LAPORAN RESMI

PRAKTIKUM SISTEM OPERASI

**ULANGAN AHKIR SEMESTER**

**PROGRAM SHELL**



**NAMA : MUHAMMAD RIZAL RAZAAN**

**NIM : 21104410061**

**KELAS : TI 4B**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ISLAM BALITAR**

**2023**

**LEMBAR PERSETUJUAN**

**ULANGAN TENGAH SEMESTER**

**PROGRAM SHELL**

**JURUSAN : TEKNIK INFORMATIKA**

**NAMA : MUHAMMAD RIZAL RAZAAN**

**KELAS : TI - B 2021**

**TGL. PERTEMUAN : 20 Juli 2023**

**Disetujui**

**Blitar, 23 Juli 2023**

**Dosen Pengampu**

**Mohammad Faried Rahmat, S.ST., M.Tr.T**

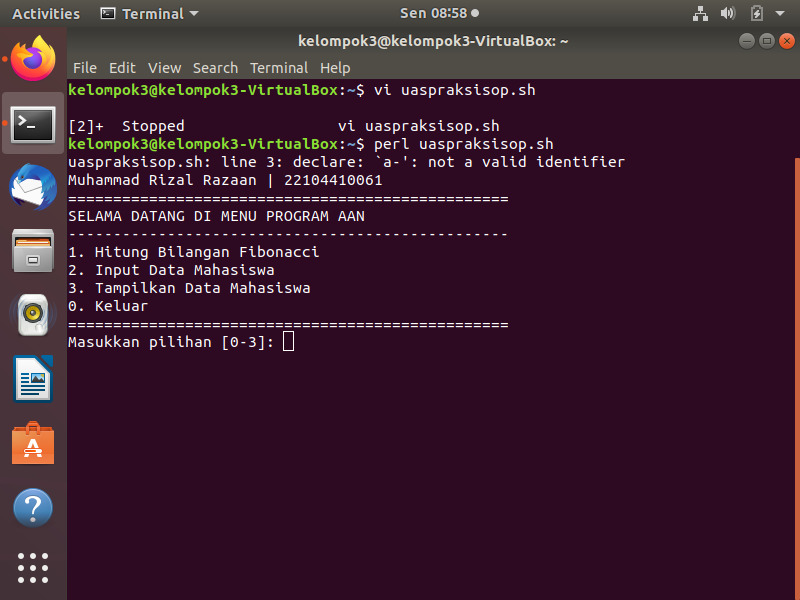
**NIDN.**

1. **Screenshot dan Penjelasan**
   1. **Tampialn untuk menjalakan program**

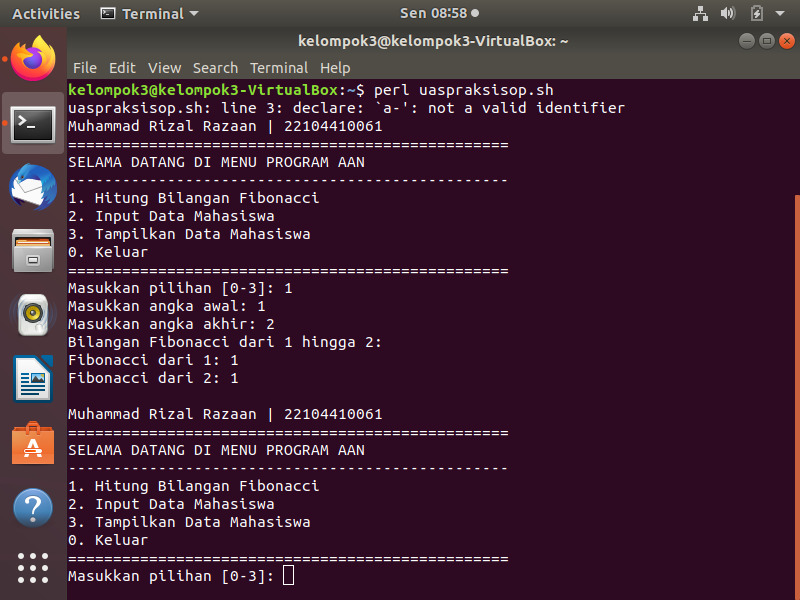
****

$perl uaspraksisop.sh digunakan untuk menjalakan prgram dari file uaspraksisop.sh

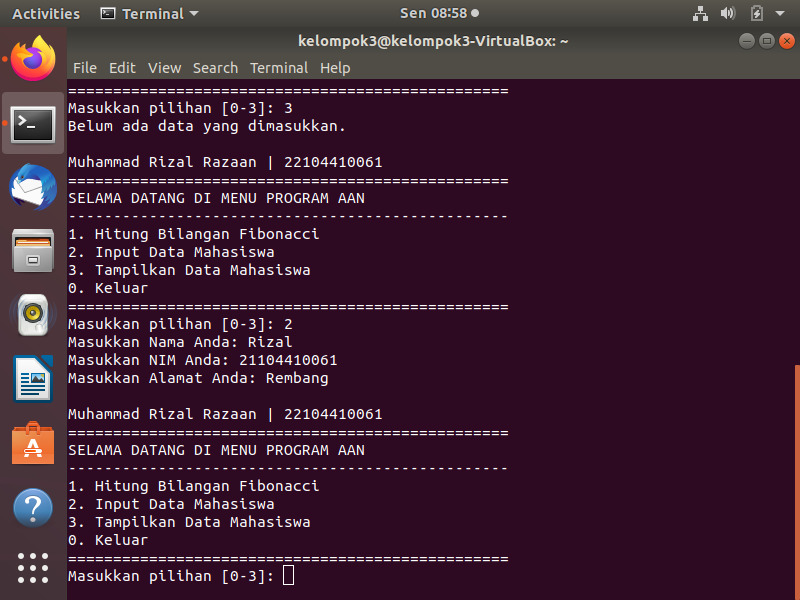
* 1. **Tampilan menu**



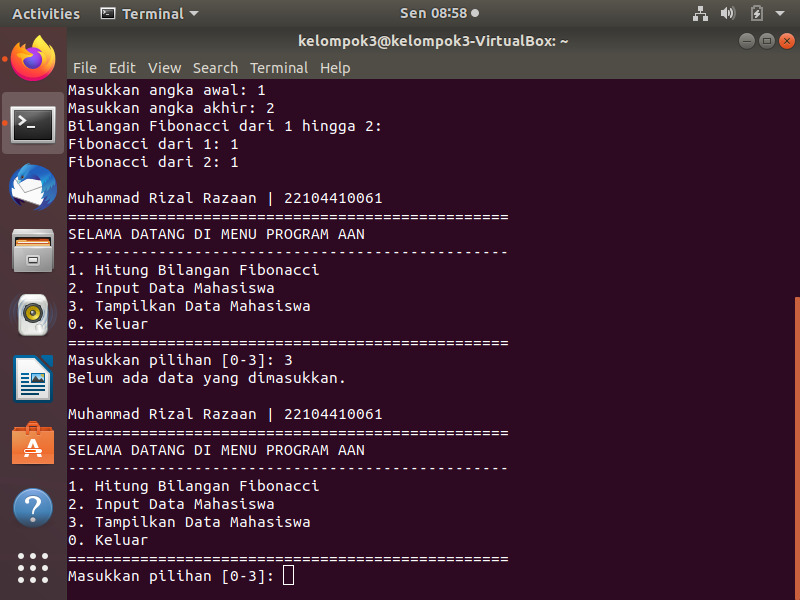
* Penjelasan : Setelah menjalakan program uaspraksisop.sh menucul menu seperti gambar diatas dengan nama dan nim pada bagian atas. Lalu di bawahnya ada tampilan selamat datang ke dalam menu program dengan program no.1 adalah menghitung bilangan finonanci no 2. menginputkan data mahasiswa ke dalam array no.3 untuk menampilkan data mahasiswa yang ada di dalam array.
  1. **Tampilan pogram perhitungan bilangan fibonanci**



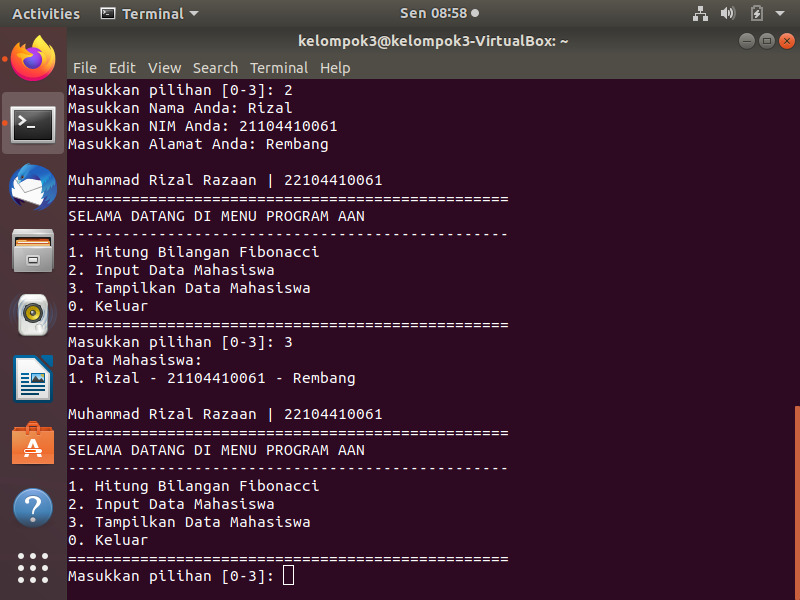
* Penjelasan : Pada program menghitung fibonanci kita memasukkan rentang data yang akan dihutung. Saya memasukkan angka 1 untuk awal dan angka 2 untuk ahkir. Maka output yang keluar ada 1 dan 1 karena 1 dan 2 angka sebelumnya ada 0 + 1 jadi outputnya 1 dan 1.
  1. **Tampilan input data mahasiswa**



* Penjelasan : Pada program ke 2 mengiputkan data mahasiswa kita akan memasukkan nama, nim , dan alamat kita yang akan tersimpan ke dalam array yang akan dipanggil di program ke 3. Pada contoh yang saya berikan adalah memasukkan nama Rizal dengan NIM 21104410061 dan beralamat di Rembang maka nanti pada program ke 3 akan muncul data tersebut.
  1. **Tampilan display data mahasiswa yang belum terisi**

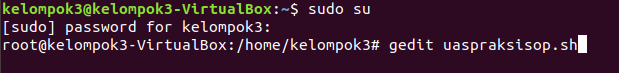


* Penjelasan : Pada tampilan di atas adalah contoh ketika pada program ke 2 tidak saya masukkan dan langsung memanggil program ke 3
  1. **Tampilan display data mahasiswa ketikda sudah di input data**



* Penjelasan : Tampailan program ke 3 adalah hasil dari input program ke 2 yang telah saya masukkan nama, NIM , dan alamat

1. **Deskripsi pembuka awal ceritakan fitur UAS Pemograman shell**
   1. **Membuat uaspraksisop.sh**

****

Pertama kita masuk kedalam root dengan mengetikkan "sudo su" dan memasukkan password. Kemudian setelah masuk ke dalam root ketik "gedit uaspraksisop.sh" untuk masuk ke dalam direcotory dan membuat file sh yang akan kita buat untuk menjalakan program.

* 1. **Mengisi uaspraksisop.sh dengan script**

Script

|  |
| --- |
| #!/bin/bash  declare a- mahasiswa  # Function untuk menghitung bilangan Fibonacci  function fibonacci() {  local num=$1  if [ $num -le 0 ]; then  echo "0"  elif [ $num -eq 1 ]; then  echo "1"  else  local a=0  local b=1  local fib=1  for ((i = 2; i <= num; i++)); do  fib=$((a + b))  a=$b  b=$fib  done  echo "$fib"  fi  }  # Function untuk input data mahasiswa  function input\_data() {  read -p "Masukkan Nama Anda: " name  read -p "Masukkan NIM Anda: " nim  read -p "Masukkan Alamat Anda: " address  # Menyimpan data dalam array students  students+=("$name - $nim - $address")  }  # Function untuk menampilkan data mahasiswa  function display\_data() {  if [ ${#students[@]} -eq 0 ]; then  echo "Belum ada data yang dimasukkan."  else  echo "Data Mahasiswa:"  for ((i = 0; i < ${#students[@]}; i++)); do  echo "$(($i + 1)). ${students[$i]}"  done  fi  }  # Function untuk menampilkan menu utama  function show\_menu() {  echo "Muhammad Rizal Razaan | 22104410061 "  echo "================================================="  echo "SELAMA DATANG DI MENU PROGRAM AAN"  echo "-------------------------------------------------"  echo "1. Hitung Bilangan Fibonacci "  echo "2. Input Data Mahasiswa"  echo "3. Tampilkan Data Mahasiswa"  echo "0. Keluar"  echo "================================================="  read -p "Masukkan pilihan [0-3]: " choice  }  # Main loop  while true; do  show\_menu  case $choice in  1)  # Pilihan 1: Hitung Bilangan Fibonacci  read -p "Masukkan angka awal: " start  read -p "Masukkan angka akhir: " end  # Validasi input  if [ $start -gt $end ]; then  echo "Angka awal harus lebih kecil dari angka akhir."  continue  fi  echo "Bilangan Fibonacci dari $start hingga $end:"  for ((i = start; i <= end; i++)); do  result=$(fibonacci $i)  echo "Fibonacci dari $i: $result"  done  ;;  2)  # Pilihan 2: Input Data Mahasiswa  input\_data  ;;  3)  # Pilihan 3: Tampilkan Data Mahasiswa  display\_data  ;;  0)  # Pilihan 0: Keluar  echo "Terima kasih, Selamat menggunakan kembali"  exit  ;;  \*)  echo "Pilihan anda salah atau bukan dengan nomer yang sudah tertera. Silakan masukkan angka dari 0 hingga 3."  ;;  esac  echo  done |

* 1. **Penjelasan Script**
* #!/bin/bash ini adalah shebang, yang memberitahu sistem bahwa skrip ini harus dijalankan menggunakan shell Bash.
* declare -a students ini mendeklarasikan variabel students sebagai array. Nama variabel students akan digunakan untuk menyimpan data mahasiswa.
* function fibonacci() { ... } ini mendefinisikan fungsi fibonacci() yang akan menghitung bilangan Fibonacci berdasarkan angka yang diberikan.
* local num=$1 ini mendeklarasikan variabel lokal num dan menginisialisasinya dengan nilai argumen pertama dari fungsi (argumen yang dilewatkan ketika fungsi dipanggil).
* if [ $num -le 0 ]; then ... elif [ $num -eq 1 ]; then ... else ... fi ini adalah pernyataan pengkondisian. Program akan memeriksa apakah num lebih kecil dari atau sama dengan 0, atau apakah num sama dengan 1, atau apakah tidak ada kondisi yang terpenuhi. Program akan menghitung bilangan Fibonacci sesuai kondisi yang terpenuhi.
* for ((i = 2; i <= num; i++)); do ... done ini adalah pernyataan loop for yang akan menjalankan perintah yang berada di dalamnya selama nilai i mulai dari 2 hingga num. Loop ini digunakan untuk menghitung bilangan Fibonacci.
* function input\_data() { ... } ini mendefinisikan fungsi input\_data() untuk mengambil input data mahasiswa seperti nama, NIM, dan alamat, dan menyimpannya dalam array students.
* function display\_data() { ... } ini mendefinisikan fungsi display\_data() untuk menampilkan data mahasiswa yang telah diinput sebelumnya dari array students.
* function show\_menu() { ... } ini mendefinisikan fungsi show\_menu() untuk menampilkan menu utama dengan beberapa pilihan.
* while true; do ... done ini adalah perulangan utama program yang akan berjalan terus menerus sampai program dihentikan. Pada setiap iterasi, program akan menampilkan menu utama dan memproses pilihan yang diambil oleh pengguna.
* case $choice in ... esac ini adalah pernyataan case yang akan memeriksa nilai variabel choice dan melakukan aksi sesuai dengan pilihan yang dipilih oleh pengguna dalam menu utama.
* read -p "Masukkan pilihan [0-3]: " choice ini adalah perintah untuk membaca input dari pengguna dan menyimpannya dalam variabel choice.
  1. **Fitur sytax yang saya pakai dalam program**
* declare: Ini adalah perintah dalam Bash untuk mendeklarasikan variabel dengan tipe tertentu. Misalnya, declare -a nama\_array akan mendeklarasikan variabel nama\_array sebagai array.
* function: Ini adalah kata kunci yang digunakan untuk mendefinisikan sebuah fungsi di Bash. Contoh penggunaan: function nama\_fungsi() { ... }
* local: Ini adalah kata kunci yang digunakan dalam Bash untuk mendeklarasikan variabel sebagai lokal, yang berarti variabel tersebut hanya akan berlaku di dalam fungsi tempat variabel tersebut dideklarasikan.
* num: Ini adalah nama variabel yang dapat diubah sesuai dengan kebutuhan program. Variabel ini dapat digunakan untuk menyimpan nilai angka atau bilangan.
* if: Ini adalah pernyataan kondisional dalam Bash yang digunakan untuk melakukan pemilihan berdasarkan kondisi tertentu.
* elif: Ini adalah kependekan dari "else if", dan digunakan dalam pernyataan kondisional untuk mengevaluasi kondisi lain jika kondisi sebelumnya tidak terpenuhi.
* echo: Ini adalah perintah yang digunakan untuk mencetak atau menampilkan teks ke layar.
* then: Ini adalah kata kunci yang digunakan dalam pernyataan kondisional untuk menandakan awal dari blok perintah yang akan dijalankan jika kondisi terpenuhi.
* for: Ini adalah kata kunci yang digunakan untuk menginisialisasi loop for dalam Bash.
* done: Ini adalah kata kunci yang digunakan untuk menandakan akhir dari blok perintah dalam loop for.
* fi: Ini adalah kependekan dari "finish", dan digunakan dalam pernyataan kondisional untuk menandakan akhir dari blok perintah dalam pernyataan if.
* read: Ini adalah perintah yang digunakan untuk membaca masukan dari pengguna dalam terminal.
* -p: Ini adalah argumen yang digunakan dengan perintah read untuk menampilkan prompt atau pesan sebelum meminta masukan dari pengguna.
* eq: Ini adalah operator dalam pernyataan kondisional untuk memeriksa kesetaraan antara dua nilai.
* main loop: Ini bukan perintah atau kata kunci, tetapi hanya sebuah keterangan yang digunakan untuk menjelaskan loop utama dalam program. Loop ini akan berjalan terus menerus selama kondisi yang diberikan benar.
* while true: Ini adalah loop while yang berjalan selama kondisi yang diberikan benar. Dalam contoh ini, loop akan berjalan selamanya karena kondisinya selalu benar (true).
* do: Ini adalah kata kunci yang digunakan untuk menandakan awal dari blok perintah yang akan dijalankan dalam loop while atau for.
* case: Ini adalah kata kunci yang digunakan dalam struktur case untuk mengevaluasi nilai dan memilih blok perintah yang sesuai berdasarkan nilai tersebut.
* choice in: Ini adalah bagian dari struktur case yang menentukan variabel choice sebagai variabel yang akan dievaluasi dalam struktur case.
* exit: Ini adalah perintah untuk mengakhiri program atau keluar dari loop atau fungsi
  1. **Cara menjalakan program**

****

* 1. **Fitur yang saya masukkan**
* **Program menghitung bilangan fibonanci**

Fibonacci Sequence (Deret angka Fibonacci) adalah deret angka yang diperoleh dengan menjumlahkan dua angka sebelumnya

* **Program menginput data mahasiswa**

Input program adalah bagian dari sebuah program komputer yang bertanggung jawab untuk menerima input dari pengguna atau sistem lain. Selain itu, input program dapat berupa apa saja, termasuk teks, angka, gambar, atau data lain yang dibutuhkan oleh program untuk melakukan tugas yang diinginkan.

* **Program menampilkan data mahasiswa yang telah di input**

Ouptut program Perintah adalah perintah yang berfungsi untuk menampilkan pernyataan sehingga muncul ke layar / hasil consol program.