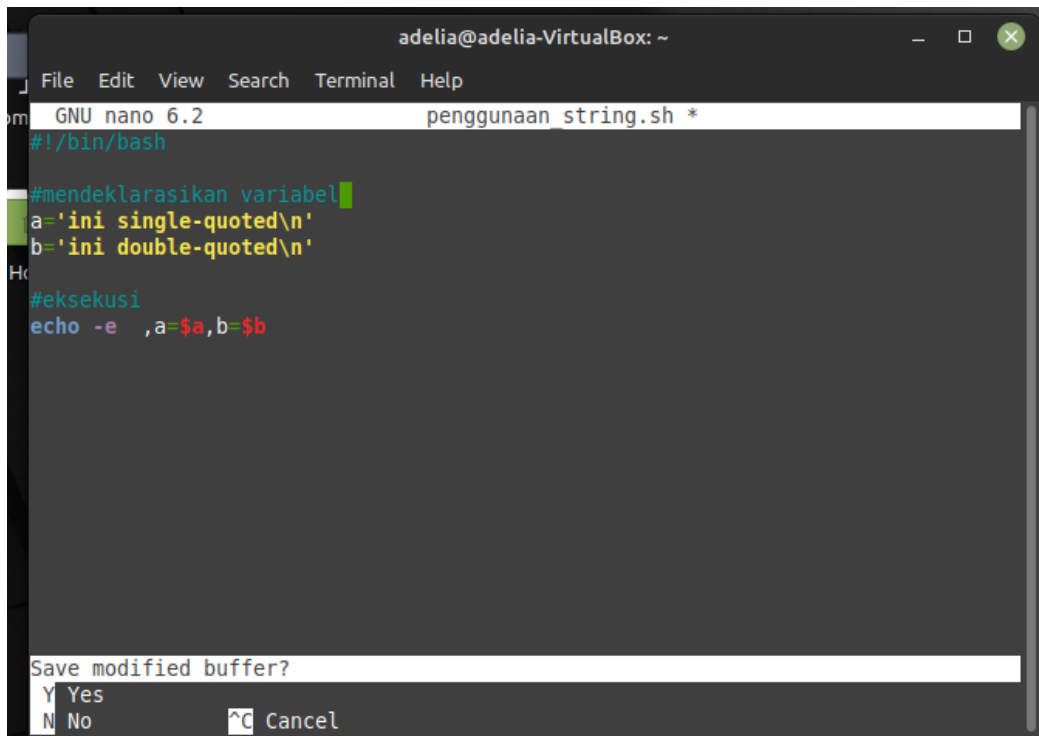


LAPORAN PERTEMUAN KETIGA

Mempelajari terkait pemrograman di dalam linux dengan membuat berkas ber-format .sh yang kemudian dijalankan pada terminal menggunakan perintah **bash nama_berkas.sh**. Lebih jelasnya mempelajari terkait string, array, integer, operasi matematika, input, output, dan percabangan.

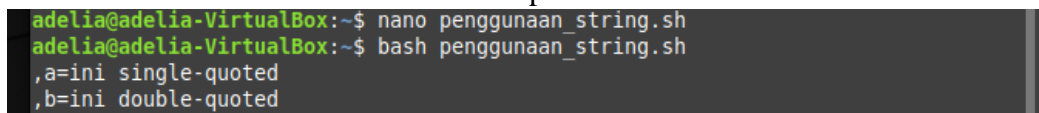
1. Penggunaan String

Mendeklarasikan string di dalam variable dengan **nama_variabel='string'**, tidak menggunakan space sebelum atau sesudah tanda sama dengan, karena kesalahan space saja berdampak pada tidak dapat dijalanannya perintah. Kemudian, ketika ingin memanggil atau mencetak variabel, menggunakan perintah **echo -e \$nama_variabel**. Tanda \$ mendeskripsikan nilai dari suatu variabel.



```
adelia@adelia-VirtualBox: ~  
File Edit View Search Terminal Help  
GNU nano 6.2      penggunaan_string.sh *  
#!/bin/bash  
  
#mendeklarasikan variabel  
a='ini single-quoted\  
b='ini double-quoted\  
  
#eksekusi  
echo -e ,a=$a,b=$b  
  
Save modified buffer?  
Y Yes  
N No      ^C Cancel
```

Output:



```
adelia@adelia-VirtualBox:~$ nano penggunaan_string.sh  
adelia@adelia-VirtualBox:~$ bash penggunaan_string.sh  
,a=ini single-quoted  
,b=ini double-quoted
```

2. Penggunaan Array

Mendeklarasikan array dengan syntax **nama_array=("array1", "array2", "array3")**. Apabila ingin memilih nilai array secara random menggunakan perintah berikut :

let perintah_random=\$RANDOM%angka

kemudian mencetak nilai tersebut dengan **echo \${nama_array[\$perintah_random]}**. Apabila perintah_random dipanggil tanpa array maka akan mencetak angka secara random dengan perintah **echo \$perintah_random**.

```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2      penggunaan_array.sh *
#!/bin/bash

#deklarasi array
distroLinux=("Mint" "Ubuntu" "Kali" "Arch" "Debian")

#random distro
let pilih=$RANDOM%5

#eksekusi
echo "Saya Memilih Distro $pilih, ${distroLinux[$pilih]} !"

```

Output:

```

adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano penggunaan_array.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_array.sh
Saya Memilih Distro 2, Kali !
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_array.sh
Saya Memilih Distro 4, Debian !
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_array.sh
Saya Memilih Distro 0, Mint !
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_array.sh
Saya Memilih Distro 0, Mint !
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_array.sh
Saya Memilih Distro 0, Mint !
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$

```

3. Penggunaan Integer

Mendeklarasikan integer sebagai nilai variabel dengan nama_variabel=integer. Ketika mendeklarasikan nilai integer tidak perlu menggunakan tanda petik. Kemudian kita dapat memanggil variabel baru sebagai hasil operasi nilai integer di dalam variabel dengan perintah **let nama_variabel=integer_1+integer_2**. Memanggil atau mencetak nilai integer dengan perintah **echo** dan variabel yang menyimpan nilai integer tersebut, **echo \$nama_variabel**.

```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2      penggunaan_integer.sh *
#!/bin/bash

#deklarasi integer
a=192
b=479

#operasi integer
let c=a+b

#eksekusi nilai c
echo "$a+$b = $c"

```

Output:

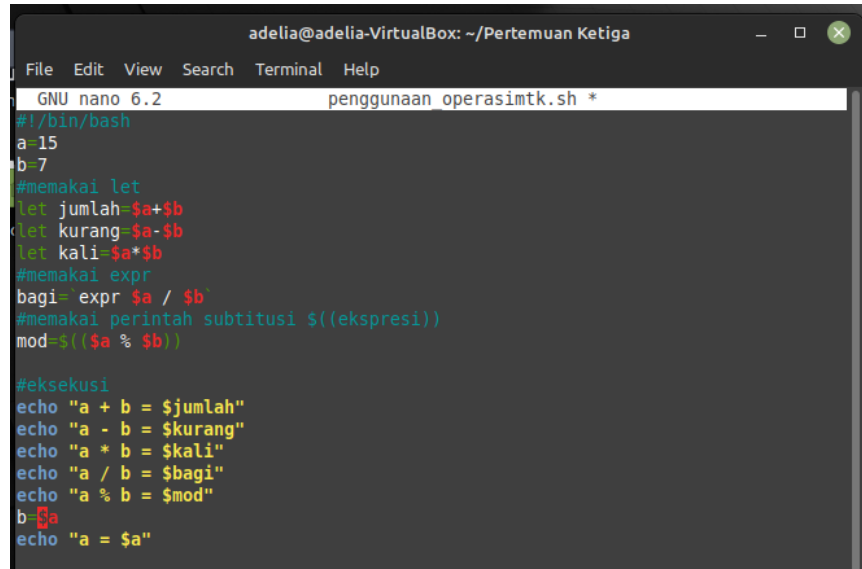
```

adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano penggunaan_integer.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_integer.sh
192+479 = 671

```

4. Penggunaan Operasi Matematika

Integer dapat dioperasikan seperti dijumlah, dikurang, dikali, dibagi dan sebagainya. Dalam pengoperasiannya menggunakan perintah `let`, `expr`, maupun perintah substitusi `$((ekspresi))`. Menurut sumber yang saya baca, `let` digunakan untuk melakukan evaluasi aritmatika dan merupakan built-in shell, `$(())` digunakan untuk melakukan ekspansi aritmatika, dan `expr` adalah perintah biner.



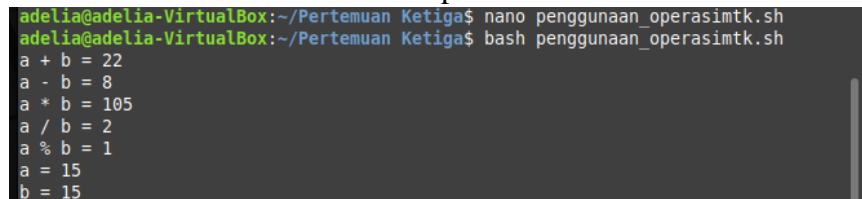
```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2      penggunaan_operasimtk.sh *
#!/bin/bash
a=15
b=7
#memakai let
let jumlah=$a+$b
let kurang=$a-$b
let kali=$a*$b
#memakai expr
bagi=`expr $a / $b`
#memakai perintah substitusi $((ekspresi))
mod=$(( $a % $b ))

#eksekusi
echo "a + b = $jumlah"
echo "a - b = $kurang"
echo "a * b = $kali"
echo "a / b = $bagi"
echo "a % b = $mod"
b=$a
echo "a = $a"

```

Output:



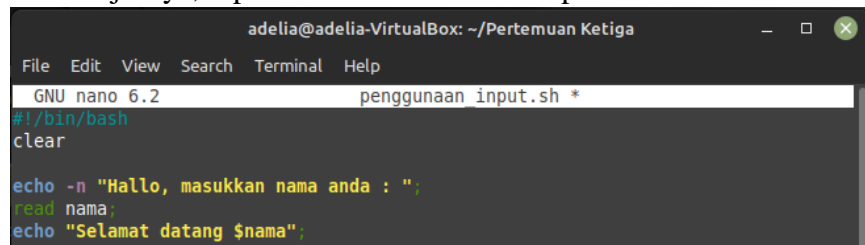
```

adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano penggunaan_operasimtk.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_operasimtk.sh
a + b = 22
a - b = 8
a * b = 105
a / b = 2
a % b = 1
a = 15
b = 15

```

5. Penggunaan Input

Dalam bahasa pemrograman bash juga dapat memasukkan input nilai. Agar inputan tersebut dapat terbaca, maka menggunakan perintah **read** dengan format penulisan **read nama_variabel**. Selanjutnya, inputan tersebut akan disimpan dalam nilai variabel.



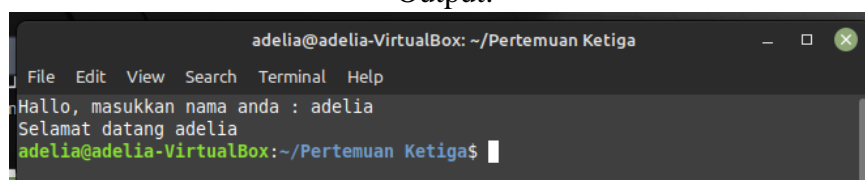
```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2      penggunaan_input.sh *
#!/bin/bash
clear

echo -n "Hallo, masukkan nama anda : ";
read nama;
echo "Selamat datang $nama";

```

Output:



```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
Hallo, masukkan nama anda : adelia
Selamat datang adelia
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$

```

6. Penggunaan Output

Echo digunakan untuk menampilkan output dengan syntax **echo "teks"** untuk menampilkan sebuah teks dan **echo \$nama_variabel** untuk menampilkan nilai dari sebuah variabel.

```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2      penggunaan_output.sh *
#!/bin/bash
matakuliah="Sistem Operasi"

echo "Siapa namamu?"
read nama
echo -e "\nHai $nama!\nSelamat datang di praktikum $matakuliah:)))))"

```

Output:

```

adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano penggunaan_output.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_output.sh
Siapa namamu?
Adelia

Hai Adelia!
Selamat datang di praktikum Sistem Operasi:)))))

```

Selain menggunakan **echo**, output juga dapat ditampilkan dengan menggunakan **printf** seperti pada bahasa pemrograman C.

```

adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2      penggunaan_output2.sh *
#!/bin/bash

# Inisialisasi var
a=55;
b=4;
distroLinux="Ubuntu 19.04 LTS";
let c=a*b;

# Output Printf
printf "OS : $distroLinux \n";
printf "$c \n";
printf "%.2f float \n" $a;
printf "%.1f float \n" $a;

```

Output :

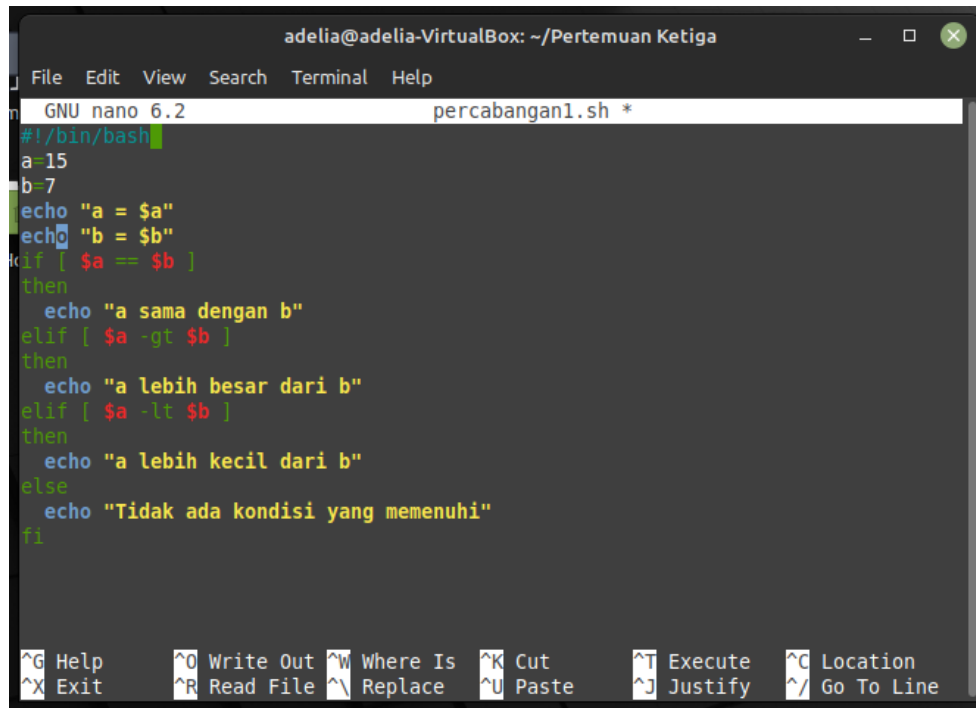
```

adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano penggunaan_output2.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash penggunaan_output2.sh
OS : Ubuntu 19.04 LTS
3
55.00 float
55.0 float

```

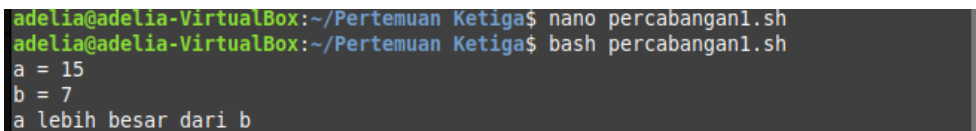
7. Percabangan

Pada script percabangan **if then** di sini untuk membandingkan nilai dari variabel a dan variabel b. Di mana ketika **a==b** akan mencetak “a sama dengan b”, ketika **a -gt b** akan mencetak “a lebih besar dari b”, ketika **a -lt b** mencetak “a lebih kecil dari b”, dan apabila tidak memenuhi ketiganya akan mencetak “Tidak ada kondisi yang memenuhi”. **fi** digunakan untuk menutup if then.



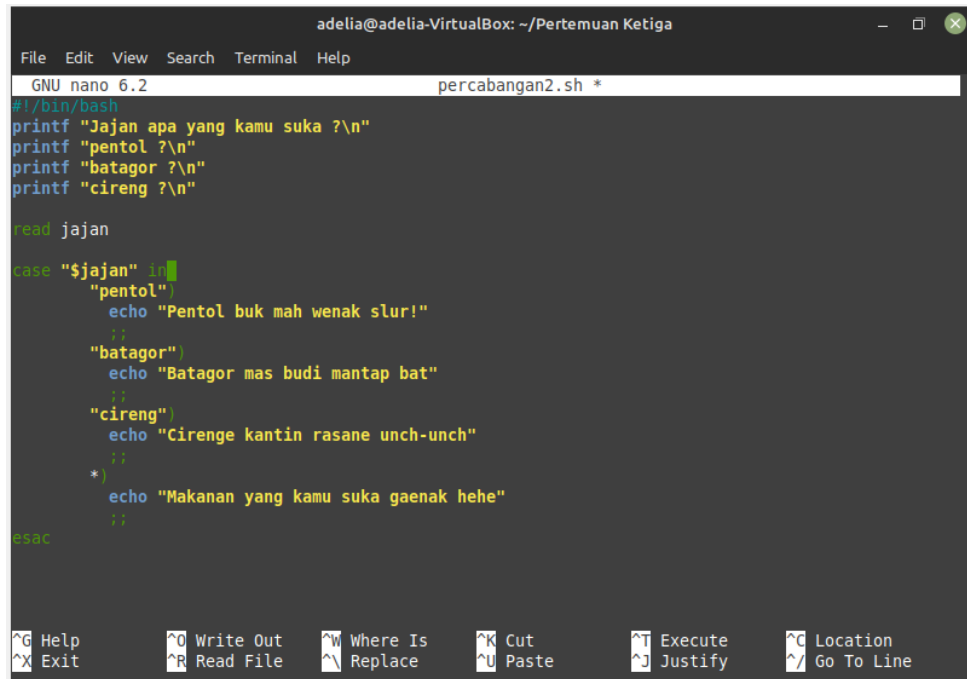
```
adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
File Edit View Search Terminal Help
GNU nano 6.2 percabangan1.sh *
#!/bin/bash
a=15
b=7
echo "a = $a"
echo "b = $b"
if [ $a == $b ]
then
    echo "a sama dengan b"
elif [ $a -gt $b ]
then
    echo "a lebih besar dari b"
elif [ $a -lt $b ]
then
    echo "a lebih kecil dari b"
else
    echo "Tidak ada kondisi yang memenuhi"
fi
```

Output:



```
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano percabangan1.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash percabangan1.sh
a = 15
b = 7
a lebih besar dari b
```

Selanjutnya pada script kedua menggunakan perintah case. Perintah ini digunakan untuk menyederhanakan pemakaian if yang berantai atau kompleks, sehingga dengan menggunakan case, kondisi dapat dikelompokkan secara logis dan lebih mudah juga jelas dalam penulisannya. Setiap intruksi diakhiri dengan tanda **;;**. Dan tanda ***)** mewakili intruksi yang dijalankan apabila seluruh kondisi tidak ada yang bernilai true. Percabangan ini diakhiri dengan **esac**.



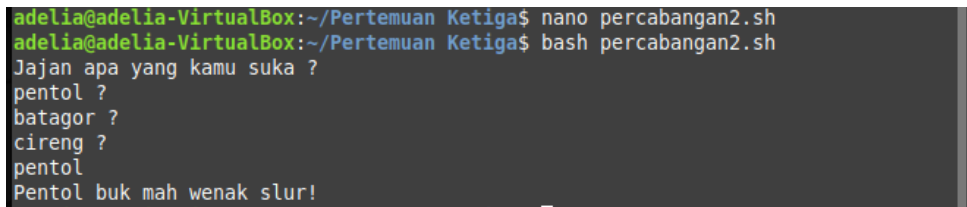
```
adelia@adelia-VirtualBox: ~/Pertemuan Ketiga
GNU nano 6.2 percabangan2.sh *
#!/bin/bash
printf "Jajan apa yang kamu suka ?\n"
printf "pentol ?\n"
printf "batagor ?\n"
printf "cireng ?\n"

read jajan

case "$jajan" in
    "pentol")
        echo "Pentol buk mah wenak slur!"
        ;;
    "batagor")
        echo "Batagor mas budi mantap bat"
        ;;
    "cireng")
        echo "Cireng kantin rasane unch-unch"
        ;;
    *)
        echo "Makanan yang kamu suka gaenak hehe"
        ;;
esac

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Output:



```
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano percabangan2.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash percabangan2.sh
Jajan apa yang kamu suka ?
pentol ?
batagor ?
cireng ?
pentol
Pentol buk mah wenak slur!
```

LATIHAN SOAL

Buatlah program percabangan sederhana aritmatika yang menerapkan beberapa konsep pemrograman bash seperti di atas!

Saya membuat program menggunakan bahasa pemrograman bash yang memuat string, array, perintah random array, integer, output, input, operasi bilangan, dan tak lupa juga percabangan. Saya mendeskripsikan judul sebagai string, kemudian operasi bilangan sebagai array agar nanti dapat memilih operasi bilangan secara acak menurut system. Selanjutnya menggunakan perintah input-an untuk nilai variabel a dan output-an untuk nilai variabel b.

Kemudian menentukan nilai dari variabel c dan d dengan mengoperasikan nilai dari variabel a dan b sedemikian rupa. Selanjutnya mengacak nilai array operasi dan menyimpannya pada variabel operasiterpilih. Nilai dari array operasi tersebut akan menentukan perintah percabangan. Ketika operasiterpilih menunjukkan “tambah” maka akan mencetak hasil dari penjumlahan variabel c dan d, apabila menunjukkan “kali” maka akan mencetak hasil dari perkalian variabel c dan d, apabila menunjukkan “kurang” maka akan mencetak hasil dari pengurangan variabel c dan d, apabila menunjukkan “bagi” maka akan mencetak hasil dari pembagian c dan d. Lalu apabila tidak dapat menunjukkan keempat nilai tersebut, akan mencetak “hayolo salah”. Saya sering kali menemukan eror ketika membuat program ini karena kesalahan penulisan yang sederhana, misalnya ketika lupa menuliskan \$, bagi saya hal yang paling sulit adalah ketika menentukan random nilai dari array. Berikut ialah script beserta dokumentasinya :

```
#string
judul='Mari kita mengoperasikan bilangan!'
echo "$judul"
#array
operasi=('tambah' 'kali' 'kurang' 'bagi')
let pilih=$RANDOM%5
#input
echo -n "masukkan angka yang anda suka: "
read a
echo "a=$a"
#output
echo "masukkan lagi angka yang anda suka: "
read b
echo "b=$b"
#operasi
c=$((a+b*4))
d=$((a*2/b))
echo "c = a+b*4, c = $c"
echo "d = a^2/b, d = $d"
#random array
operasiterpilih=${operasi[$pilih]}
echo "operasi yang dipilih: $operasiterpilih"
```

```

case $operasiterpilih in
    "tambah")
        echo "$c + $d = [$c+$d]"
        ;;
    "kali")
        echo "$c * $d = [$c*$d]"
        ;;
    "kurang")
        echo "$c - $d = [$c-$d]"
        ;;
    "bagi")
        echo "$c / $d = [$c/$d]"
        ;;
    *)
        echo "hayolo salah"
        ;;
esac

```

```

#string
judul='Mari kita mengoperasikan bilangan!'
echo "$judul"
#array
operasi=('tambah' 'kali' 'kurang' 'bagi')
let pilih=$RANDOM%5
#input
echo -n "masukkan angka yang anda suka: "
read a
echo "a=$a"
#output
echo "masukkan lagi angka yang anda suka: "
read b
echo "b=$b"
#operasi
c=$((a+b*4))
d=$((a*2/b))
echo "c = a+b*4, c = $c"
echo "d = a^2/b, d = $d"
#random array
operasiterpilih=${operasi[$pilih]}
echo "operasi yang dipilih: $operasiterpilih"

```

```

case $operasiterpilih in
    "tambah")
        echo "$c + $d = [$c+$d]"
        ;;
    "kali")
        echo "$c * $d = [$c*$d]"
        ;;
    "kurang")
        echo "$c - $d = [$c-$d]"
        ;;
    "bagi")
        echo "$c / $d = [$c/$d]"
        ;;
    *)
        echo "hayolo salah"
        ;;
esac

```


Output:

```
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ nano jawabansoal.sh
adelia@adelia-VirtualBox:~/Pertemuan Ketiga$ bash jawabansoal.sh
Mari kita mengoperasikan bilangan!
masukkan angka yang anda suka: 76
a=76
masukkan lagi angka yang anda suka:
95
b=95
c = a+b*4, c = 456
d = a^2/b, d = 1
operasi yang dipilih: tambah
456 + 1 = 457
```