

# **Laporan Sistem Operasi Ulangan Akhir Semester**



**Nama : Luthfi Triaswangga**  
**Kelas : 1B / 17**  
**NIM : 2341720208**

**Teknologi Informasi  
Teknik Informatika  
Politeknik Negeri Malang  
2024**

# **Bab I**

## **Pendahuluan**

### **A. Latar Belakang**

Dalam era digital saat ini, efisiensi dan kemudahan dalam pengelolaan sistem operasi sangat penting bagi para pengguna komputer, baik itu pemula maupun profesional. Sistem operasi Linux, khususnya distribusi Ubuntu, sering digunakan karena kestabilan dan fleksibilitasnya. Namun, tidak semua pengguna memiliki pemahaman yang mendalam tentang berbagai perintah dasar yang dapat digunakan untuk mengelola file dan direktori melalui terminal.

### **B. Tujuan Proyek**

Tujuan dari proyek ini adalah untuk membuat sebuah skrip shell yang dapat menjalankan berbagai operasi dasar pada sistem operasi Linux Ubuntu melalui menu interaktif. Skrip ini dirancang untuk mempermudah pengguna dalam:

1. Melihat isi direktori: Menyediakan cara cepat untuk menampilkan daftar file dan folder dalam direktori.
2. Melihat isi file: Membantu pengguna membaca isi file teks tanpa harus mengingat perintah cat.
3. Membuat file baru: Mempermudah pembuatan file baru dan membuka file tersebut untuk pengeditan.
4. Menyalin file: Memudahkan proses penyalinan file dari satu lokasi ke lokasi lain.
5. Menghapus file: Menyederhanakan penghapusan file dengan cara yang aman.
6. Mengganti nama file: Memfasilitasi penggantian nama file tanpa kesalahan.
7. Mengedit file: Membuka file untuk pengeditan menggunakan editor teks.
8. Memindahkan file ke folder: Mempermudah pemindahan file ke direktori yang diinginkan.
9. Membuat folder baru: Memungkinkan pengguna untuk membuat folder baru dengan mudah.
10. Menghapus folder: Menghapus folder beserta isinya secara sederhana.
11. Mengompres folder: Membuat arsip kompresi dari folder untuk menghemat ruang penyimpanan.
12. Mengekstrak folder: Mengekstrak arsip kompresi ke direktori tujuan.
13. Mengecek koneksi jaringan: Menguji koneksi jaringan dengan melakukan ping ke alamat tertentu.
14. Mengecek spesifikasi perangkat: Menampilkan informasi hardware dan sistem yang terperinci.

Dengan adanya skrip ini, diharapkan pengguna dapat lebih efisien dalam mengelola sistem operasinya dan mengurangi kesalahan yang mungkin terjadi akibat ketidaktahuan atau ketidakpahaman tentang perintah-perintah dasar Linux. Skrip ini juga dapat menjadi alat belajar yang efektif bagi pengguna baru yang ingin mengenal lebih dalam tentang administrasi sistem Linux.

## BAB II

### Deskripsi Program

```
=====
|               Luthfi Triaswangga               |
|               UAS Sistem Operasi               |
|               2341720208                       |
|=====|
|               Mon Jun  3 14:45:58 WIB 2024      |
|=====|
| Pilih operasi yang ingin Anda lakukan:         |
| 1. Lihat isi direktori                         |
| 2. Lihat isi file                             |
| 3. Create file                                |
| 4. Copy file                                  |
| 5. Delete file                                |
| 6. Rename file                               |
| 7. Edit file                                  |
| 8. Pindahkan file ke folder                   |
| 9. Create folder                             |
| 10. Delete folder                            |
| 11. Compress folder                          |
| 12. Extract folder                           |
| 13. Cek ping jaringan                        |
| 14. Cek spesifikasi perangkat                |
| 0. Keluar                                     |
|=====|
| Masukkan pilihan Anda: 1 - 14 / 0 |
```

#### 1. Lihat Isi Direktori

Pada case ini, perintah `ls` digunakan untuk menampilkan daftar isi dari direktori saat ini. Perintah `ls` sangat berguna untuk melihat semua file dan folder yang terdapat dalam direktori kerja saat ini. Setelah menampilkan isi direktori, skrip ini akan memanggil kembali skrip utama `./luthfiuas.sh` untuk menampilkan kembali menu utama kepada pengguna,

#### 2. Lihat Isi File

Di sini, pengguna diminta untuk memasukkan nama file yang ingin dilihat isinya. Skrip memeriksa apakah file tersebut ada menggunakan perintah `if [ -f "$nama_file" ]`. Jika file ditemukan, isinya ditampilkan dengan perintah `cat`. Jika file tidak ditemukan, pesan error ditampilkan. Setelah itu, skrip akan kembali ke menu utama.

#### 3. Create File

Fitur ini meminta pengguna untuk memasukkan nama file baru yang ingin dibuat. Skrip kemudian membuka editor nano dengan nama file yang dimasukkan, memungkinkan pengguna untuk mengedit dan menyimpan file baru tersebut. Setelah editor ditutup, skrip mengkonfirmasi bahwa file berhasil dibuat dan kembali ke menu utama.

#### **4. Copy File**

Pengguna diminta memasukkan nama file sumber yang ingin disalin dan nama file tujuan. Skrip memeriksa apakah file sumber ada, dan jika ada, menggunakan perintah cp untuk menyalin file tersebut ke nama file tujuan yang diberikan. Jika file sumber tidak ditemukan, pesan error akan ditampilkan.

#### **5. Delete File**

Pengguna memasukkan nama file yang ingin dihapus. Skrip memeriksa keberadaan file dengan if [ -f "\$nama\_file\_hapus" ], dan jika file ada, menggunakan perintah rm untuk menghapus file tersebut. Setelah file dihapus.

#### **6. Rename File**

Fitur ini meminta pengguna memasukkan nama file yang ingin diganti dan nama baru untuk file tersebut. Skrip memeriksa keberadaan file lama dan jika ditemukan, menggunakan perintah mv untuk mengganti nama file. Jika file lama tidak ditemukan, pesan error akan ditampilkan.

#### **7. Edit File**

Pengguna diminta memasukkan nama file yang ingin diedit. Skrip memeriksa apakah file tersebut ada dan jika ada, membuka editor nano untuk mengedit file tersebut. Setelah selesai, skrip mengkonfirmasi bahwa file berhasil diedit dan kembali ke menu utama.

#### **8. Pindahkan File ke Folder**

Pengguna memasukkan nama file yang ingin dipindahkan dan nama folder tujuan. Skrip memeriksa keberadaan file dan folder, kemudian menggunakan perintah mv untuk memindahkan file ke folder tujuan jika keduanya ada. Setelah file dipindahkan.

#### **9. Create Folder**

Pengguna diminta memasukkan nama folder baru yang ingin dibuat. Skrip menggunakan perintah mkdir -p untuk membuat folder tersebut, memastikan folder dibuat dengan struktur direktori yang diperlukan.

#### **10. Delete Folder**

Pengguna memasukkan nama folder yang ingin dihapus. Skrip memeriksa keberadaan folder dan jika ditemukan, menggunakan perintah rm -r untuk menghapus folder tersebut beserta isinya.

#### **11. Compress Folder**

Pengguna memasukkan nama folder yang ingin dikompres dan nama file arsip. Skrip memeriksa keberadaan folder dan jika ada, menggunakan perintah tar -czvf untuk mengkompres folder tersebut menjadi file arsip dengan nama yang diberikan.

#### **12. Extract Folder**

Pengguna diminta memasukkan nama file arsip dan direktori tujuan. Skrip memeriksa keberadaan file arsip dan jika ada, menggunakan perintah tar -xzvf untuk mengekstrak file arsip ke direktori tujuan. Jika file arsip tidak ditemukan, pesan error akan ditampilkan.

### **13. Cek Ping Jaringan**

Pengguna memasukkan alamat IP atau domain yang ingin di-ping. Skrip menjalankan perintah `ping -c 4` untuk mengirim empat paket ping ke alamat yang diberikan dan menampilkan hasilnya.

### **14. Cek Spesifikasi Perangkat**

Fitur ini menampilkan informasi sistem seperti hostname, sistem operasi, kernel, CPU, memori, dan disk. Skrip menggunakan berbagai perintah seperti `hostname`, `uname -o`, `uname -r`, `lscpu`, `free -h`, dan `df -h --total` untuk mengumpulkan dan menampilkan informasi tersebut.

### **15. Keluar**

Pada case ini, skrip mengucapkan terima kasih dan keluar dari eksekusi dengan menggunakan perintah `exit 0`. Ini mengakhiri skrip dan menutup program.

## BAB III

### Kode Pemrograman Shell

1. Lihat Isi Direktori

```
1)
ls
./luthfiuas.sh
;;
```

2. Lihat Isi File

```
2)
read -p "Masukkan nama file yang ingin dilihat: " nama_file
if [ -f "$nama_file" ]; then
    cat "$nama_file"
else
    echo "File tidak ditemukan."
fi
./luthfiuas.sh
;;
```

3. Create File

```
3)
read -p "Masukkan nama file yang ingin dibuat: " nama_file_baru
nano "$nama_file_baru"
echo "File berhasil dibuat."
./luthfiuas.sh
;;
```

4. Copy File

```
4)
read -p "Masukkan nama file yang ingin disalin: " nama_file_sumber
if [ -f "$nama_file_sumber" ]; then
    read -p "Masukkan nama file tujuan: " nama_file_tujuan
    cp "$nama_file_sumber" "$nama_file_tujuan"
    echo "File berhasil disalin."
else
    echo "File sumber tidak ditemukan."
fi
./luthfiuas.sh
;;
```

5. Delete File

```
5)
read -p "Masukkan nama file yang ingin dihapus: " nama_file_hapus
if [ -f "$nama_file_hapus" ]; then
    rm "$nama_file_hapus"
    echo "File berhasil dihapus."
else
    echo "File tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 6. Rename File

```
6)
''
read -p "Masukkan nama file yang ingin diganti: " nama_file_lama
if [ -f "$nama_file_lama" ]; then
    read -p "Masukkan nama file baru: " nama_file_baru
    mv "$nama_file_lama" "$nama_file_baru"
    echo "Nama file berhasil diganti."
else
    echo "File tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 7. Edit File

```
7)
''
read -p "Masukkan nama file yang ingin diedit: " nama_file_edit
if [ -f "$nama_file_edit" ]; then
    nano "$nama_file_edit"
    echo "File berhasil diedit."
else
    echo "File tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 8. Pindahkan File ke Folder

```
8)
''
read -p "Masukkan nama file yang ingin dipindahkan: " nama_file_pindah
if [ -f "$nama_file_pindah" ]; then
    read -p "Masukkan nama folder tujuan: " folder_tujuan
    if [ -d "$folder_tujuan" ]; then
        mv "$nama_file_pindah" "$folder_tujuan"
        echo "File berhasil dipindahkan ke folder $folder_tujuan."
    else
        echo "Folder tujuan tidak ditemukan."
    fi
else
    echo "File tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 9. Create Folder

```
9)
''
read -p "Masukkan nama folder yang ingin dibuat: " nama_folder_baru
mkdir -p "$nama_folder_baru"
echo "Folder berhasil dibuat."
./luthfiuas.sh
;;
```

## 10. Delete Folder

```
10)
read -p "Masukkan nama folder yang ingin dihapus: " nama_folder_hapus
if [ -d "$nama_folder_hapus" ]; then
    rm -r "$nama_folder_hapus"
    echo "Folder berhasil dihapus."
else
    echo "Folder tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 11. Compress Folder

```
11)
read -p "Masukkan nama folder yang ingin dikompres: " nama_folder
read -p "Masukkan nama file arsip (misal: arsip.tar.gz): " nama_arsip
if [ -d "$nama_folder" ]; then
    tar -czvf "$nama_arsip" "$nama_folder"
    echo "Folder berhasil dikompres menjadi $nama_arsip."
else
    echo "Folder tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 12. Extract Folder

```
12)
read -p "Masukkan nama file arsip yang ingin diekstrak: " nama_arsip
read -p "Masukkan direktori tujuan: " direktori_tujuan
if [ -f "$nama_arsip" ]; then
    mkdir -p "$direktori_tujuan"
    tar -xzf "$nama_arsip" -C "$direktori_tujuan"
    echo "File arsip berhasil diekstrak ke $direktori_tujuan."
else
    echo "File arsip tidak ditemukan."
fi
./luthfiuas.sh
;;
```

## 13. Cek Ping Jaringan

```
13)
read -p "Masukkan alamat IP atau domain yang ingin di-ping: " target_ping
ping -c 4 "$target_ping"
./luthfiuas.sh
;;
```



#### 14. Cek Spesifikasi Perangkat

```
14)
    echo "Informasi Sistem:"
    echo "====="
    echo "Hostname: $(hostname)"
    echo "OS: $(uname -o)"
    echo "Kernel: $(uname -r)"
    echo "CPU: $(lscpu | grep 'Model name' | awk -F: '{print $2}')"
    echo "Memori: $(free -h | grep 'Mem:' | awk '{print $2}')"
    echo "Disk: $(df -h --total | grep 'total' | awk '{print $2}')"
    echo "====="
    ./luthfiuas.sh
;;
```

#### 15. Keluar

```
0)
    echo "Terima kasih telah menggunakan skrip ini."
    exit 0
;;
*)
    echo "Pilihan tidak valid."
    ./luthfiuas.sh
;;
```

## BAB IV

### Output Program

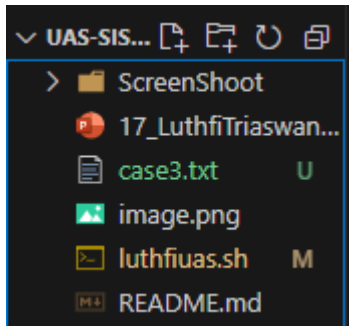
1. Lihat Isi Direktori

```
Masukkan pilihan Anda: 1
17_LuthfiTriaswangga_UAS.pptx  README.md  ScreenShoot  image.png  luthfiuas.sh
```

2. Lihat Isi File

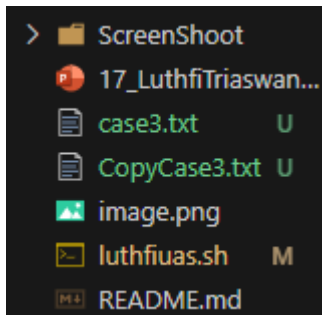
```
Masukkan pilihan Anda: 2
Masukkan nama file yang ingin dilihat: case3.txt
Ini adalah switch case 3 by Luthfi Triaswangga
```

3. Create File



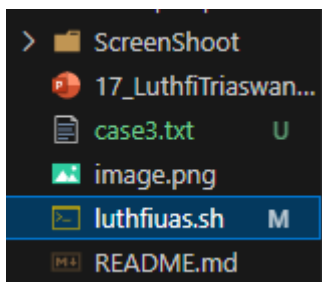
```
Masukkan pilihan Anda: 3
Masukkan nama file yang ingin dibuat: case3.txt
File berhasil dibuat.
```

4. Copy File



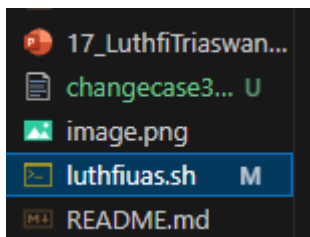
```
Masukkan pilihan Anda: 4
Masukkan nama file yang ingin disalin: case3.txt
Masukkan nama file tujuan: CopyCase3.txt
File berhasil disalin.
```

5. Delete File



```
Masukkan pilihan Anda: 5
Masukkan nama file yang ingin dihapus: CopyCase3.txt
File berhasil dihapus.
```

6. Rename File



```
Masukkan pilihan Anda: 6
Masukkan nama file yang ingin diganti: case3.txt
Masukkan nama file baru: changecase3.txt
Nama file berhasil diganti.
```

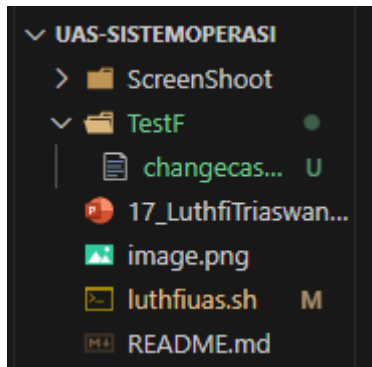
## 7. Edit File

```
Masukkan pilihan Anda: 7
Masukkan nama file yang ingin diedit: changecase3.txt
File berhasil diedit.
```

```
Masukkan pilihan Anda: 2
Masukkan nama file yang ingin dilihat: changecase3.txt
Ini adalah switch case 3 by Luthfi Triaswangga

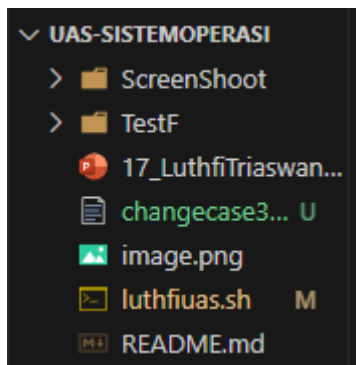
Edit :
2341720208
TI - 1B / 17
```

## 8. Pindahkan File ke Folder



```
Masukkan pilihan Anda: 8
Masukkan nama file yang ingin dipindahkan: changecase3.txt
Masukkan nama folder tujuan: TestF
File berhasil dipindahkan ke folder TestF.
```

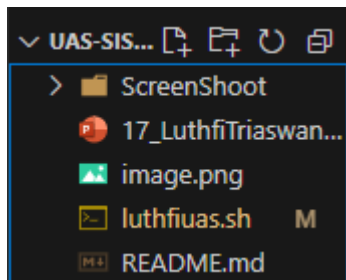
## 9. Create Folder



```
Masukkan pilihan Anda: 9
Masukkan nama folder yang ingin dibuat: TestF
Folder berhasil dibuat.
```

## 10. Delete Folder

```
Masukkan pilihan Anda: 10
Masukkan nama folder yang ingin dihapus: TestF
Folder berhasil dihapus.
```

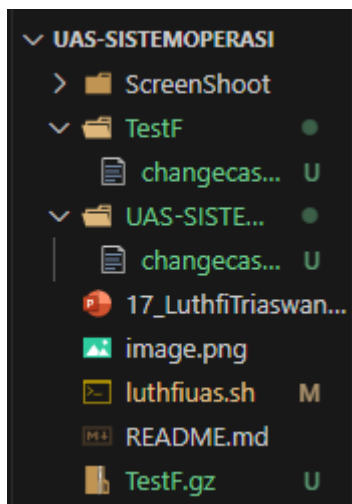


## 11. Compress Folder

```
Masukkan pilihan Anda: 11
Masukkan nama folder yang ingin dikompres: TestF
Masukkan nama file arsip (misal: arsip.tar.gz): TestF.gz
TestF/
TestF/changeCase3.txt
Folder berhasil dikompres menjadi TestF.gz.
```

## 12. Extract Folder

```
Masukkan pilihan Anda: 12
Masukkan nama file arsip yang ingin diekstrak: TestF.gz
Masukkan direktori tujuan: UAS-SISTEMOPERASI
TestF/
TestF/changeCase3.txt
File arsip berhasil diekstrak ke UAS-SISTEMOPERASI.
```



### 13. Cek Ping Jaringan

```
Masukkan pilihan Anda: 13
Masukkan alamat IP atau domain yang ingin di-ping: discord.com
PING discord.com (162.159.137.232) 56(84) bytes of data.
64 bytes from 162.159.137.232 (162.159.137.232): icmp_seq=1 ttl=52 time=60.9 ms
64 bytes from 162.159.137.232 (162.159.137.232): icmp_seq=2 ttl=52 time=58.5 ms
64 bytes from 162.159.137.232 (162.159.137.232): icmp_seq=3 ttl=52 time=32.4 ms
64 bytes from 162.159.137.232: icmp_seq=4 ttl=52 time=47.6 ms

--- discord.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3350ms
rtt min/avg/max/mdev = 32.434/49.861/60.939/11.238 ms
```

### 14. Cek Spesifikasi Perangkat

```
Masukkan pilihan Anda: 14
Informasi Sistem:
=====
Hostname: DESKTOP-FSV34RJ
OS: GNU/Linux
Kernel: 5.15.146.1-microsoft-standard-WSL2
CPU: Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz
Memori: 7.7Gi
Disk: 1.3T
=====
```

### 15. Keluar

```
Masukkan pilihan Anda: 0
Terima kasih telah menggunakan skrip ini.
upiwijen@DESKTOP-FSV34RJ:/mnt/d/Kuliah/Belajar/Semester 2/SISOT/UAS-SistemOperasi/UAS-SISTEMOPERASI$
```

## **BAB V**

### **Penutup**

#### **A. Kesimpulan**

Pembuatan skrip shell interaktif untuk sistem operasi Linux Ubuntu ini berhasil mencapai tujuannya dalam menyederhanakan berbagai operasi dasar yang sering dilakukan oleh pengguna. Melalui menu interaktif yang mudah digunakan, skrip ini memungkinkan pengguna untuk menjalankan berbagai tugas administratif tanpa perlu menghafal dan mengetik perintah-perintah yang mungkin rumit dan membingungkan.

#### **B. Saran**

Dengan terus mengembangkan dan menyempurnakan skrip ini, diharapkan dapat menjadi alat yang semakin berguna dan dapat diandalkan bagi pengguna dalam mengelola sistem operasi Linux mereka dengan lebih efektif dan efisien.

- a. **Pengembangan Fitur Tambahan:** Skrip ini dapat terus dikembangkan dengan menambahkan fitur-fitur lain yang lebih kompleks, seperti manajemen pengguna dan grup, pengaturan izin file dan direktori, serta monitoring sistem.
- b. **Antarmuka Pengguna yang Lebih Baik:** Meskipun saat ini skrip menggunakan antarmuka berbasis teks, pengembangan ke arah antarmuka grafis sederhana (GUI) dapat lebih memudahkan pengguna yang kurang terbiasa dengan lingkungan command-line.
- c. **Peningkatan Keamanan:** Menambahkan lapisan keamanan seperti konfirmasi tambahan untuk operasi yang berpotensi merusak (seperti penghapusan file dan folder) dapat mencegah kesalahan yang tidak disengaja.
- d. **Dokumentasi dan Bantuan:** Menyediakan dokumentasi yang lebih mendetail serta fitur bantuan di dalam skrip dapat sangat membantu pengguna baru dalam memahami fungsi dan penggunaan setiap fitur yang tersedia.
- e. **Lokalisasi Bahasa:** Menyediakan dukungan untuk multiple languages dalam antarmuka dan pesan skrip akan membuatnya lebih mudah diakses oleh pengguna dari berbagai latar belakang bahasa.
- f. **Automasi Tugas:** Menambahkan kemampuan untuk mengotomatisasi beberapa tugas rutin melalui skrip dapat lebih meningkatkan efisiensi, seperti backup otomatis, pembaruan sistem, dan pemantauan log.