

Nama : Radya Ardi Ninang Pudyastuti

NPM : 21083010097

Kelas : Sistem Operasi A

## BRIEF TUGAS 4

### PERULANGAN

**Perulangan** adalah struktur program yang dapat digunakan untuk melakukan statement yang sama secara terus-menerus dengan cara yang efisien dan efektif selama kondisi tersebut terpenuhi (bernilai *true*)

#### 1. While Loop

- While loop digunakan untuk mengeksekusi serangkaian perintah berulang kali selama suatu kondisi terpenuhi. Perintah yang dilakukan akan tetap berjalan sampai perintah kondisi berhasil dijalankan.
- Pada fungsi perulangan ini kita akan melihat apakah a lebih kecil daripada operan sebelah kanan. Ketika  $a=0$ , kemudian pada syntax  $a=$((a + 2))$  itu berarti nilai a akan ditambah 2 pada setiap operasi perbandingannya.

```
radyaardi@radyaardi: ~/Tugas4
Berkas  Sunting  Tampilan  Cari  Terminal  Bantuan
radyaardi@radyaardi:~/Tugas4$ nano while_loop.sh
radyaardi@radyaardi:~/Tugas4$ bash while_loop.sh
0
2
4
6
8
radyaardi@radyaardi:~/Tugas4$
```

```
radyaardi@radyaardi: ~/Tugas4
Berkas  Sunting  Tampilan  Cari  Terminal  Bantuan
GNU nano 2.9.3      while_loop.sh      Modified

a=0

while [ $a -lt 10 ]
do
    echo $a
    a=$((a + 2))
done
```

## 2. For Loop1

- For loop digunakan untuk mengulang serangkaian perintah untuk setiap item pada daftar.
- Sebelum itu, kita membuat syntax menggunakan `nano for_loop.sh`
- Kemudian pada syntax for loop yang pertama, maknanya adalah

**For angka in 1 2 3 4 5**

Untuk angka(value individual dari variabel) di 1 2 3 4 5 (atau item yang instruksinya ingin anda ikuti)

**Do**

Lakukan (keyword ini digunakan untuk mengaktifkan loop)

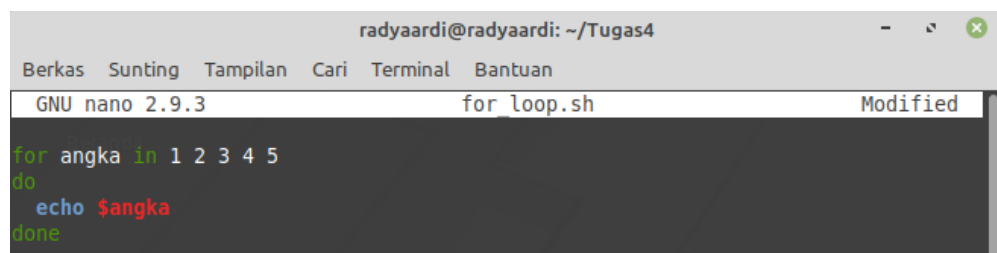
**Echo \$angka**

echo adalah kode yang digunakan untuk memanggil angka (value individual dari variabel yang telah kita misalkan tadi)

**done**

done digunakan untuk memberhentikan loop

```
radyaardi@radyaardi:~/Tugas4$ nano for_loop.sh
radyaardi@radyaardi:~/Tugas4$ bash for_loop.sh
1
2
3
4
5
radyaardi@radyaardi:~/Tugas4$
```



```
radyaardi@radyaardi: ~/Tugas4
Berkas Sunting Tampilan Cari Terminal Bantuan
GNU nano 2.9.3 for_loop.sh Modified

for angka in 1 2 3 4 5
do
  echo $angka
done
```

## 3. For Loop2

- Sebelum itu, kita membuat syntax menggunakan `nano for_loop2.sh`
- Kemudian pada syntax for loop yang kedua, maknanya adalah

**For ({angka=1; angka<=5; angka=angka+1})**

**Do**

Lakukan (keyword ini digunakan untuk mengaktifkan loop)

**Echo \$angka**

echo adalah kode yang digunakan untuk memanggil angka (value individual dari variabel yang telah kita misalkan tadi)

**done**

done digunakan untuk memberhentikan loop

```
radyaardi@radyaardi: ~/Tugas4
Berkas  Sunting  Tampilan  Cari  Terminal  Bantuan
5
radyaardi@radyaardi:~/Tugas4$ nano for_loop2.sh
radyaardi@radyaardi:~/Tugas4$ bash for_loop2.sh
1
2
3
4
5
```

```
radyaardi@radyaardi: ~/Tugas4
Berkas  Sunting  Tampilan  Cari  Terminal  Bantuan
GNU nano 2.9.3      for_loop2.sh      Modified
for ((angka=1; angka<=5; angka=angka+1))
do
    echo $angka
done
```

#### 4. Select Loop

- Select loop digunakan ketika kita ingin membuat sebuah program dengan beberapa daftar pilihan yang bisa dipilih oleh user, misalnya daftar menu
- Dalam fungsi select dibawah ini, kami menggunakan daftar minuman, yaitu teh, kopi, air, jus, susu, semua, gaada, untuk memungkinkan pengguna memilih minuman yang mereka inginkan. Tetapi dengan beberapa kondisi. Yaitu, ketika pengguna memilih teh, kopi, air, semua, maka sistem akan mengeluarkan kalimat “maaf habis”. Ketika pengguna memilih jus dan susu, maka akan muncul kalimat “tersedia”. Namun ketika pengguna memilih gaada (tidak ada) maka akan muncul kalimat “tidak ada di daftar menu”.

```
radyaardi@radyaardi:~/Tugas4$ nano select.sh
radyaardi@radyaardi:~/Tugas4$ bash select.sh
1) teh
2) kopi
3) air
4) jus
5) susu
6) semua
7) gaada
#? 6
maaf, habis
#? 5
Tersedia
#? 1
maaf, habis
#? ^C
radyaardi@radyaardi:~/Tugas4$
```

```
radyaardi@radyaardi: ~/Tugas4
Berkas Sunting Tampilan Cari Terminal Bantuan
GNU nano 2.9.3 select.sh Modified

select minuman in teh kopi sir jus susu semua gaada
do
    case $minuman in
        teh|kopi|air|semua)
            echo "maaf, habis"
            ;;
        jus|susu)
            echo "Tersedia"
            ;;
        gaada)
            break
            ;;
        *) echo "Tidak ada di daftar menu"
            ;;
    esac
done
```

## 5. Until Loop

- Until loop digunakan untuk mengeksekusi serangkaian perintah berulang kali sampai suatu kondisi terpenuhi. Perulangan ini hampir sama dengan while, hanya saja jika pada perulangan while loop, instruksi dalam perulangan dilakukan sewaktu kondisi benar (*true*). Sedangkan pada perulangan until, instruksi dalam perulangan dilakukan sewaktu kondisi salah (*false*).

```
radyaardi@radyaardi:~/Tugas4$ nano until_loop.sh
radyaardi@radyaardi:~/Tugas4$ bash until_loop.sh
0
2
4
6
8
radyaardi@radyaardi:~/Tugas4$
```

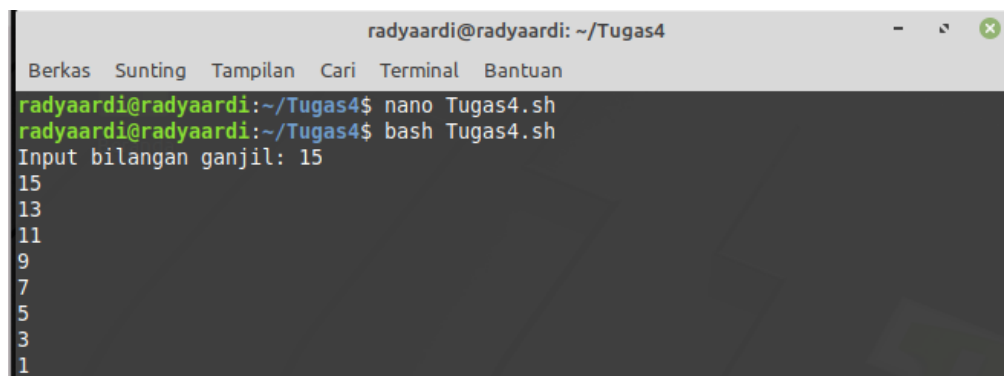
```
radyaardi@radyaardi: ~/Tugas4
Berkas Sunting Tampilan Cari Terminal Bantuan
GNU nano 2.9.3 until_loop.sh Modified

a=0


until [ ! $a -lt 10 ]
do
    echo $a
    a=$((a + 2))
done
```

## 6. Latihan Soal

- Buatlah program perulangan yang menerapkan beberapa konsep pemrograman bash seperti di atas dengan ketentuan sebagai berikut!
  - User menginputkan acuan bilangan
  - Mencetak bilangan positif kelipatan ganjil 3 5 6 7 9 dst dari bilangan acuan.
- Pada tugas kali ini, untuk menginputkan acuan bilangan saya menggunakan fungsi input. Kemudian saya menggunakan fungsi until loop dalam pengeksekusian.
- Ketika a=0, sistem akan memeriksa apakah nilai operan kiri (bilangan ganjil) lebih besar daripada operan kanan (a).  
Dalam hal ini, bilangan ganjil perbandingan didapatkan dari bilangan ganjil-2. Kemudian akan muncul hasilnya.



```
radyaardi@radyaardi: ~/Tugas4
Berkas  Sunting  Tampilan  Cari  Terminal  Bantuan
radyaardi@radyaardi:~/Tugas4$ nano Tugas4.sh
radyaardi@radyaardi:~/Tugas4$ bash Tugas4.sh
Input bilangan ganjil: 15
15
13
11
9
7
5
3
1
```



```
radyaardi@radyaardi: ~/Tugas4
Berkas  Sunting  Tampilan  Cari  Terminal  Bantuan
GNU nano 2.9.3      Tugas4.sh

echo -n "Input bilangan ganjil: ";
read ganjil;

a=0

until [ ! $ganjil -gt $a ]
do
    echo $ganjil
    ganjil=$((ganjil-2))
done
```