

HW5. 구조체 배열

부산대학교 정보컴퓨터공학부

2018-12345

홍길동

Github ID: 본인의 github 작성

제출일: 2020-06-01

1. 구현 내용에 대한 설명 (50점)

(1) 주요 변수 설명:

Extern : for문을 위한 int 변수 ij=0, 입력하는 이름을 받기위한 변수 nametmp[10]

Static : switch문을 몇번 실행했는지 확인

Name[10] : 이름이 입력될 구조체 변수

PhoneNumber[13] : 번호가 입력될 구조체 변수

Size : 구조체 배열의 실제크기

(2) 주요 자료 구조 설명

Register.c : 구조체 배열이 모두 차있지 않으면 이름과 전화번호를 입력받아 구조체 멤버 변수에 이름과 번호를 저장한다

Search.c : 구조체 배열의 처음부터 끝까지 입력받은 이름과 같은 이름이 존재하면 이름과 전화번호를 출력한다

Print.c : 구조체 배열에 입력된 값을 처음부터 끝까지 출력한다

Delete.c : 지우고 싶은 이름을 입력해 그에 해당하는 이름과 전화번호를 0으로 초기화한다

Phonebookmain.c : switch문을 이용해 해당하는 함수를 실행한다

(3) 주요 함수 구현 방법 설명

`void registerPhoneData(struct Contact *PhoneBook)` : size크기를 확인한 후 자리가 있으면 이름과 전화번호 입력 후 구조체 배열에 저장. 저장 후 size크기를 한번 늘린다.

`void printAll(struct Contact PhoneBook)` : 구조체 배열0부터 size까지 이름과 전화번호를 프린트한다

`void searchByName(struct Contact *PhoneBook)` : 이름을 nametmp에 입력받고 구조체 배열 0부터 size까지 비교 후 같은 이름이 존재하면 이름과 전화번호를 프린트한다

`void deleteByName(struct Contact *PhoneBook)` : 이름을 nametmp에 입력받고 Name에 같은 이름이 존재하면 memset함수를 이용해 해당 배열을 0으로 초기화한다.

(4) Makefile 설명

cc = gcc

CFAGS = -W -Wall

TARGET = PhoneBookMain_exe

OBJECTS = delete.o print.o register.o search.o phoneBookMain.o

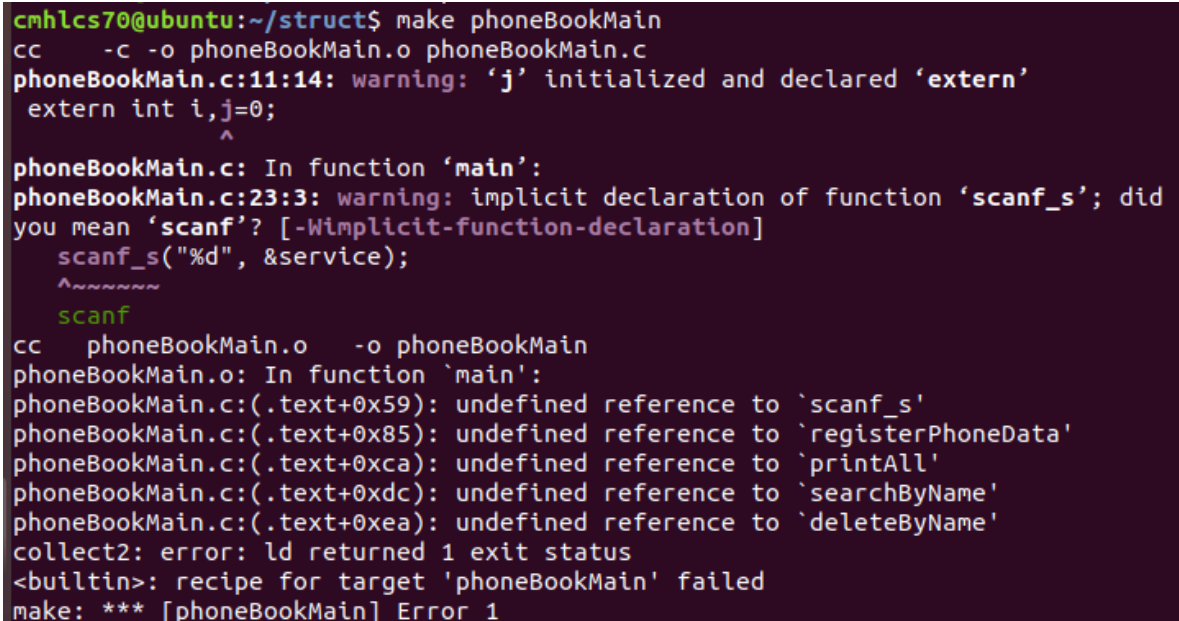
```
all : $(TARGET)
$(TARGET) : $(OBJECTS)
    $(CC) $(CFLAGS) -o $@ $^

clean :
rm *.o PhoneBookMain_exe
```

objects는 모든 .c파일을 .o파일로 만든다
all을 이용해 한번에 컴파일. 실행파일의 이름은 PhoneBookMain이 되도록 했다.

2. 실행 방법 설명 (20점)

- (1) 사용한 운영체제 및 컴파일러의 종류 : 리눅스, gcc
- (2) 컴파일 방법 및 실행 방법 : struct directory에 각 헤더파일, c파일, makefile을 넣은 후
make phoneBookMain 명령어 실행
- (3) 동작을 확인할 수 있는 실행 화면 캡처



```
cmhlcs70@ubuntu:~/struct$ make phoneBookMain
cc -c -o phoneBookMain.o phoneBookMain.c
phoneBookMain.c:11:14: warning: 'j' initialized and declared 'extern'
extern int i,j=0;
               ^
phoneBookMain.c: In function 'main':
phoneBookMain.c:23:3: warning: implicit declaration of function 'scanf_s'; did
you mean 'scanf'? [-Wimplicit-function-declaration]
scanf_s("%d", &service);
    ^~~~~~
scanf
cc phoneBookMain.o -o phoneBookMain
phoneBookMain.o: In function `main':
phoneBookMain.c:(.text+0x59): undefined reference to `scanf_s'
phoneBookMain.c:(.text+0x85): undefined reference to `registerPhoneData'
phoneBookMain.c:(.text+0xca): undefined reference to `printAll'
phoneBookMain.c:(.text+0xdc): undefined reference to `searchByName'
phoneBookMain.c:(.text+0xea): undefined reference to `deleteByName'
collect2: error: ld returned 1 exit status
<built-in>: recipe for target 'phoneBookMain' failed
make: *** [phoneBookMain] Error 1
```

에러가 발생해 실행하지 못했다.

3. Github 화면 (20점)

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

```
cmhlcs70@ubuntu:~/struct$ git config --global user.name "cmhlcs70"
cmhlcs70@ubuntu:~/struct$ git config --global user.email cmhlcs70@gmail.com
cmhlcs70@ubuntu:~/struct$ git add delete.c
cmhlcs70@ubuntu:~/struct$ git phoneBookMain.c
git: 'phoneBookMain.c' is not a git command. See 'git --help'.
cmhlcs70@ubuntu:~/struct$ git add phoneBookMain.c
cmhlcs70@ubuntu:~/struct$ git add print.c
cmhlcs70@ubuntu:~/struct$ gid add register.c
```

Command 'gid' not found, but can be installed with:

```
sudo apt install id-utils
```

```
cmhlcs70@ubuntu:~/struct$ git add register.c
cmhlcs70@ubuntu:~/struct$ git add search.c
cmhlcs70@ubuntu:~/struct$ git add phone.h
cmhlcs70@ubuntu:~/struct$ git commit -m "Add delete.c
> "
[master (root-commit) 0659e11] Add delete.c
6 files changed, 120 insertions(+)
create mode 100644 struct/delete.c
create mode 100644 struct/phone.h
create mode 100644 struct/phoneBookMain.c
create mode 100644 struct/print.c
create mode 100644 struct/register.c
create mode 100644 struct/search.c
```

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

```
cmhlcs70@ubuntu:~/struct$ git remote add origin https://github.com/cmhlcs70/test
cmhlcs70@ubuntu:~/struct$ git push origin master
Username for 'https://github.com': cmhlcs70
Password for 'https://cmhlcs70@github.com':
remote: Invalid username or password.
fatal: Authentication failed for 'https://github.com/cmhlcs70/test/'
cmhlcs70@ubuntu:~/struct$ git push origin master
Username for 'https://github.com': cmhlcs70
Password for 'https://cmhlcs70@github.com':
Counting objects: 9, done.
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 2.05 KiB | 2.05 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0)
To https://github.com/cmhlcs70/test
 * [new branch]      master -> master
cmhlcs70@ubuntu:~/struct$ git add Makefile
cmhlcs70@ubuntu:~/struct$ git push origin master
Username for 'https://github.com': cmhlcs70@gmail.com
Password for 'https://cmhlcs70@gmail.com@github.com':
Everything up-to-date
cmhlcs70@ubuntu:~/struct$
```

cmhlcs70 / test

Unwatch 1
Star 0
Fork 0

Code
Issues 0
Pull requests 0
Actions
Projects 0
Wiki
Security 0
Insights
Settings

Branch: master test / struct /

Create new file
Upload files
Find file
History

cmhlcs70 Add delete.c Latest commit 0659e11 9분 전

..		
delete.c	Add delete.c	9분 전
phone.h	Add delete.c	9분 전
phoneBookMain.c	Add delete.c	9분 전
print.c	Add delete.c	9분 전
register.c	Add delete.c	9분 전
search.c	Add delete.c	9분 전

4. 논의 사항 (10점)

구조체 배열의 사용법에 대한 복습

Github에 올리기 위해 파일이 있는 디렉토리에서 작업해야 하는데 홈디렉토리에서 작업해 오류가 났다. 디렉토리를 바꾸니 정상적으로 수행됨.