|  |
| --- |
| Name: Phạm Đức Thể  ID: 19522253  Class: IT007.M14.2 |

OPERATING SYSTEM  
LAB 2 REPORT

**SUMMARY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | | **Status** | **Page** |
| Section 2.5 | Ex 1 | Hoàn thành | 2 |
| Ex 2 | Hoàn thành | 6 |
| Ex 3 | Hoàn thành | 7 |
| Ex 4 | Hoàn thành | 8 |
|  |  |  |
|  |  |  |
|  |  |  |

**Self-scrores: 10/10**

*\*Note: Export file to* ***PDF*** *and name the file by following format:* ***LAB X – <Student ID>.pdf***

# Section 2.5

## Task name 1: Chạy tất cả các đoạn lệnh ví dụ ở phần 2.4. Chụp hình kết quả chạy các file script và lưu vào báo cáo.

* Ví dụ 2-1: variables.sh

Text

Description automatically generated

Hình 1: Kết quả chạy script variables.sh

* Ví dụ 2-2: try\_variables.sh

Text

Description automatically generated

Hình 2: Kết quả chạy script try\_variables.sh

* Ví dụ 2-3: if\_control.sh

Text

Description automatically generated

Hình 3: Kết quả chạy script if\_control.sh khi nhập yes

* Ví dụ 2-4: elif\_control.sh

Text

Description automatically generated

Hình 4: Kết quả chạy script elif\_control.sh khi nhập MORNING và nhập no

* Ví dụ 2-5: elif\_control2.sh

Text

Description automatically generated

Hình 5: Kết quả chạy script elif\_control2.sh khi nhấn Enter

* Ví dụ 2-6: for\_loop.sh

Text

Description automatically generated

Hình 6: Kết quả chạy script for\_loop.sh

* Ví dụ 2-7: for\_loop2.sh

Text

Description automatically generated

Hình 7: Kết quả chạy script for\_loop2.sh

* Ví dụ 2-8: password.sh

Text

Description automatically generated

Hình 8: Kết quả chạy script password.sh

* Ví dụ 2-9: while\_for.sh

Text

Description automatically generated

Hình 9: Kết quả chạy script while\_for.sh

* Ví dụ 2-10: until\_user.sh

Text

Description automatically generated

Hình 10: Kết quả chạy script until\_user.sh

* Ví dụ 2-11: case1.sh

Text

Description automatically generated

Hình 11: Kết quả chạy script case1.sh

* Ví dụ 2-12: case2.sh

Text

Description automatically generated

Hình 12: Kết quả chạy script case2.sh

* Ví dụ 2-13 case3.sh

Text

Description automatically generated

Hình 13: Kết quả chạy script case3.sh

## Task name 2: Viết chương trình cho phép nhập vào tên và mssv. Kiểm tra nếu mssv đó không trùng với mình thì bắt nhập lại. In ra màn hình kết quả.

- Viết file script để thực thi đặt tên là Bai2.sh nằm trong thư mục LAB02.

Text

Description automatically generated

Hình 14: Nội dung file script Bai2.sh

- Để viết chương trình theo như yêu cầu bài 2 ta tạo file script có tên là **Bai2.sh** bằng lệnh: **gedit Bai2.sh**.Sau đó, cấp quyền thực thi cho script bằng lệnh: **chmod +x Bai2.sh**. Cuối cùng, ta gọi thực thi script bằng lệnh: **./Bai2.sh**

Text

Description automatically generated

Hình 15: Kết quả chạy script Bai2.sh

- Để đảm bảo file script mà ta đã viết là đúng, ta có thể nhập một vài test-case để kiểm tra chương trình như sau: Dòng đầu tiên chương trình yêu cầu nhập tên, ta nhập **The** (có thể nhập tên khác). Dòng tiếp theo chương trình yêu cầu nhập mssv, ta nhập **sv19522253**, chương trình sẽ tiến hành kiểm tra và so sánh với **“19522253”** lúc này vì hai mssv này không giống nhau nên chương trình sẽ yêu cầu ta nhập lại. Ta nhập lại **1952**, chương trình sẽ tiến hành kiểm tra và so sánh với **“19522253”** lúc này vì hai mssv này không giống nhau nên chương trình sẽ yêu cầu ta nhập lại. Ta nhập lại **19522253**, chương trình sẽ tiến hành kiểm tra và so sánh với **“19522253”** lúc này vì hai mssv này là giống nhau nên chương trình sẽ in ra màn hình dòng chữ “**The co MSSV la 19522253”** và kết thúc chương trình. Có nghĩa là chúng ta đã viết thành công chương trình theo như yêu cầu bài 2.

## Task name 3: Viết chương trình cho phép nhập vào một số n. Kiểm tra nếu n < 10 thì bắt nhập lại. Tính tổng các số từ 1 đến n. In kết quả ra màn hình.

- Viết file script để thực thi đặt tên là Bai3.sh nằm trong thư mục LAB02.

Text

Description automatically generated

Hình 16: Nội dung file script Bai3.sh

- Để viết chương trình theo như yêu cầu bài 3 ta tạo file script có tên là **Bai3.sh** bằng lệnh: **gedit Bai3.sh**.Sau đó, cấp quyền thực thi cho script bằng lệnh: **chmod +x Bai3.sh**. Cuối cùng, ta gọi thực thi script bằng lệnh: **./Bai3.sh**

Text

Description automatically generated

Hình 17: Kết quả chạy script Bai3.sh

- Để đảm bảo file script mà ta đã viết là đúng, ta có thể nhập một vài test-case để kiểm tra chương trình như sau: Dòng đầu tiên chương trình yêu cầu nhập một số **n**, ta nhập **9**, chương trình sẽ tiến hành kiểm tra và so sánh với **10** lúc này vì 9 < 10 nên chương trình sẽ yêu cầu ta nhập lại. Ta nhập lại **10**, chương trình sẽ tiến hành kiểm tra và so sánh với **10** lúc này vì 10 = 10 nên chương trình sẽ tính tổng các số từ 1 đến 10 và thu được kết quả là 55 nên chương trình sẽ in ra màn hình dòng chữ **“Tong cac so tu 1 den 10 la: 55”** và kết thúc chương trình. Có nghĩa là chúng ta đã viết thành công chương trình theo như yêu cầu bài 3.

## Task name 4: Viết trình cho phép nhập vào một chuỗi. Kiểm tra chuỗi đó có tồn tại trong một file text (ví dụ test.txt) cùng thư mục hay không.

- Viết file script để thực thi đặt tên là Bai4.sh nằm trong thư mục LAB02

Graphical user interface, text

Description automatically generated

Hình 18: Nội dung script Bai4.sh

- Để viết chương trình theo như yêu cầu bài 4 ta tạo file script có tên là **Bai4.sh** bằng lệnh: **gedit Bai4.sh**.Sau đó, cấp quyền thực thi cho script bằng lệnh: **chmod +x Bai4.sh**. Cuối cùng, ta gọi thực thi script bằng lệnh: **./Bai4.sh**

Text

Description automatically generated

Hình 19: Kết quả chạy script Bai4.sh

- Để đảm bảo file script mà ta đã viết là đúng, ta có thể nhập một vài test-case để kiểm tra chương trình như sau: Khi thực thi chương trình ở lần đầu tiên, chương trình yêu cầu ta nhập chuỗi, ta nhập **IT007.M14**, chương trình sẽ tiến hành tìm và kiểm tra nội dung của tất cả các file trong cùng thư mục và không tìm thấy file nào có nội dung chứa chuỗi **IT007.M14** nên chương trình không xuất ra màn hình kết quả gì mà kết thúc chương trình luôn. Khi thực thi chương trình ở lần thứ hai, chương trình yêu cầu nhập chuỗi, ta nhập **main**, chương trình sẽ tiến hành tìm và kiểm tra nội dung của tất cả các file trong cùng thư mục và tìm thấy file **first.sh** có nội dung chứa chuỗi **main** nên đã xuất kết quả là nội dung của file **first.sh** ra màn hình. Dùng lệnh **ls** để kiểm tra xem có tồn tại file **first.sh** trong cùng thư mục (thư mục hiện hành) hay không, lúc này ta thấy có tồn tại file **first.sh** trong thư mục hiện hành. Có nghĩa là chúng ta đã viết thành công chương trình theo như yêu cầu bài 4.