

HW5. 구조체 배열

부산대학교 바이오산업기계공학과

2014-45710

김지훈

Github ID: JihoonKim201445710

제출일: 2020-06-02

1. 구현 내용에 대한 설명 (50점)

- (1) 주요 변수 설명 : extern contact*는 phoneBookMain.c의 contact* 전역 변수 선언, static은 int_count_service = 0;와 int pw_count = 0;를 정적 함수로 선언하여 해당 c파일에서 만 참조 가능하게 했고, char로 형식 지정을 했습니다. 또 link.c와 save_data.c를 따로 만들어서 txt파일에 등록한 정보를 저장하게끔 했습니다.
- (2) 주요 자료 구조 설명 : typedef struct Contact는 구조체 선언, contact* curr = head는 링크드 리스트 구조체의 헤드 생성, contact* _new = (contact*)malloc(sizeof(contact));는 _new라는 구조체를 만들어서 새로 등록하는 리스트를 저장, curr2 = curr->next->next; 포인터에 다음 다음 구조체 주소 저장을 하기위해 구현, curr = curr->next;는 while 구문이 돌게 하기 위해서 구조체 선언을 하였고, contact* curr2;는 구조체를 하나 더 만들어서 delete.c에서 활용하기 위해서 선언함으로써 head->data->data->data(더미, 삭제용)->tail로 링크드 리스트 구조체를 사용하게 했습니다.
- (3) 주요 함수 구현 방법 설명 : register.c에서는 우선 패스워드 점검을 위해 패스워드 지정값을 만들어주고, 패스워드 입력 카운트가 최대3이 되게끔 지정하여서 성공 할시 strcpy로 입력값을 구조체에 등록하게 하여 이름과 번호를 저장시키게 하고, 각각 틀린 횟수마다 if 구문을 추가하여 틀리다고 알리는 경고문을 송출하게 하였습니다. search.c에서는 while구문으로 링크드 리스트를 활용해서 전체 리스트 송출과 이름검색을 하게 하였고 입력한 이름이 맞을시 이름과 번호를 출력, 그렇지 않을시 틀리다고 안내문을 송출하게 하였습니다. delete.c에서는 search와 비슷하게 구현하였지만 삭제를 위한 구조체를 하나 더 만들었고 free(curr->next);로 메모리 주소를 해제 함으로써 주소록을 삭제 할 수 있게 하였습니다. fp = fopen("D:\\vap\\phonelist.txt", "rb");로 txt파일에 등록할 수 있고 지울수 있고 하게끔 유도시켰습니다.
- (4) Makefile 설명 :
test.out : phoneBookMain.o register.o search.o delete.o link.o save_data.o
gcc -o test.out phoneBookMain.o register.o search.o delete.o link.o save_data.o

컴파일링을 끝낸 것을 결과물로 도출

phoneBookMain.o : phoneBookMain.c

gcc -c -o phoneBookMain.o phoneBookMain.c

phoneBookMain.c를 컴파일링

register.o : register.c

gcc -c -o register.o register.c

register.c를 컴파일링

search.o : search.c

gcc -c -o search.o search.c

search.c를 컴파일링

delete.o : delete.c

gcc -c -o delete.o delete.c

delete.c를 컴파일링

link.o : link.c

gcc -c -o link.o link.c

link.c 컴파일링

save_data.o : save_data.c

gcc-c -o save_data.o save_data.c

save_data.c 컴파일링

clean :

rm *.o test.out

코드가 오류가 났거나 다른이유에서 컴파일링한 파일과 결과물 파일을 한번에 지우고자 하는 명령어

2. 실행 방법 설명 (20점)

- (1) 사용한 운영체제 및 컴파일러의 종류 : Windows 10 // Visual Studio 2019 and Ubuntu Linux
- (2) 컴파일 방법 및 실행 방법 : Linux 로 makefile을 만들어 gcc를 이용하여 컴파일링
- (3) 동작을 확인할 수 있는 실행 화면 캡처

```

===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Exit >>>
Please enter your service number (1-5)>

===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Exit >>>
Please enter your service number (1-5)> 1
Registration
qwer1234

New user name: jihoon
PhoneNumber: 01062463180
jihoon registered!
main 010-1234-5678
jihoon 010-6246-3180
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Exit >>>
Please enter your service number (1-5)> 2
Print all contents in the PhoneBook
<<Display all contact in the PhoneBook>>
main
010-1234-5678
jihoon
010-6246-3180

main 010-1234-5678
jihoon 010-6246-3180
i 010-6246-3180
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Exit >>>
Please enter your service number (1-5)> 3
Search by Name
>Enter a name to search:
main
main 010-1234-5678
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Exit >>>
Please enter your service number (1-5)>

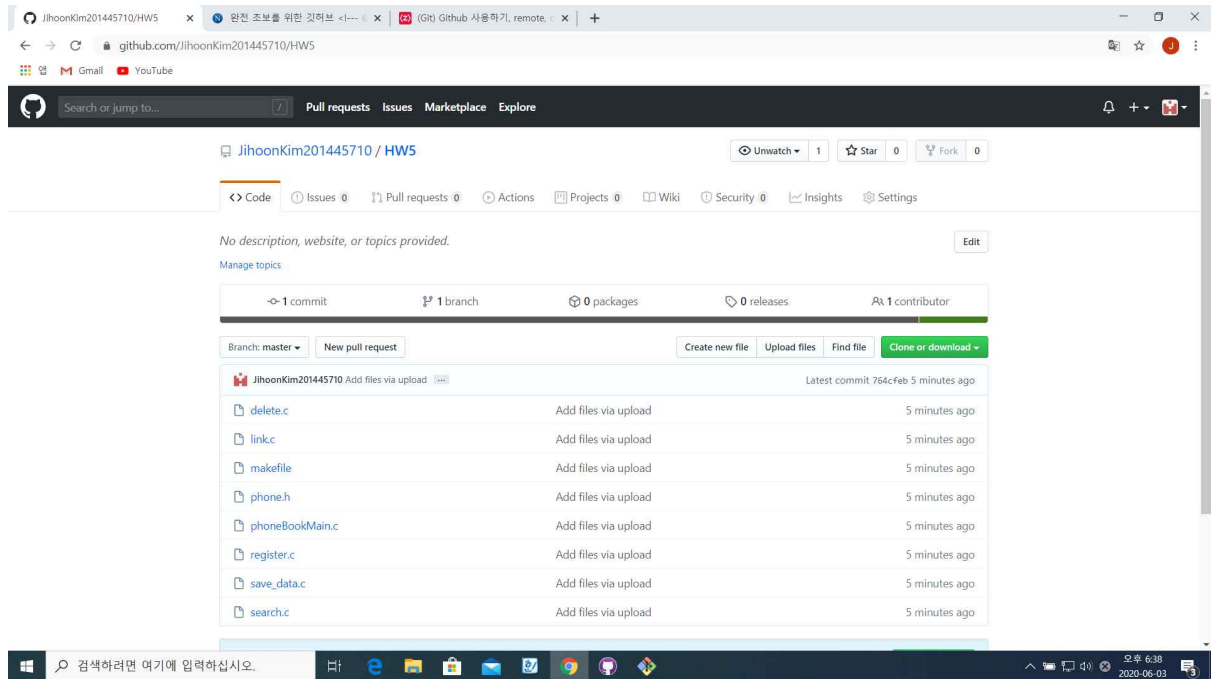
```

```
Microsoft Visual Studio 디버그 콘솔
rain 010-1234-5678
===== Telephone Book Management =====
<<<1. Register 2. Print All 3. Search by ID 4. Delete 5. Exit >>>
Please enter your service number (1-5)> 1
Registration
password:
adad
ot matched !!!
password:
sdasdas
ot matched (twice) !!!
password:
ssda
ot matched (3 times) !!!
password:
ddd
You are not allowed to access PhoneBook.
C:\Users\User\Desktop\지훈아 공부하자\오픈소스\makefile제작용\test3\Debug\test3.exe(프로세스 3396개)이(가) 종료되었습니다.
(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사
도를 설정합니다.
창을 닫으려면 아무 키나 누르세요...

rain 010-1234-5678
jihoon 010-6246-3180
ji 010-6246-3180
===== Telephone Book Management =====
<<<1. Register 2. Print All 3. Search by ID 4. Delete 5. Exit >>>
Please enter your service number (1-5)> 4
Deletion is done
>>Enter a name to Delete: rain
rain is deleted
===== Telephone Book Management =====
<<<1. Register 2. Print All 3. Search by ID 4. Delete 5. Exit >>>
Please enter your service number (1-5)> _
```

3. Github 화면 (20점)

- (1) cloning, adding, committing, push을 위한 github 명령들을 포함
- (2) 소스 코드와 makefile을 push한 후, 본인의 Github repository를 스크린 캡처하여 포함



4. 논의 사항 (10점)

- 숙제를 하면서 새로이 알게 된 내용(수업 시간 이외의 내용)

하다가 막히는 건 인터넷을 찾아가며 했는데, 링크드 리스트를 이용해서 구조체를 연결시키는 법을 알았습니다. 근데 2주 뒤에 배울 내용이라서 미리 연습한 걸로 쳐야겠습니다.

- 숙제를 하는 중에 어려웠던 점 등을 기술

과제를 뒤 늦게 보고 급하게 하느라 디버깅도 안되고 막히는 부분이 많았습니다. 특히 등록 부분에서 비밀번호 입력 할 때 비밀번호가 맞을시 등록 화면으로 넘어가야 하는데 자꾸 안되서 여러번 하다가 if문 사이에 비밀번호 부분을 넣는 것으로 해결을 했습니다. 그리고 저장한 정보가 프로그램을 켜다가 켜면 날라가기도 했어서 그것도 해결하느라 인터넷 찾아보고 링크드 리스트 구조로 다시 코드 잡아 엮고 하니 해결이 되었습니다.