

Tugas 4

Pemrograman Shell 2

Nama : Muhamad Haris Hartanto

NPM : 21083010045

Kelas : Sistem Operasi – A

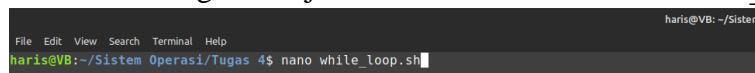
A. Tutorial

1. While Loop

Merupakan perulangan yang terus dilakukan selama kondisi yang didefinisikan terpenuhi.

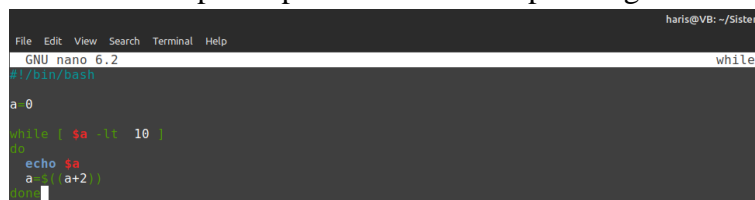
a. Membuat file

Dilakukan dengan menjalankan command `nano while_loop.sh`.



b. Menuliskan skrip

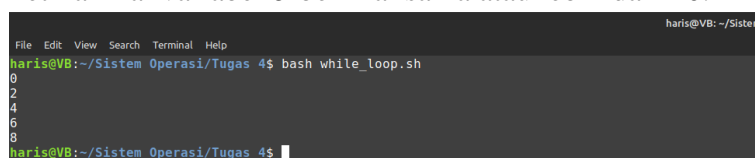
- Diinisialisasikan variabel `a` dengan nilai 0 (`a=0`).
- Ketika (`while`) nilai variabel `a` kurang dari 10 (`[$a -lt 10]`),
Dicetak nilai variabel `a` (`echo $a`), kemudian dilakukan operasi penjumlahan terhadap nilai variabel `a` dengan ditambahkan bilangan 2 (`a=$((a+2))`) yang ditampung dalam variabel `a`.
- Ketika variabel `a` bernilai sama atau lebih dari 10, perulangan akan terhenti.
- `done` merupakan penanda akhir dari perulangan.



c. Menjalankan skrip

Baris pertama dihasilkan output 0 dikarenakan variabel `a` diinisialisasikan dengan nilai 0, lalu dilakukan operasi penjumlahan terhadap nilai variabel `a` dengan ditambahkan dengan bilangan 2 yang ditampung oleh variabel `a` sehingga variabel `a` yang sebelumnya bernilai 0 (`a=0`) berubah menjadi 2 (`a=2`).

Akibatnya, pada baris ke-2 dihasilkan output berupa 2. Operasi ini akan terus berlanjut selama kondisi yang didefinisikan (`[$a -lt 10]`) terpenuhi, dan akan terhenti ketika nilai variabel `a` bernilai sama atau lebih dari 10.



2. For Loop (1)

Merupakan perulangan terhadap item yang telah didefinisikan.

a. Membuat file (for_loop1.sh)

Dilakukan dengan menjalankan command `nano for_loop1.sh`.

```
File Edit View Search Terminal Help
haris@VB: ~/Sistem
haris@VB:~/Sistem Operasi/Tugas 4$ nano for_loop1.sh
```

b. Menuliskan skrip

- angka merupakan pengganti untuk variabel.
- Untuk (for) setiap angka di dalam (in) item (1 2 3 4 5)
- Lakukan (do) → `echo $angka`

Dilakukan `echo $angka` untuk setiap elemen yang terdapat dalam item sebanyak `n` kali.

`*n` merupakan jumlah elemen dalam item, yaitu 5.

- `done` merupakan penanda akhir dari perulangan.

```
File Edit View Search Terminal Help
GNU nano 6.2
# /bin/bash
for angka in 1 2 3 4 5
do
    echo $angka
done
```

c. Menjalankan skrip

Dihasilkan output berupa setiap elemen yang terdapat pada item yang dicetak per baris.

```
File Edit View Search Terminal Help
haris@VB: ~/Sistem
haris@VB:~/Sistem Operasi/Tugas 4$ bash for_loop1.sh
1
2
3
4
5
haris@VB:~/Sistem Operasi/Tugas 4$
```

3. For Loop (2)

a. Membuat file

Dilakukan dengan menjalankan command `nano for_loop2.sh`.

```
File Edit View Search Terminal Help
haris@VB: ~/Sistem
haris@VB:~/Sistem Operasi/Tugas 4$ nano for_loop2.sh
```

b. Menuliskan skrip

- Untuk (for) angka dengan elemen item yang diawali dengan bilangan 1 (`angka=1`), sampai kurang dari atau sama dengan bilangan 5 (`angka<=5`), dengan interval +1 antara setiap elemen atau bilangan (`angka+=1`). Dengan demikian, item akan berisi bilangan 1, 2, 3, 4, 5.

- Lakukan (do) → `echo $angka`

Dilakukan `echo $angka` setiap elemen yang terdapat dalam item sebanyak `n` kali.

`*n` merupakan jumlah elemen dalam item, yaitu 5.

- `done` merupakan penanda akhir dari perulangan.

```
haris@VB: ~/Sistem
File Edit View Search Terminal Help
GNU nano 6.2 for loop
#!/bin/bash

for ((angka=1; angka<=5; angka=angka+1))
do
    echo $angka
done
```

c. Menjalankan skrip

Dihasilkan output berupa setiap elemen yang terdapat pada item yang dicetak per baris.

```
haris@VB: ~/Sistem
File Edit View Search Terminal Help
haris@VB:~/Sistem Operasi/Tugas 4$ bash for_loop2.sh
1
2
3
4
5
haris@VB:~/Sistem Operasi/Tugas 4$
```

4. Select Loop

Merupakan perulangan yang menyediakan pilihan bernomor dari setiap elemen yang terdapat dalam item.

a. Membuat file

Dilakukan dengan menjalankan command `select_loop.sh`.

```
haris@VB: ~/Sistem
File Edit View Search Terminal Help
haris@VB:~/Sistem Operasi/Tugas 4$ nano select_loop.sh
```

b. Menuliskan skrip

- minuman merupakan pengganti untuk variabel.
- `select minuman` di dalam `(in)` item dengan elemen teh, kopi, air, jus, susu, semua, gaada.
- Lakukan `(do)` percabangan `case . . esac` dengan pola sebagai berikut:
 - Apabila memilih nomor elemen teh/kopi/air/semua akan ditampilkan output “Maaf, habis”.
 - Apabila memilih nomor elemen jus/susu akan ditampilkan output “Tersedia”
 - Apabila memilih nomor elemen gaada perulangan akan terhenti (`break`).
 - Apabila memilih nomor selain dari nomor elemen yang tersedia dalam item, maka akan ditampilkan output berupa “Tidak ada di daftar menu”.
- `done` merupakan penanda akhir dari perulangan.

```
haris@VB: ~/Sistem
File Edit View Search Terminal Help
GNU nano 6.2 select loop
#!/bin/bash

select minuman in teh kopi air 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
```

c. Menjalankan skrip

Diberikan kolom input untuk user memilih salah satu dari elemen yang terdapat dalam item berdasarkan nomor indeks yang tertera. Hasil dari input yang dilakukan oleh user akan menyesuaikan kondisi percabangan yang didefinisikan saat menuliskan skrip.

```

haris@VB: ~/Sistem
File Edit View Search Terminal Help
haris@VB:~/Sistem Operasi/Tugas 4$ bash select_loop.sh
1) teh
2) kopi
3) air
4) jus
5) susu
6) semua
7) gaada
#? 1
Maaf, habis
#? 2
Maaf, habis
#? 3
Maaf, habis
#? 4
Tersedia
#? 5
Tersedia
#? 6
Maaf, habis
#? 0
Tidak ada di daftar menu
#? 7
haris@VB:~/Sistem Operasi/Tugas 4$

```

5. Until Loop

Merupakan perulangan yang akan berhenti ketika kondisi yang didefinisikan terpenuhi.

a. Membuat file

Dilakukan dengan menjalankan command `until_loop.sh`.

```

haris@VB: ~/Sistem
File Edit View Search Terminal Help
haris@VB:~/Sistem Operasi/Tugas 4$ nano until_loop.sh

```

b. Menuliskan skrip

- Diinisialisasikan variabel `a` dengan nilai 0 (`a=0`).
- Sampai (`until`) nilai variabel `a` tidak kurang dari 10 (`[! $a -lt 10]`), dicetak nilai variabel `a` (`echo $a`), lalu dilakukan operasi penjumlahan terhadap nilai variabel `a` dengan ditambahkan bilangan 2 yang ditampung dalam variabel `a` (`a=$((a+2))`).
- Ketika nilai variabel `a` bernilai sama atau lebih dari 10, perulangan akan berhenti.
- `done` merupakan penanda akhir dari perulangan.

```

haris@VB: ~/Sistem
File Edit View Search Terminal Help
GNU nano 6.2
# until loop
a=0
until [ ! $a -lt 10 ]
do
    echo $a
    a=$((a+2))
done

```

c. Menjalankan skrip

Dihasilkan output berupa nilai variabel `a` yang terus bertambah (+2), dan ketika kondisi yang didefinisikan terpenuhi (`[! $a -lt 10]`) perulangan akan berhenti.

```

haris@VB: ~/Sistem
File Edit View Search Terminal Help
haris@VB:~/Sistem Operasi/Tugas 4$ bash until_loop.sh
0
2
4
6
8
haris@VB:~/Sistem Operasi/Tugas 4$

```

B. Latihan Soal

1. Membuat file (`tugas_4.sh`)

Dilakukan dengan menjalankan command `nano tugas_4.sh`.

```
File Edit View Search Terminal Help
haris@VB: ~/Sistem Operasi/Tugas 4$ nano Tugas_4.sh
```

2. Menuliskan skrip

```
File Edit View Search Terminal Help
GNU nano 6.2
Tugas
#!/bin/bash

header() {
  msg="# $* #"
  edge=$(echo "$msg" | sed 's/./=/g')
  echo "$edge"
  echo "$msg"
  echo "$edge"
}

header "Bilangan Positif Kelipatan Ganjil"
echo ""
read -p 'Input: ' bil
let gg=$bil%2
echo ""
if [ $bil -gt 0 ]
then
  if [ $gg == 0 ]
  then
    echo $bil merupakan bilangan genap.
    bil=$((bil+1))
    bil2=$((bil-1))
    echo -e "Rekomendasi bilangan yang tepat: $bil1 atau $bil2\n"
    echo -e "Input: $bil1\n"
    while [ $bil1 -gt 0 ]
    do
      echo $bil1
      bil1=$((bil1-2))
    done
    echo -e "\natau\n"
    echo -e "Input: $bil2\n"
    until [ ! $bil2 -gt 0 ]
    do
      echo $bil2
      bil2=$((bil2-2))
    done
  else
    while [ $bil -gt 0 ]
    do
      echo $bil
      bil=$((bil-2))
    done
  fi
else
  echo "Masukkan bilangan positif!"
fi
```

- `read -p 'Input: ' bil`
Hasil input bilangan yang dilakukan user ditampung oleh variabel bil.
- `let gg=$bil%2`
Dilakukan perhitungan modulus 2 (%2) terhadap nilai variabel bil untuk memeriksa bilangan yang diinputkan oleh user merupakan bilangan genap atau ganjil. Hasil perhitungan ditampung oleh variabel gg.
- Percabangan pertama
Apabila nilai variabel a lebih besar dari 0 (`[$a > 0]`), maka akan dilanjutkan ke tahap selanjutnya. Namun, jika tidak lebih besar dari 0, maka akan diberikan output berupa "Masukkan bilangan positif!".
- Percabangan kedua
Apabila hasil perhitungan modulus = 0 atau dapat dikatakan sebagai bilangan genap, maka akan diberikan output berupa "\$bil merupakan bilangan genap."
Selanjutnya dilakukan operasi penjumlahan (+1) dan pengurangan (-1) terhadap nilai variabel bil agar menjadi bilangan ganjil. Operasi penjumlahan disimpan dalam

variabel `bil1` dan operasi pengurangan disimpan dalam variabel `bil2`. Kemudian dilakukan perulangan `while` sebagai berikut:

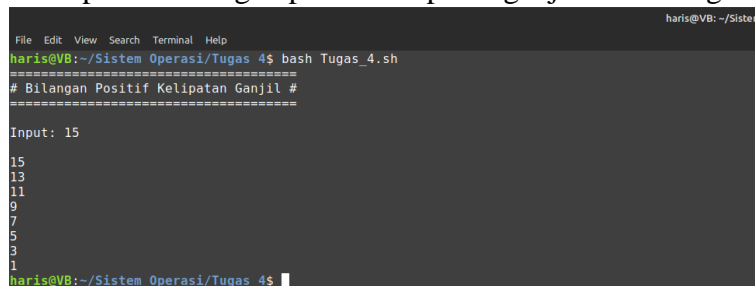
- Perulangan `while`
Dilakukan perulangan `while` dengan kondisi nilai variabel `bil1` lebih besar dari 0 (`[$bil1 > 0]`), dengan command `echo $bil1` dan dilakukan operasi pengurangan (-2) terhadap nilai variabel `bil1` (`bil1=$((bil1-2))`).
- Perulangan `until`
Dilakukan perulangan `until` dengan kondisi nilai variabel `bil2` tidak lebih besar dari 0 (`[! $bil2 > 0]`), dengan command `echo $bil2` dan dilakukan operasi pengurangan (-2) terhadap nilai variabel `bil2` (`bil2=$((bil2-2))`).

Namun, jika hasil perhitungan modulus $\neq 0$ atau dapat dikatakan sebagai bilangan ganjil, maka akan langsung dilakukan perulangan `while` sebagai berikut:

- Perulangan `while`
Dilakukan perulangan `while` dengan kondisi nilai variabel `bil` lebih besar dari 0 (`[$bil1 > 0]`), dengan command `echo $bil` dan dilakukan operasi pengurangan (-2) terhadap nilai variabel `bil1` (`bil1=$((bil1-2))`).

3. Menjalankan skrip

- Menginputkan bilangan ganjil
Diinputkan bilangan 15 yang merupakan bilangan ganjil sehingga dapat langsung ditampilkan bilangan positif kelipatan ganjil dari bilangan 15 sebagai berikut:



```
haris@VB: ~/Sistem Operasi/Tugas 4$ bash Tugas_4.sh
# Bilangan Positif Kelipatan Ganjil #
Input: 15
15
13
11
9
7
5
3
1
haris@VB: ~/Sistem Operasi/Tugas 4$
```

- Menginputkan bilangan genap
Diinputkan bilangan 14 yang merupakan bilangan genap sehingga ditampilkan output “14 merupakan bilangan genap.” dan diberikan rekomendasi bilangan bulat ganjil terdekat dari 14, yaitu 15 dan 13. Sesudah itu, ditampilkan bilangan positif kelipatan ganjil dari bilangan 15 dan 13.

```
File Edit View Search Terminal Help
haris@VB: ~/Sistem
haris@VB:~/Sistem Operasi/Tugas 4$ bash Tugas_4.sh
=====
# Bilangan Positif Kelipatan Ganjil #
=====

Input: 14
14 merupakan bilangan genap.
Rekomendasi bilangan yang tepat: 15 atau 13

Input: 15
15
13
11
9
7
5
3
1
atau
Input: 13
13
11
9
7
5
3
1
haris@VB:~/Sistem Operasi/Tugas 4$
```

- Mengiinputkan bilangan kurang dari atau sama dengan 0
Diinputkan bilangan -15 yang merupakan bilangan negatif sehingga ditampilkan output berupa “Masukkan bilangan positif!”.

```
File Edit View Search Terminal Help
haris@VB: ~/Sistem
haris@VB:~/Sistem Operasi/Tugas 4$ bash Tugas_4.sh
=====
# Bilangan Positif Kelipatan Ganjil #
=====

Input: -15
Masukkan bilangan positif!
haris@VB:~/Sistem Operasi/Tugas 4$
```