

سوال دوم :

در این سوال ابتدا دستور ورودی ای که در کنسول از کاربر گرفته میشود توسط چند خط کد زیر پروسس میشوند و به صورت آرایه از دستورات (commands) درمی آید که به عنوان ورودی به تابع اصلی که وظیفه اجرای دستورات را دارد داده میشود -> execute_args

```
char** args ;
printf("prompt> " ) ;
char* line = read_line() ;
char *token = strtok(line , "|");
int l = strlen(token)-1 ;
char* commands[l] ;
int i=0 ;
while (token != NULL)
{
    commands[i] = token ;
    token = strtok(NULL , "|");
    i ++ ;
}
```

به تعداد دستورات ورودی که با | از یکدیگر جدا شده بودند یک pipe ایجاد میکنیم . به این صورت که کامند به غیر از کامند اول ورودی خود را از پایپ قبلی گرفته و خروجی اش را در پایپ بعدی میریزد . البته آخرین دستور خروجی اش را در کنسول نمایش میدهد و در پایپی نمیریزد . برای ایجاد هر دستور هم یک پروسس جدید با fork ساخته میشود که مسئول اجرا ان دستور باشد .

نکته مهمی که وجود دارد این است که برای اجرای صحیح هر دستور بایستی قبل از استفاده از execvp تمامی file descriptor ها که در اینجا پایپ ها هستن بسته شوند .

نمونه از خروجی برنامه :

```
zahra@400522058:~/Desktop/git/IUST-OS4021/Assignments/HW5/Questions/Q2$ ./Q2
prompt> ls -l | grep Q
-rwxrwxr-x 1 zahra zahra 17016 02:05 7   Q2
-rw-rw-r-- 1 zahra zahra 3152 02:29 6   Q2.c
zahra@400522058:~/Desktop/git/IUST-OS4021/Assignments/HW5/Questions/Q2$ ls -l | grep !
zahra@400522058:~/Desktop/git/IUST-OS4021/Assignments/HW5/Questions/Q2$ ls -l | grep Q
-rwxrwxr-x 1 zahra zahra 17016 02:05 7   Q2
-rw-rw-r-- 1 zahra zahra 3152 02:29 6   Q2.c
zahra@400522058:~/Desktop/git/IUST-OS4021/Assignments/HW5/Questions/Q2$ S
```

خروجی برنامه با خروجی کامند اصلی در ترمینال کاملاً مثل هم هستند 😊