**프로그램 구조 설명**

Command-line argument를 이용하여 옵션을 실행할 수 있도록 하였다.

**Main 함수 안에서**

구조체 포인터 list를 선언하고 malloc 함수를 사용하여 동적 메모리를 할당하였다.

Read mode로 data.txt 파일을 열어(fopen) 파일이 끝날 때까지 한 string씩 읽어 ‘:’을 기준으로 짤라(strtok) list의 각 멤버 변수에 복사해(strcpy) 넣었다. 그 후 data.txt 파일을 닫았다. (fclose)

Argument의 개수가 1이면 기본 수행이기 때문에 사용법 설명 (tel\_info 실행)①

Argument의 개수가 1보다 크면 밑과 같이 실행

-첫번째 argument vector == -a (strcmp) -> 전화번호 추가 (tel\_add 실행)②

-첫번째 argument vector == -d (strcmp) -> 전화번호 삭제 (tel\_delt 실행)③

-첫번째 argument vector == -l (strcmp) -> 전화번호 리스트 (tel\_list 실행)④

-argument 개수 == 2 -> 전화번호 검색 (tel\_srch 실행)⑤

**tel\_info 함수 ①**

search, add, delete, list mode 설명을 출력

**tel\_add 함수②**

옵션 입력할 때 공백을 구분으로 name, phone, memo를 함께 입력한다.

이 정보들을 출력하고 추가여부를 물어보고 Y가 입력되면(getchar) 다음과 같이 실행한다.

정보들을 llst의 각 멤버 변수에 복사한다.(strcpy)

Write 모드로 data.txt 파일을 열었다.(fopen)

문자열 포인터 input을 선언하고 동적 메모리를 할당하였다. Input에 list의 name을 복사해 넣고(strcpy) :, phone, :, list, \n 순으로 연결해(strcat) 한 문장으로 만들어 넣었다. 그리고 input의 내용을 data.txt파일에 넣었다. (fputs)

그 후 data.txt 파일을 닫았다. (fclose)

**tel\_delt 함수③**

smem -> 정수형 배열 -> 삭제하고자하는 내용이 있는 list의 번호

count -> 정수형 변수 -> 삭제하고자하는 내용이 있는 list 개수

list의 멤버변수들에 입력된 문자열 k가 있다면(strstr) count를 증가하고 count와 함께 그 정보들을 출력하고 smem[count]에 그 list의 번호를 저장한다.

list의 멤버변수들에 입력된 문자열 k가 없다면 -> count == 0 -> no match found 출력

list의 멤버변수들에 입력된 문자열 k가 있다면 -> count != 0 -> 다음과 같이 실행

삭제하고자하는 list의 내용의 count를 입력 = num

Smem[num] = list의 번호 = num2

인덱스를 하나씩 앞당기면서 list[num2] 내용 삭제

Write 모드로 data.txt 파일을 열었다.(fopen)

문자열 포인터 input을 선언하고 동적 메모리를 할당하였다. Input에 list의 name을 복사해 넣고(strcpy) :, phone, :, list, \n 순으로 연결해(strcat) 한 문장으로 만들어 넣었다. 그리고 input의 내용을 data.txt파일에 넣었다. (fputs) 이 과정을 0부터 한문장이 빠진 크기까지 반복했다.

그 후 data.txt 파일을 닫았다. (fclose)

**tel\_list 함수④**

member 개수 == 0 -> empty 출력

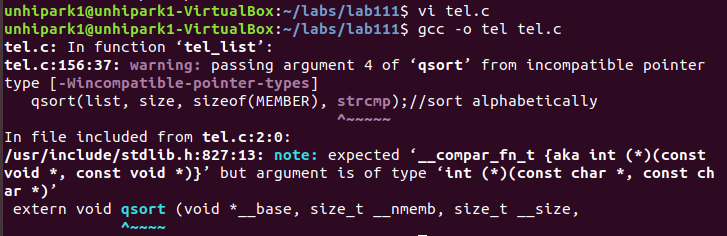
member 개수 == 1 -> 내용 출력

member 개수 >= 2 -> qsort 후 모두 출력

**tel\_srch 함수⑤**

list의 멤버변수들에 입력된 문자열 k가 있다면(strstr) count를 증가하고 count와 함께 그 정보들을 출력한다

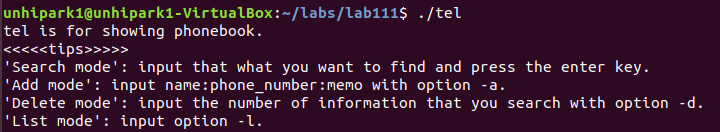
**컴파일 방법**



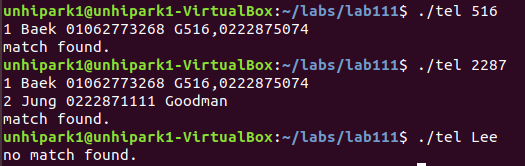
Modular programming을 사용하지 않아 gcc -o tel tel.c의 방법을 통해 컴파일을 하였다.

**예제수행과 수행 캡처 파일**

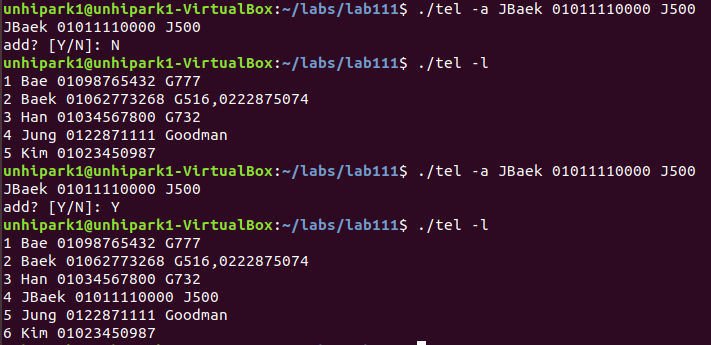
-기본



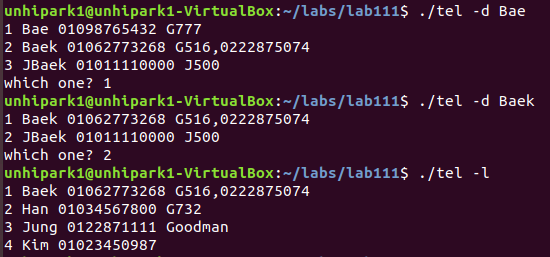
-search



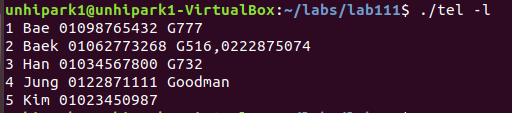
-add



-delete



-list



**소스코드**

