

시스템프로그래밍

assignment1-1

학번: 2021202045

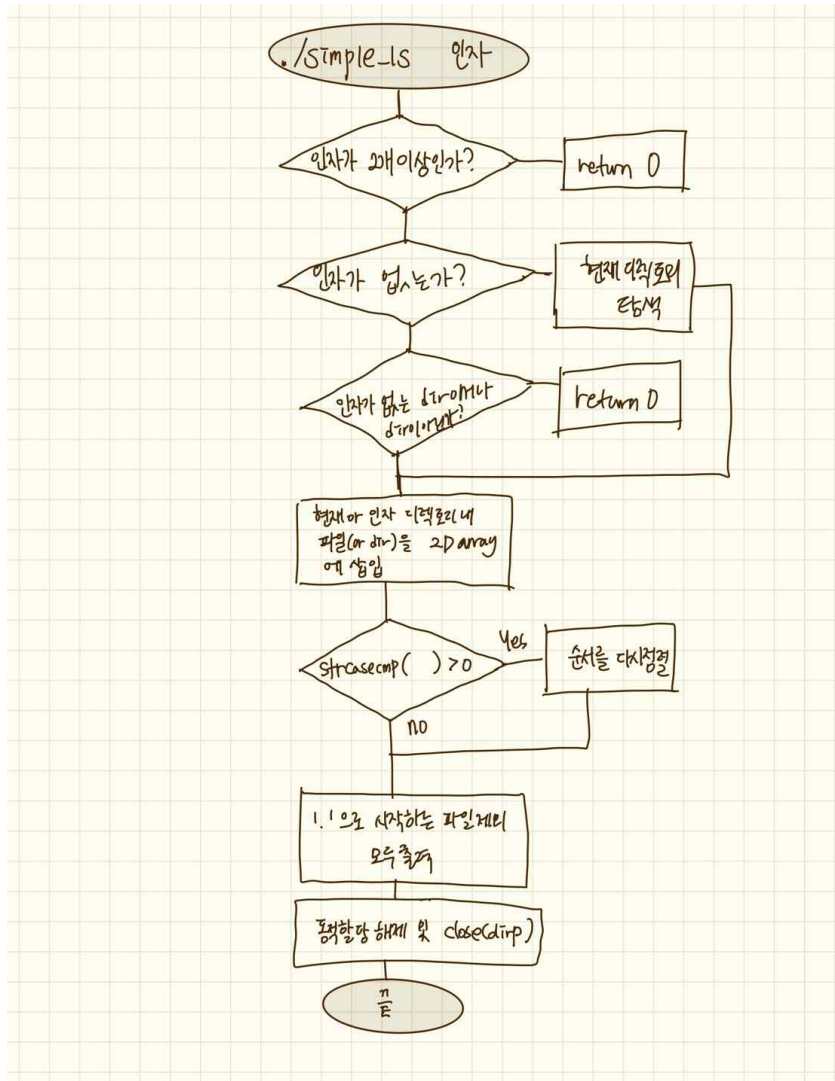
이름: 김예은

담당 교수님: 최상호 교수님

I . Introduction

리눅스 시스템에서 파일 이름만 출력하는 (옵션이 없는) ls기능을 하는 `simple_ls` 명령어를 구현하는 과제이다. './simple_ls 인자'로 구성되며, 인자값이 없을 때는 default로 current directory를 탐색하여 결과를 출력한다. 그 외에 인자값이 두 개 이상이거나 없는 디렉토리나 디렉토리가 아닌 파일을 인자값으로 넣으면 오류 메시지를 출력한다. 파일 이름이 결과 화면에 출력될 때에는 대소문자 구분 없이 알파벳 순으로 정렬되어 출력되게끔 한다. 실행 파일 이름은 `simple_ls`, c파일 이름은 `학번_simple_ls.c`로 한다.

II . flowchart



III. Pseudo code

```
'./simple_ls 인자값'으로 구성된 명령어에서  
if(인자값이 두개 이상이면) {  
    오류메세지를 출력하고 프로그램을 끝낸다.  
}
```

```
if(인자값이 없는 디렉토리 또는 디렉토리가 아닌 파일을 명시){ 오류 메세지를 띄우고 프로그  
램을 끝낸다.}
```

```
if(인자값이 없을 때) { 현재 디렉토리를 탐색한다.}
```

디렉토리 내 파일명을 저장하기 위해 2D array를 만든다.
세로는 파일의 갯수, 가로는 파일 이름의 최대길이+1로 만든다.

```
while(dirp에 저장된 디렉토리를 읽으면서 NULL값이 아닌동안){  
    디렉토리 내의 파일이름을 2D array[dir_num1]에 복사한다.  
    dir_num1은 +1증가  
}  
for(i=0부터 파일 갯수-1까지 하나씩 증가){  
    for(j=i+1부터 파일갯수까지 하나씩 증가){  
        만약 arr[j]가 arr[i]보다 사전식 앞에 있는 단어일때(대,소문자 구분없이)  
            tmp에 arr[i]값 임시저장  
            arr[i]값은 arr[j]로 바뀜  
            arr[j]는 tmp값으로 바뀜  
        }  
    }  
}
```

```
for(i =0부터 파일갯수미만까지 하나씩 증가){  
    if(파일이름이 '.'으로 시작) {출력을 생략한다.}  
    파일명을 출력한다.  
}
```

```
for(k=0부터 파일개수미만까지 하나씩 증가){  
    arr[k] 동적메모리할당 해제  
}  
arr동적할당 해제  
tmp동적할당 해제  
탐색한 디렉토리 닫기
```

IV. 결과화면

```
kw2021202045@ubuntu:~/kw_hw$ make
gcc -o simple_ls 2021202045_simple_ls.c
kw2021202045@ubuntu:~/kw_hw$ ./simple_ls
2021202045_simple_ls.c
apple
Banana
hello1.c
Makefile
simple_ls
test.txt
test2
```

./simple_ls하고 인자값이 없을 때 default인 current directory를 탐색하여 출력하는 것을 볼 수 있다. 이때, apple과 Banana의 경우 아스키코드로 하면 B(66), a(97)이지만, 대소문자 구분 없이 정렬하므로 apple이라는 디렉토리가 Banana보다 먼저 출력되는 것을 볼 수 있다. 나머지 파일들도 알파벳순서로 출력이 잘 된 것을 볼 수 있다.

```
kw2021202045@ubuntu:~/kw_hw$ ./simple_ls hello1.c
simple_ls: cannot access 'hello1.c' : No such directory
kw2021202045@ubuntu:~/kw_hw$ ./simple_ls NOT_file
simple_ls: cannot access 'NOT_file' : No such directory
kw2021202045@ubuntu:~/kw_hw$ ./simple_ls apple Banana
simple_ls: only one directory path can be processed
```

첫 번째, hello1.c의 경우 디렉토리가 아닌 파일이다. ./simple_ls hello1.c를 했을 때, 오류 메시지를 출력하고 프로그램이 끝나는 것을 볼 수 있다.

두 번째, NOT_file의 경우 없는 파일이자 디렉토리이다. 따라서 ./simple_ls NOT_file을 했을 때, 오류메시지를 출력하고 프로그램이 끝난다.

세 번째, 디렉토리 두 개를 즉, 파일 경로를 두 개 이상 주었을 때이다. 하나의 directory path만 처리 가능하다는 오류 메시지를 출력하고 프로그램이 끝난다.

```
kw2021202045@ubuntu:~/kw_hw$ ./simple_ls apple
good
test
kw2021202045@ubuntu:~/kw_hw$ cd apple
kw2021202045@ubuntu:~/kw_hw/apple$ ls
good test
kw2021202045@ubuntu:~/kw_hw/apple$
```

apple이라는 디렉토리를 인자로 주면 apple 디렉토리 내의 파일들을 알파벳순서로 출력해준다. 실제로 cd apple을 해본 후, ls를 해주면 같은 결과가 나오는 것을 볼 수 있다.

위의 세 경우에서 '.'으로 시작하는 파일은 출력이 생략된 것을 확인 할 수 있다.

V. 고찰

우선 리눅스 시스템에서 터미널로 인자값을 넘겨 받는 것은 처음이기 때문에 구글링을 해보았다. c언어에서는 명령행 인자를 main함수의 입력 인자로 받아 사용할 수 있는데 다음과 같다.

int main(int argc, char**argv)에서 argc란, 명령행 인자의 개수, argv란, 명령행 인자 벡터를 뜻한다. argc의 최소 1개 이상이며 명령행 인자를 공백 단위로 구분하여 단어별로 문자열을 만들고 이들의 위치를 순차적으로 보관한 벡터가 argv이다.

c언어는 오랜만이라서. string형이 아니라 char형으로 여러 개의 문자열을 어떻게 받아오면 좋을지 고민을 오래했다. 2차원 동적할당을 통해 문자열을 받아오는 것을 통해 앞으로의 과제에서 점점 c언어에 대해 감을 찾고싶다.

VI. reference

1

<https://ehpub.co.kr/%eb%a6%ac%eb%88%85%ec%8a%a4-%ec%8b%9c%ec%8a%a4%ed%85%9c-%ed%94%84%eb%a1%9c%ea%b7%b8%eb%9e%98%eb%b0%8d-7-3-%ed%99%98%ea%b2%bd-%eb%b3%80%ec%88%98-getenv-putenv-%ed%95%a8%ec%88%98/>

:main함수의 입력인자 받기에 대한 자료