

# Operating System Project #1

2017012351 이영섭

## 내용

1. SimpleShell 알고리즘
2. 소스파일
3. 컴파일 화면캡처
4. 명령어 실행결과

## Simple Shell 알고리즘

1. 공백으로 구분되어 있는 문자열 입력을 받아 공백을 제외하여 데이터를 저장함.
  - A. fgets()로 전체 문자열을 받고 (fgets()는 파일과 키보드로 입력을 받을 수 있고 개행문자를 기점으로 입력을 종료하기 때문에 공백으로 구분된 문자를 받기 용이하다. 또한, 개행문자까지 입력으로 처리하기 때문에 마지막 개행문자를 NULL값으로 바꿔줘야한다.)
  - B. strtok으로 나눈다.
2. 파싱된 데이터에 &가 존재하는지 확인
3. fork()를 실행하여 자식 프로세스를 생성
4. 문자열 입력을 저장해 놓은 배열을 완전탐색으로 살피는 동안
  - A. pid1 값이 음수값이면 오류처리
  - B. "<" 연산자를 발견하면 dup2를 이용해 입력 redirection으로 처리
    - i. 파일을 읽기 전용으로 열고 연산자가 위치한 곳을 NULL로 치환
  - C. ">" 연산자를 발견하면 dup2를 이용해 출력 redirection으로 처리
    - i. 파일을 쓰기 전용으로 열고 연산자가 위치한 곳을 NULL로 치환
  - D. "|" 연산자를 발견하면 pipe를 생성하고 fork()를 실행
    - i. pid2 값이 음수값이면 오류처리
    - ii. 자식 프로세스의 파이프를 dup2을 이용해 출력으로 설정
    - iii. 부모 프로세스의 파이프를 dup2를 이용해 입력으로 설정
5. is\_ampersand를 확인하여
  - A. 값이 1이면 자식을 기다리지 않고 프로세스 종료
  - B. 값이 0이면 자식을 기다렸다가 프로세스 종료

## 프로그램 소스파일

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <sys/wait.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <fcntl.h>

#define MAX_LINE 80

int main(int argc, char const *argv[]) {
    char *args[MAX_LINE/2 + 1];           //배열 선언
    int should_run = 1;                   //should_run 선언
    int mypipe[2];                         //파이프 선언
    int status;                           //process status
    int switch_run = 0;                   //should_run 상태

    while(should_run){
        memset(args, '\0', sizeof(args)); //포인터 배열 초기화
        printf("osh>");
        fflush(stdout);
        char input[MAX_LINE];              //임시 저장공간 선언
        fgets(input, MAX_LINE, stdin);     //문자열 입력
        input[strlen(input) - 1] = '\0';   //개행문자를 0으로 초기화

        char *ptr = strtok(input, " ");    //입력받은 문자열 파싱
        int ptrnum = 0;
        while(ptr != NULL){
            args[ptrnum] = ptr;
            ptrnum++;
            if(strcmp(ptr, "exit") == 0) switch_run = 1; //exit를 만나면 switch_run
의 상태가 바뀜
            ptr = strtok(NULL, " ");
        }

        if(switch_run == 1) should_run = 0;

        int i = 0, j = 0;                  //while 문 진행동안 진행위치 확인을 위한 변수
        int is_ampersand = 0;              //& 존재를 확인하기 위한 변수

        while(args[j] != NULL){            //입력받은 문자열에서 &를 발견하면 is_ampersa
nd의 상태가 바뀜
            if(strcmp(args[j], "&") == 0){
                is_ampersand = 1;
            }
        }
    }
}
```

```

    args[j] = NULL;
}
j++;
}
pid_t pid1, pid2; //fork()를 위한 pid 선언
pid1 = fork();

while(args[i] != NULL)
{

    if(pid1 < 0) //오류
    {
        fprintf(stderr, "Fork Failed\n");
        return 1;
    }

    else if(pid1 == 0){ //자식 프로세스

        if(is_ampersand == 1) //& 유무 확인
        {
            signal(SIGINT, SIG_IGN); //존재하면 신호무시
            printf("getpid : %d\n", getpid());
        }
        else
        {
            signal(SIGINT, SIG_DFL); //존재하지않으면 신호받음
        }

        if(strcmp(args[i], "<") == 0) // < redirection
        {
            int fd = open(args[i+1], O_RDONLY);
            dup2(fd, STDIN_FILENO);
            close(fd);
            args[i] = NULL;
            execvp(args[0], args);
        }

        else if(strcmp(args[i], ">") == 0) // > redirection
        {
            int fd = open(args[i+1], O_WRONLY | O_CREAT, 0666);
            dup2(fd, STDOUT_FILENO);
            close(fd);
            args[i] = NULL;
            execvp(args[0], args);
        }

        else if(strcmp(args[i], "|") == 0) // | 파이프 설정
        {

```

```

    args[i] = NULL;
    pipe(mypipe);
    pid2 = fork();

    if(pid2 < 0)                                //오류
    {
        fprintf(stderr, "Fork Failed\n");
        return 1;
    }

    else if(pid2 == 0)                            //출력 프로세스
    {
        dup2(mypipe[1], STDOUT_FILENO);
        close(mypipe[0]);
        execvp(args[0], args);
    }

    else                                          //입력 프로세스
    {
        dup2(mypipe[0], STDIN_FILENO);
        close(mypipe[1]);
        execvp(args[i+1], &args[i+1]);
    }
}

else                                          //어떠한 redircetion 도 없
을 경우
{
    if(args[i+1] == NULL) execvp(args[0], args);
}

else                                          //부모 프로세스
{
    if(is_ampersand == 1)                    //자식 프로세스가 백그라운드
로 수행중이라면
    {
        waitpid(pid1, &status, WNOHANG);    //부모 프로세스는 자식을 기
다리지 않음
    }
    else
    {
        waitpid(pid1, &status, 0);          //자식 프로세스가 백그라운드
에서 실행중이 아니라면
    }
    //부모 프로세스는 자식 프로
세스를 기다렸다가 실행
}
i++;

```

```

    }
}
return 0;
}

```

## 프로그램 컴파일 과정

```

(null)
(null)
Child Complete
Child Complete
osh>exit
is_ampersand : 0
exit
(null)
(null)
Child Complete
AC
~ gcc shell.c
~/a.out
osh>AC
gcc -v shell.c
Apple clang version 12.0.0 (clang-1200.0.32.27)
Target: x86_64-apple-darwin20.3.0
Thread model: posix
InstalledDir: /Library/Developer/CommandLineTools/usr/bin
"/Library/Developer/CommandLineTools/usr/bin/clang" -cc1 -triple x86_64-apple-macosx11.0.0 -Wdeprecated-objc-isa-usage -Werror=deprecated-objc-isa-usage -Werror=implicit-function-declaration -emit-obj -mrelax-all -disable-free -disable-llvm-verifier -discard-value-names -main-file-name shell.c -mrelocation-model pic -pic-level 2 -mthread-model posix -mframe-pointer=all -fno-strict-return -masm-verbose -munwind-tables -target-sdk-version=11.0 -target-cpu penryn -dwarf-column-info -debugger-tuning=lldb -target-linker-version 609.6 -v -resource-dir /Library/Developer/CommandLineTools/usr/lib/clang/12.0.0 -isysroot /Library/Developer/CommandLineTools/SDKs/MacOSX.sdk -I/usr/local/include -internal-isystem /Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/local/include -internal-isystem /Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include -internal-externc-isystem /Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include -Wno-c99-designator -Wno-final-dtor-non-final-class -Wno-extra-semi-stmt -Wno-misleading-indentation -Wno-quoted-include-in-framework-header -Wno-implicit-fallthrough -Wno-enum-enum-conversion -Wno-enum-float-conversion -fdebug-compilation-dir /Users/lys/desktop/univ/cpp -ferror-limit 19 -fmessage-length 80 -stack-protector 1 -fstack-check -mdarwin-stkchk-strong-link -fblocks -fencode-extended-block-signature -fregister-global-dtors-with-atexit -fgnuc-version=4.2.1 -fobjc-runtime=macosx-11.0.0 -fmax-type-align=16 -fdiagnostics-show-option -fcolor-diagnostics -o /var/folders/4v/gvsn66p55kb82v8gyphfqp_r0000gn/T/shell-df4b18.o -x c shell.c
clang -cc1 version 12.0.0 (clang-1200.0.32.27) default target x86_64-apple-darwin20.3.0
ignoring nonexistent directory "/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/local/include"
ignoring nonexistent directory "/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/Library/Frameworks"
#include "...": search starts here:
#include <...>: search starts here:
/usr/local/include
/Library/Developer/CommandLineTools/usr/lib/clang/12.0.0/include
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/include
/Library/Developer/CommandLineTools/usr/include
/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/System/Library/Frameworks (framework directory)
End of search list.
"/Library/Developer/CommandLineTools/usr/bin/ld" -demangle -lto_library /Library/Developer/CommandLineTools/usr/lib/libLTO.dylib -no_deduplicate -dynamic -arch x86_64 -platform_version macos 11.0.0 11.0 -syslibroot /Library/Developer/CommandLineTools/SDKs/MacOSX.sdk -o a.out -L/usr/local/lib -L/var/folders/4v/gvsn66p55kb82v8gyphfqp_r0000gn/T/shell-df4b18.o -lSystem /Library/Developer/CommandLineTools/usr/lib/clang/12.0.0/lib/darwin/libclang_rt.osx.a
~/desktop/univ/cpp
py base

```

## 명령어 실행결과

ls -l

```

osh>ls -l
total 224
-rw-r--r--  1 lys  staff   735  2  8 23:09 1158.cpp
-rw-r--r--  1 lys  staff   106  3  3 16:07 1699.cpp
-rw-r--r--  1 lys  staff  1023  2  9 00:02 1764.cpp
-rw-r--r--@ 1 lys  staff  1206  9  9 2020 669-1.cpp
-rw-r--r--  1 lys  staff   655 12 26 15:49 C.cpp
-rw-r--r--  1 lys  staff   518 12 26 16:09 D.cpp
-rwxr-xr-x  1 lys  staff 50408  3 28 21:20 a.out
-rw-r--r--@ 1 lys  staff  1167 10 13 15:36 lowpivot.cpp
-rw-r--r--@ 1 lys  staff  1397 10 13 15:32 midpivot.cpp
-rw-r--r--@ 1 lys  staff   138  3 28 20:33 out.txt
-rw-r--r--@ 1 lys  staff  1167 10 13 15:32 randompivot.cpp
-rw-r--r--@ 1 lys  staff  4571  3 28 21:20 shell.c
-rw-r--r--  1 lys  staff   208  3 23 19:39 test.c
-rw-r--r--@ 1 lys  staff  1125 10 11 16:39 test.cpp
-rw-r--r--@ 1 lys  staff    4  3 27 23:52 test.txt
osh>

```

ls -l &

```
osh>ls -l &
osh>getpid : 13256
getpid : 13256
total 224
-rw-r--r--  1 lys  staff    735  2  8 23:09 1158.cpp
-rw-r--r--  1 lys  staff    106  3  3 16:07 1699.cpp
-rw-r--r--  1 lys  staff   1023  2  9 00:02 1764.cpp
-rw-r--r--@  1 lys  staff   1206  9  9 2020 669-1.cpp
-rw-r--r--  1 lys  staff    655 12 26 15:49 C.cpp
-rw-r--r--  1 lys  staff    518 12 26 16:09 D.cpp
-rwxr-xr-x  1 lys  staff  50408  3 28 21:20 a.out
-rw-r--r--@  1 lys  staff   1167 10 13 15:36 lowpivot.cpp
-rw-r--r--@  1 lys  staff   1397 10 13 15:32 midpivot.cpp
-rw-r--r--@  1 lys  staff    138  3 28 20:33 out.txt
-rw-r--r--@  1 lys  staff   1167 10 13 15:32 randompivot.cpp
-rw-r--r--@  1 lys  staff   4571  3 28 21:20 shell.c
-rw-r--r--  1 lys  staff    208  3 23 19:39 test.c
-rw-r--r--@  1 lys  staff   1125 10 11 16:39 test.cpp
-rw-r--r--@  1 lys  staff     4  3 27 23:52 test.txt
```

ls -s > ls.txt

```
osh>ls -s > ls.txt
osh>
```

ls -h > lsh.txt &

```
osh>ls -h > lsh.txt &
osh>getpid : 13769
getpid : 13769
getpid : 13769
```

cat -n < ls.txt

```
osh>cat -n < ls.txt
 1 total 240
 2  8 1158.cpp
 3  8 1699.cpp
 4  8 1764.cpp
 5  8 669-1.cpp
 6  8 C.cpp
 7  8 D.cpp
 8 104 a.out
 9 16 cat.txt
10  8 lowpivot.cpp
11  0 ls.txt
12  8 midpivot.cpp
13  8 out.txt
14  8 randompivot.cpp
15 16 shell.c
16  8 test.c
17  8 test.cpp
18  8 test.txt
osh>
```

cat -e < lsh.txt

```
osh>cat -e < lsh.txt &
osh>getpid : 13992
getpid : 13992
getpid : 13992
1158.cpp$
1699.cpp$
1764.cpp$
669-1.cpp$
C.cpp$
D.cpp$
a.out$
cat.txt$
lowpivot.cpp$
ls.txt$
lsh.txt$
midpivot.cpp$
out.txt$
randompivot.cpp$
shell.c$
test.c$
test.cpp$
test.txt$
█
```

ls -l | less -N

```
osh>ls -l | less -N
osh>█
```

```
1 total 256
2 -rw-r--r-- 1 lys staff 735 2 8 23:09 1158.cpp
3 -rw-r--r-- 1 lys staff 106 3 3 16:07 1699.cpp
4 -rw-r--r-- 1 lys staff 1023 2 9 00:02 1764.cpp
5 -rw-r--r--@ 1 lys staff 1206 9 9 2020 669-1.cpp
6 -rw-r--r-- 1 lys staff 655 12 26 15:49 C.cpp
7 -rw-r--r-- 1 lys staff 518 12 26 16:09 D.cpp
8 -rwxr-xr-x 1 lys staff 50408 3 28 21:36 a.out
9 -rw-r--r--@ 1 lys staff 5551 3 28 21:28 cat.txt
10 -rw-r--r--@ 1 lys staff 1167 10 13 15:36 lowpivot.cpp
11 -rw-r--r-- 1 lys staff 231 3 28 21:30 ls.txt
12 -rw-r--r-- 1 lys staff 161 3 28 21:31 lsh.txt
13 -rw-r--r--@ 1 lys staff 1397 10 13 15:32 midpivot.cpp
14 -rw-r--r--@ 1 lys staff 808 3 28 21:23 out.txt
15 -rw-r--r--@ 1 lys staff 1167 10 13 15:32 randompivot.cpp
16 -rw-r--r--@ 1 lys staff 4571 3 28 21:20 shell.c
17 -rw-r--r-- 1 lys staff 208 3 23 19:39 test.c
18 -rw-r--r--@ 1 lys staff 1125 10 11 16:39 test.cpp
19 -rw-r--r--@ 1 lys staff 4 3 27 23:52 test.txt
```

(END)



```
ls -l | less -N &
```

```
osh>ls -l | less -N &  
osh>getpid : 14152  
getpid : 14152  
getpid : 14152
```

```
1 total 256  
2 -rw-r--r-- 1 lys staff 735 2 8 23:09 1158.cpp  
3 -rw-r--r-- 1 lys staff 106 3 3 16:07 1699.cpp  
4 -rw-r--r-- 1 lys staff 1023 2 9 00:02 1764.cpp  
5 -rw-r--r--@ 1 lys staff 1206 9 9 2020 669-1.cpp  
6 -rw-r--r-- 1 lys staff 655 12 26 15:49 C.cpp  
7 -rw-r--r-- 1 lys staff 518 12 26 16:09 D.cpp  
8 -rwxr-xr-x 1 lys staff 50408 3 28 21:36 a.out  
9 -rw-r--r--@ 1 lys staff 5551 3 28 21:28 cat.txt  
10 -rw-r--r--@ 1 lys staff 1167 10 13 15:36 lowpivot.cpp  
11 -rw-r--r-- 1 lys staff 231 3 28 21:30 ls.txt  
12 -rw-r--r-- 1 lys staff 161 3 28 21:31 lsh.txt  
13 -rw-r--r--@ 1 lys staff 1397 10 13 15:32 midpivot.cpp  
14 -rw-r--r--@ 1 lys staff 808 3 28 21:23 out.txt  
15 -rw-r--r--@ 1 lys staff 1167 10 13 15:32 randompivot.cpp  
16 -rw-r--r--@ 1 lys staff 4571 3 28 21:20 shell.c  
17 -rw-r--r-- 1 lys staff 208 3 23 19:39 test.c  
18 -rw-r--r--@ 1 lys staff 1125 10 11 16:39 test.cpp  
19 -rw-r--r--@ 1 lys staff 4 3 27 23:52 test.txt
```

```
(END)
```