|  |
| --- |
| HW5. 구조체 배열  학번: 201824468  학과:정보컴퓨터공학전공  이름: 박건우  **Github ID:parkgunwoo123**  제출일: 2020-05-19 |

보고서에는 다음의 내용을 포함할 것.

# 구현 내용에 대한 설명 (50점)

1. 주요 변수 설명: extern과 static 변수는 반드시 설명함

PhoneBook변수는 사람의 정보를 담아두는 구조체이고,

header파일에 있는 int size = 0;은 구조체 PhoneBook의 원소역할을 수행한다.

구조체 Contact의 Name과 PhoneNumber은 각각 사람의 이름과 전화번호를 각각 저장하는 변수이다.

헤더파일의 static size를 사용하기 위해서 메인함수에 plus,minus findsize,returnsize함수를 만들어 static size 값을 메인함수의 전역변수Size에 저장하고 extern을 사용해 각 소스코드에 Size를 사용할 수 있게 하였다.

소스파일에 char Name은 임시로 입력받은 이름 문자열을 저장해서 자신이 원하는 이름 문자열과 비교하기 위해서 선언한 것이고, char password도 입력한 password가 맞는지 아닌지 확인하기위해서 임시로 저장을 해놓기 위해서 선언하였고, register.c의 Num변수는 패스워드가 만약 틀릴 경우에 3번까지 물어보는 것이 주어졌음으로, Num이 4가 되면 빠while문을 빠져나오게 하기위해서 선언하였다.

1. 주요 자료 구조 설명

구조체 배열 PhoneBook을 선언하여 [0~최대49]까지의 정보를 저장할 수 있다.

각각의 소스파일의 함수에서 선언된 name이나 password등의 변수들은 스택영역에서 일시적으로 저장되었다가, 함수가 끝나면 사라진다. 헤더파일의 static size는 정적 전역 변수로서 사용되고, 스택과는 다른 영역에 저장되어 메인함수가 끝나야 메모리가 해제된다.

Static이 사용되어서 그 소스코드 내에서만 사용 가능하다.

1. 주요 함수 구현 방법 설명

Delete는 지워야 할 배열원소에 그 다음 원소들의 값들을 집어넣어서 한 칸씩 당겨서 결론적으로 그 지워야 할 부분만 없어지고 끝의 원소가 1 감소한다. 입력된 이름이 없다면 없다는 문구를 출력한다.

print함수는 for문을 사용해서 원소들을 출력하였다.

register부분은 지정된 패스워드를 저장하는 char문을 미리 선언해놓고, 입력된 패스워드가 맞는지를 확인하기 위해서 while문에 num !=4를 조건으로 지정했는데, 3번까지 입력의 기회가 있기 때문이다. String라이브러리에 있는 strcmp함수로 패스워드가 맞는지 확인하고, 맞다면 이름과 전화번호를 입력받게 소스코드를 작성하였다.

search부분은 입력받은 이름으로 저장된 정보를 확인하는 함수인데, for문으로 입력받은 이름과 일치하는 정보를 strcmp함수로 확인하고, 있다면 그 이름에 일치하는 정보를 출력하고 없다면 없다는 문구를 출력한다. 메인함수의 returnsize findsize minus plus는 각각

static size를 메인함수의 Size로 옮기면서 인덱스 값을 변화 시키거나 Size를 다시 size로 옮기는 함수들이다.

1. Makefile 설명

1 CC = gcc // gcc사용

2 CFLAG = -std=c11 // c11사용

3 OBJECTFILE= delete.o print.o register.o search.o phoneBookMain.o

4 all: phoneBookMain // 시작점

5

6 %.o:%.c // 모든 c파일을 오브젝트파일로 만든다.

7 $(CC) -c $(CFLAG) $<

8 phoneBookMain: $(OBJECTFILE) // phoneBookMain이란 이름의 실행파일을 만든다.

9 $(CC) -o phoneBookMain $(OBJECTFILE)

10 clean: // phoneBookMain실행파일과 오브젝트파일들을 지운다.

11 rm -f phoneBookMain $(OBJECTFILE)

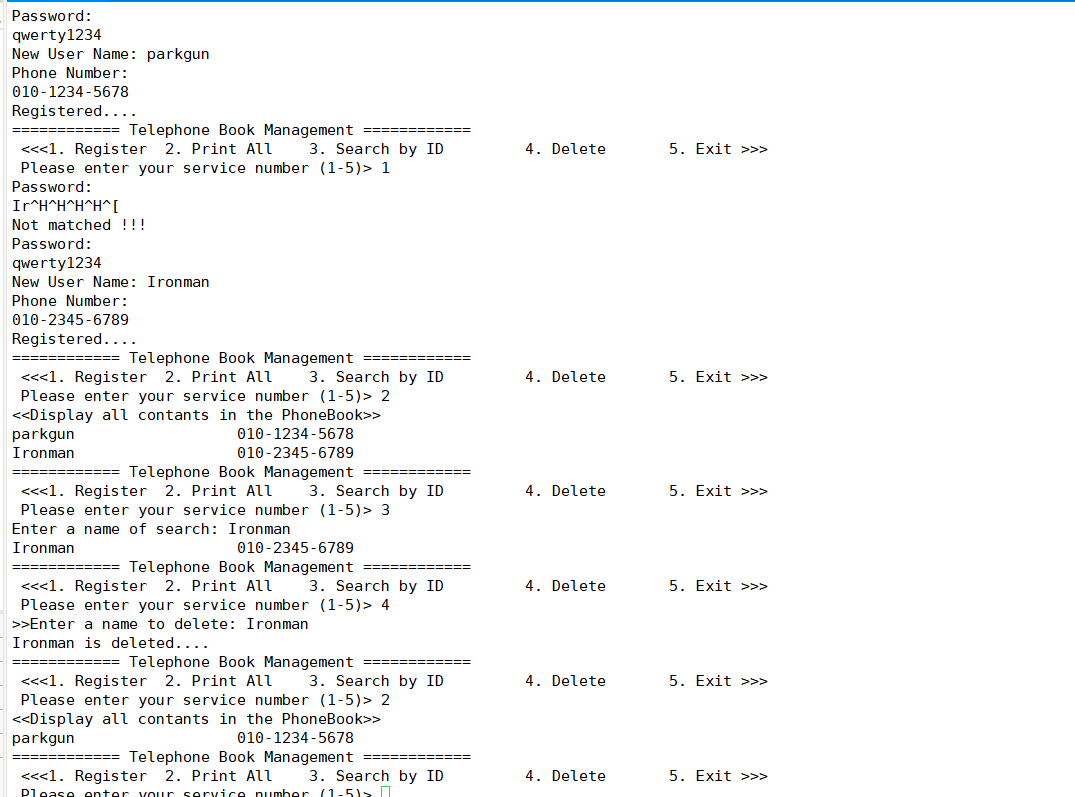
~

# 실행 방법 설명 (20점)

1. Unix에 gcc컴파일러를 사용하였습니다.
2. Gcc를 이용하여 컴파일 하였습니다.

Gcc -c 소스파일.c와 gcc -o 실행파일 이름 오브젝트 파일 명령어 사용

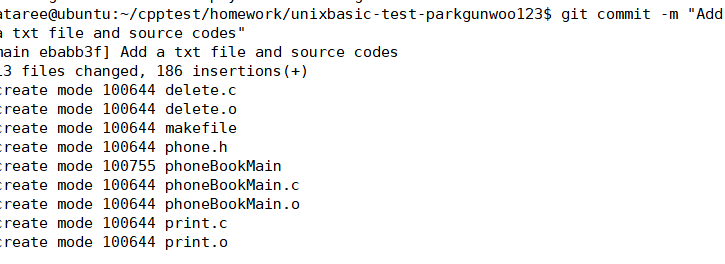
(3) 동작을 확인할 수 있는 실행 화면 캡처

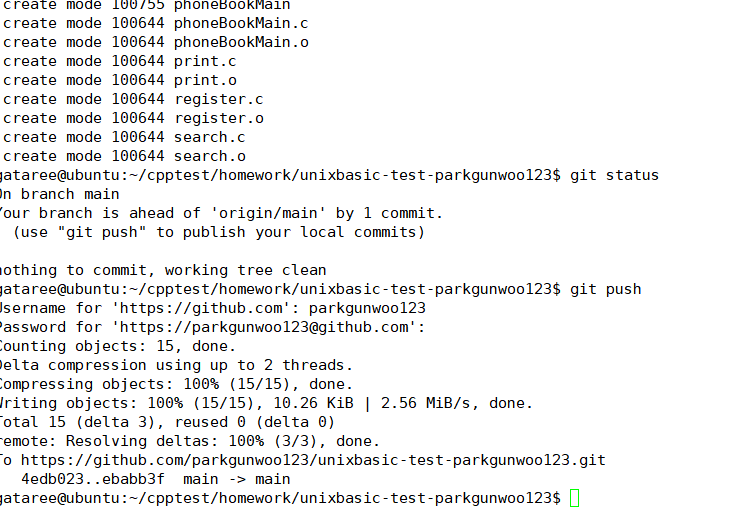


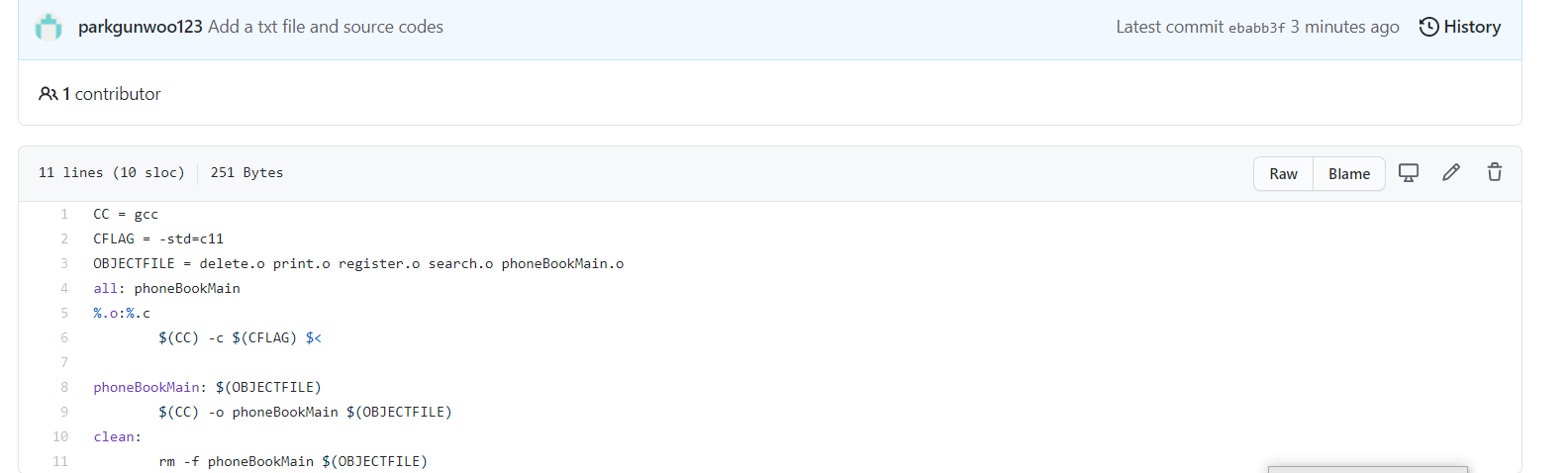
# Github 화면 (20점)

(1) cloning, adding, committing, push을 위한 github 명령들을 포함

(2) 소스 코드와 makefile을 push한 후, 본인의 Github repository를 스크린 캡쳐하여 포함







# 논의 사항 (10점)

- 숙제를 하면서 새로이 알게 된 내용(수업 시간 이외의 내용),

- 숙제를 하는 중에 어려웠던 점 등을 기술

Github를 처음 사용해보았는데, 사용하기가 조금 복잡하였지만 이제부터 명령들을 자주 사용하면서 과제를 풀어나간다면 발전된 프로그래밍기술을 습득할 수 있을 것 같다.

Extern이 외부에 선언된 변수를 사용할 수 있게 해주는 것을 알았다.