run, '4' is entered, and the output is 'hasil konversi binernya adalah 100'. In the second run, '6' is entered, and the output is 'hasil konversi binernya adalah 110'. The prompt 'fajarranz@Fajarranz:~\$' is visible at the end of each command line.

```
fajarranz@Fajarranz:~$ nano praktikum3-4.sh
fajarranz@Fajarranz:~$ sh praktikum3-4.sh
masukkan bilangan desimal
4
hasil konversi binernya adalah 100
fajarranz@Fajarranz:~$ sh praktikum3-4.sh
masukkan bilangan desimal
6
hasil konversi binernya adalah 110
fajarranz@Fajarranz:~$
```

BAB VI

PENUTUP

6.1 Kesimpulan

Pengulangan dan pengkondisian sangat diperlukan dalam memecahkan sebuah masalah yang memiliki perulangan ataupun pengkondisian.

6.2 Saran

Asprak lebih jelas dalam menerangkan materi yang akan diajarkan kepada praktikan.