

# ĐỒ ÁN

# SHELL

Trường: ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Môn học: HỆ ĐIỀU HÀNH

Lớp: CQ2017/21

Giáo viên: Thầy Lê Giang Thanh

Cô Nguyễn Thị Thanh Huyền

## 1. DANH SÁCH THÀNH VIÊN

MSSV	Họ & tên	Phân công
1712117	Nguyễn Huỳnh Thảo Nhi	Tìm hiểu đề bài; Code
1712177	Nguyễn Anh Thư	Làm báo cáo; Code

## 2. TỔ CHỨC/THIẾT KẾ CỦA ĐỒ ÁN

- Hàm `inputCommand(char* input)`: đọc command được người dùng input và lưu vào chuỗi `char* input`
- Hàm `parseCommand(char* input, char* args[])`: tách command người dùng nhập thành các token (phân biệt bởi khoảng trắng) và lưu vào mảng chuỗi `char* args[]`, mảng chuỗi này luôn có chuỗi kết thúc là `NULL`
- Hàm `main()`: với command người dùng nhập vào, chương trình gọi hàm `fork()` để tạo một tiến trình con (giá trị trả về  $< 0 \rightarrow$  tạo tiến trình thất bại; giá trị trả về  $= 0 \rightarrow$  đây là tiến trình con; giá trị trả về  $> 0 \rightarrow$  giá trị trả về chính là id của tiến trình con, đây là tiến trình cha). Với command kết thúc bằng kí tự “&”, thì tiến trình con và cha chạy song hành. Ngược lại, tiến trình cha sẽ gọi hàm `wait(&status)` để đợi tiến trình con thực thi xong. Ngoài ra, chương trình còn có cơ chế lưu lại command gần nhất vào history.
  - Command “!!”  $\rightarrow$  **history**: chương trình output command gần nhất được người dùng nhập và thực thi command đó. Nếu chương trình chạy vòng lặp lần đầu và chưa có command nào được lưu thì chương trình hiện thông báo “No commands in history”
  - Command có kí tự “<” hoặc “>”  $\rightarrow$  **input/output redirection**: chương trình gọi hàm `dup2(fd,STDIN_FILENO) / dup2(fd,STDOUT_FILENO)` để những gì mặc định hiện lên trên terminal (STDOUT) hoặc input từ bàn phím (STDIN) sẽ

được chuyển hướng vào file descriptor fd. Vì vậy kết quả từ command sẽ không được hiện lên màn hình terminal mà sẽ được lưu vào file .txt (trường hợp output redirection) và input từ người dùng sẽ được lấy ra từ file .txt (trường hợp input redirection)

- Command có kí tự “|” → **pipe**: Tiến trình con (\*) sau khi tạo thành công sẽ gọi hàm pipe(int fd[2]) với fd[1] đóng vai trò như cổng ghi và fd[0] đóng vai trò như cổng đọc, đồng thời tiến trình con(\*) tạo thêm một tiến trình con(\*\*) nữa. (\*) sẽ thực thi command ở trước “|” và gửi vào fd[1] sử dụng dup2() với cơ chế như output redirection. (\*\*) sẽ đọc từ fd[0] sử dụng dup2() với cơ chế như input redirection và thực thi lệnh nằm sau “|”

### 3. ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH

CHỨC NĂNG	ĐÁNH GIÁ
1. Tạo tiến trình con và thực thi command trong tiến trình con	✓
2. Chức năng lưu command gần nhất vào history	✓
3. Input & Output redirection	✓
4. Tiến trình giao tiếp qua pipe	✓
TỔNG	100%