

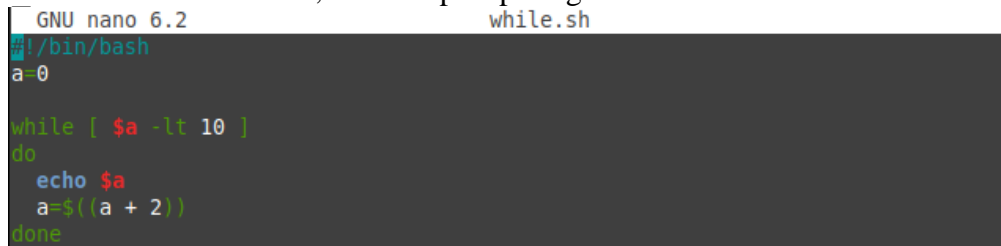
Nama : Aisyah Kirana Putri Isyanto
NPM : 21083010065
Kelas : Sistem Operasi

DOKUMENTASI PRAKTIKUM MINGGU-4 (PERULANGAN/LOOP)

Perulangan (Loop) merupakan 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). Macam-macam perulangan antara lain:

1. While Loop

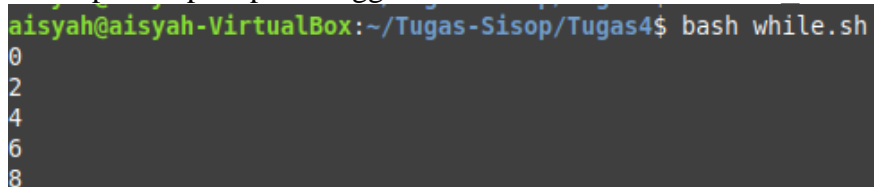
While loop digunakan untuk mengeksekusi serangkaian perintah berulang kali selama suatu kondisi terpenuhi. Pertama, buat text editor dengan command `nano` nama `file.sh`. Kemudian, buat script seperti gambar dibawah.



```
GNU nano 6.2                                while.sh
#!/bin/bash
a=0

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

Penjelasan script: buat variabel `a` yang dimulai dengan nilai 0. Setiap variabel yang nilainya lebih kecil dari 10 akan dicetak berurutan lalu ditambah dengan 2. Untuk menampilkan output script dapat menggunakan command `bash` nama `file.sh`.



```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash while.sh
0
2
4
6
8
```

2. For Loop

- For Loop 1



```
GNU nano 6.2                                forloop.sh
#!/bin/bash

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

Pada dasarnya, sintaks loop yang paling sederhana mengulangi terjadinya satu set variabel. Pada gambar diatas, langkah pertama yaitu membuat variabel awal bernama 'angka'. Sementara itu, `$angka` adalah nilai individual dari variabel tersebut. Kemudian command `do` yang merupakan kata kunci yang memulai loop akan menjalankan intruksi sebanyak `n` kali, dengan `n` menjadi jumlah total item. Sehingga

script dapat mencetak setiap elemen dari daftar yang disediakan. Berikut adalah outputnya:

```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash forloop.sh
1
2
3
4
5
```

- For Loop 2

```
GNU nano 6.2 forloop2.sh
#!/bin/bash

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

Kode menjelaskan bahwa angka awal bernilai 1. Loop akan dieksekusi, selama kondisi 2 benar, yang berarti tidak boleh lebih besar dari 5. Selain itu, angka+1 menunjukkan bahwa tiap angka naik 1 nilai. Ini kemudian akan mengulangi loop satu per satu mulai dari nilai awal. Outputnya adalah:

```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash forloop2.sh
1
2
3
4
5
```

3. Select Loop

Select Loop adalah alternatif untuk skrip yang mengharuskan pengguna meneruskan argumen untuk melakukan suatu tindakan. Owner dapat menambahkan lebih banyak argumen dan pengguna hanya perlu memilih opsi agar program dapat melakukan tugasnya.

```
GNU nano 6.2 select.sh
#!/bin/bash

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

Pada gambar diatas, looping dimulai dengan membuat variabel “minuman”. Kemudian buat daftar elemen yaitu the, kopi, air, jus, susu, semua, dan gaada. Lalu buat beberapa kondisi untuk dapat menampilkan outputnya.

- Pada kondisi 1, user memilih teh/kopi/air/semua, maka output yang keluar adalah string “Maaf, habis kak”.
- Pada kondisi 2, jika user memilih jus/susu, maka output yang muncul adalah string “Tersedia”.
- Pada kondisi 3, apabila user memilih gaada, maka sistem akan menampilkan output string “Tidak ada di daftar menu”.

```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash select.sh
1) teh
2) kopi
3) air
4) jus
5) susu
6) semua
7) gaada
#? 1
Maaf, habis kak
#? 2
Maaf, habis kak
#? 3
Maaf, habis kak
#? 4
Tersedia
#? 5
Tersedia
#? 6
Maaf, habis kak
#? 7
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$
```

4. Until Loop

Berbeda dengan while, until loop digunakan untuk mengeksekusi serangkaian perintah berulang kali sampai suatu kondisi terpenuhi. Kondisi ini dievaluasi sebelum menjalankan perintah. Jika kondisi mengevaluasi ke false, perintah dijalankan. Jika tidak, jika kondisi mengevaluasi ke true, loop akan dihentikan dan kontrol program akan diteruskan ke perintah berikutnya.

```
GNU nano 6.2                               until.sh
#!/bin/bash

a=0

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

Pada script diatas, buat variabel a yang bernilai 0 untuk menginisiasi nilai. Kemudian dengan command until, buat sebuah kondisi dimana ketika nilai a lebih kecil dari 10 maka nilai a akan dicetak dengan ketentuan nilai a ditambah 2 angka tiap perulangannya. Output dari command until akan muncul seperti gambar dibawah.

```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash until.sh
0
2
4
6
8
```

SOAL LATIHAN

Buatlah program perulangan yang menerapkan beberapa konsep pemrograman bash seperti diatas dengan ketentuan sbb!

- user menginputkan acuan bilangan
- mencetak bilangan positif kelipatan ganjil 3 5 7 9 dst dari bilangan acuan

1. Langkah pertama adalah membuat text editor berformat .sh dengan command nano nama_file.sh.
2. Buat script bash dengan memanfaatkan konsep perulangan

```
GNU nano 6.2                                     tugas4.sh *
#!/bin/bash

index=0

echo -n "Batas Bilangan: "
read batas

while [ $index -lt $batas ]
do
    remainder=$(( $index % 2 ))

    if [ $remainder -ne 0 ]
    then
        echo $index
    fi

    index=$(( $index+1 ))
done

echo "Selesai"
```

- Atur nilai INDEX variabel ke 0.
 - Mulai loop sementara yang dijalankan jika nilai INDEX kurang dari batas yang telah ditentukan.
 - Pada setiap nilai di batas, hitung REMAINDER/SISA apakah habis dibagi 2.
 - Periksa apakah SISA/REMAINDERnya bukan nol, jika benar INDEX akan dicetak karena itu adalah angka ganjil.
 - Tiap nilai INDEX akan ditambah 1 angka sebelum menjalankan iterasi berikutnya dari *while loop*.
3. Output yang dihasilkan seperti gambar dibawah. Untuk menampilkan bilangan-bilangan ganjil, masukkan batas bilangannya terlebih dahulu.

```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash tugas4.sh
Batas Bilangan:
```

4. Berikut ini adalah outputnya:

```
aisyah@aisyah-VirtualBox:~/Tugas-Sisop/Tugas4$ bash tugas4.sh
Batas Bilangan: 15
1
3
5
7
9
11
13
Selesai
```

- Batas yang diinginkan adalah 15, maka akan dilakukan perulangan `while` apakah INDEX lebih kecil dari 15.
- Ternyata benar bahwa INDEX lebih kecil dari batas, maka akan dilanjutkan untuk menentukan apakah pembagiannya habis dibagi 2.
- Periksa apakah SISA/REMAINDERnya bukan nol, jika selain nol maka perhitungan akan dilanjutkan karena merupakan angka ganjil.
- Kemudian, tiap nilai di INDEX akan ditambah 1 angka sebelum menjalankan iterasi berikutnya dari *while loop*.

Index = 0 (tampilan dimulai dari angka 0)

1:2 = sisa 1

2:2 = sisa 0 (tidak dilanjutkan)

3:2 = sisa 1

4:2 = sisa 0 (tidak dilanjutkan)

.

.

.

14:2 = sisa 0 (tidak dilanjutkan)

Karena 15 adalah batas akhir bilangannya, maka angka 15 tidak ikut ditampilkan dalam hasil. Jadi bilangan ganjil yang ditampilkan hanya sampai 13.