

HW6. 함수 포인터와 void 포인터

부산대학교 환경공학과

2015-22505

권영규

Github ID: vkdrb2003@naver.com

제출일: 2020-06-05

1. 구현 내용에 대한 설명 (60점)

(1) 주요 변수 설명: 함수 포인터 설명

Hw6Main.c에서 pFunc[5]라는 포인터 배열을 사용하여 함수포인터를 사용하였다.

각 배열은 함수이름으로 초기화하여 우리가 원하는 숫자를 넣으면 해당 기능을 배열 인덱스로 입력받아서 해당 기능을 하는 함수를 부르는 형태이다.

Sort.c에서 typedef를 사용하여 함수포인터에 대한 별칭을 cmp, swap으로 선언하였다. 선언한 함수포인터는 sortPhoneBook의 매개변수로 사용되고, 매개변수로 사용되는 각각의 포인터에 대한 함수는 contactCmpr, contactSwap으로 상세히 구현한다.

(2) 주요 자료 구조 설명

자료구조는 phone.h에 선언된 Contact 구조체를 이용한다. Contact 구조체는 이름을 입력받을 char배열, 전화번호를 입력받을 PhoneNumber배열로 구성되고 구조체배열의 최대 사이즈를 50으로 정하여 50명 이하의 인원만 입력받을 수 있도록 하였다.

(3) 주요 함수 구현 방법 설명

registerPhoneDate()=먼저 패스워드를 입력받는다. 입력받은 패스워드가 정해진 패스워드와 일치하면, PhoneBook에 이름과 전화번호를 입력받을 수 있도록 인터페이스를 제공하고, 패스워드 입력에 실패할 경우 실패한 횟수를 세어서 횟수에 맞는 메시지를 출력한다. 다른 소스파일에 쉽게 사용할 수 있도록 전역변수로 등록한 사람수(total_people)를 만들어 놓는다.

Print()= register.c파일의 total_people 변수를 참고하여 현재 PhoneBook에 들어있는 이름과 전화번호를 차례로 출력한다.

Search()= 검색하고 싶은 이름을 입력받으면 그 이름이 PhoneBook안에 이름과 맞는지 비교한다. Phonebook의 i번째 사람의 이름과 입력받은 이름을 한글자씩 비교해야 하기 때문에 2중for문을 사용한다. is_found는 전화번호부 사람수, is_match는 현재 비교대상끼리 일치하는지여부이다. i번째 사람과 입력받은 이름이 일치하지 않으면 is_found를 1씩 줄여나간다. 일치하는 이름이 있으면 for문을 탈출할 것이므로 is_found는 0이 될 수 없고, 일치하는 이름이 출력될 것이고, 일치하는 이름이 없으면 for문을 전부 순회하고 is_found는 0이 되어서 이름이 없다는 메시지가 출력된다.

deleteByName()= Search()함수와 거의 동일하게 구현했다. 차이점은 search함수는 일치하는 이름이 있으면 출력하는 기능이고, delete함수는 해당 이름과 전화번호를 지우고 뒤부터 한칸씩 앞으로 땡겨오면 된다.

contactCmpr함수= i번째 이름과 j번째 이름을 사전순으로 비교하는 함수이다. Strcmp를

사용하는데 strcmp(a,b) 에서 a가 b보다 사전적으로 뒤에 있으면 1을 반환한다.

contactSwap함수= i번째, j번째 이름, 전화번호를 바꾸는 함수이다. 임시 저장소로 구조체 tmp를 선언해서 i번째 정보를 tmp에, j번째 정보를 i에, tmp정보를 j로 넣어주었다.

sortPhoneBook함수= 함수포인터 cfn, sfn을 매개변수로 받는다. Cfn이 contactCmpr, sfn이 contactSwap함수에 대응된다. 구현할 때 cfn, sfn으로 사용하고, 사용할때, contactCmpr, contactSwap으로 넘겨주면, 함수가 저장된 주소로 가서 해당 함수를 실행한다. PhoneBook의 이름을 차례로 순회하되 한번 순회하고나서 개수를 한 개 줄여 다시 순회하는 과정을 사람수만큼 진행한다. 현재 이름과 뒤 이름이 바뀌어야되면, Swap함수를 실행하고, 아니면 그냥 넘어간다.

Sort()= print()함수를 사용한다. sortPhoneBook함수를 실행해서 정렬시키고 다시 print하여 정렬 전후로 비교할 수 있도록 하였다.

- Makefile 설명

CC = gcc

CFLAGS = -g -Wall

OBJS = print.o register.o search.o delete.o sort.o hw6Main.o

all : main

%.o : %.c

gcc -c -o \$@ \$(CFLAGS) \$<

main : \$(OBJS)

gcc -o main \$(OBJS)

clean:

rm -f main \$(OBJS)

각 소스파일을 차례로 컴파일하고 만들어진 오브젝트를 모아서 main으로 만든다. 나중에 충돌이 일어나지 않도록 clean으로 만들어진 오브젝트파일과 main을 지운다.

2. 실행 방법 설명 (10점)

(1) 사용한 운영체제 및 컴파일러의 종류

우분투 64bit, gcc컴파일러사용

(2) 컴파일 방법 및 실행 방법

makefile매크로를 사용하여 최종 output파일을 바로 생성할 수 있다. \$make clean으로 먼저 기존의 오브젝트파일을 지워주고, \$make 명령으로 생성 ./main 으로 output파일을 실행할 수 있다.

(3) 동작을 확인할 수 있는 실행 화면 캡처

```

vkdrb2003@ubuntu:~/homework$ clear

vkdrb2003@ubuntu:~/homework$ make clean
rm -f main print.o register.o search.o delete.o sort.o hw6Main.o
vkdrb2003@ubuntu:~/homework$ make
gcc -c -o print.o -g -Wall print.c
print.c: In function 'print':
print.c:14:37: warning: comparison with string literal results in unspecified behavior [-Waddress]
    if( (((Contact*)book)->Name)=="" && (((Contact*)book)->PhoneNumber)=="") ) break;
                                   ^~
print.c:14:76: warning: comparison with string literal results in unspecified behavior [-Waddress]
    if( (((Contact*)book)->Name)=="" && (((Contact*)book)->PhoneNumber)=="") ) break;
                                   ^~
gcc -c -o register.o -g -Wall register.c
gcc -c -o search.o -g -Wall search.c
gcc -c -o delete.o -g -Wall delete.c
gcc -c -o sort.o -g -Wall sort.c
gcc -c -o hw6Main.o -g -Wall hw6Main.c
hw6Main.c:12:12: warning: 'count_service' defined but not used [-Wunused-variable]
    static int count_service = 0;
               ^
vkdrb2003@ubuntu:~/homework$ gcc -o main print.o register.o search.o delete.o sort.o hw6Main.o
vkdrb2003@ubuntu:~/homework$ ./main
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 1
Ask to type password
password:
qwerty1234
New User Name:kwon
PhoneNumber:01011111111
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 1
Ask to type password
password:
qwerty1234
New User Name:kim
PhoneNumber:01022222222
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 1
Ask to type password
password:
qwerty1234
New User Name:choi
PhoneNumber:01033333333
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 1
Ask to type password
password:
qwerty1234
New User Name:lee
PhoneNumber:01044444444
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 1
Ask to type password
password:
qwerty1234
New User Name:tom
PhoneNumber:01055555555
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 2
Print all contents in the PhoneBook
name:kwon      phone:01011111111
name:kim      phone:01022222222
name:choi     phone:01033333333
name:lee      phone:01044444444
name:tom      phone:01055555555
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 3
Enter a name to search:kang
Oops! kang is not in the PhoneBook.
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 3
Enter a name to search:choi
name:choi      phone:01033333333
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 4
Enter a name to search:kwon
kwon is deleted...

```

```

===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search by ID      4. Delete      5. Sort      6. Exit >>>
Please enter your service number (1-6)> 5
-----Before sorting-----
Print all contents in the PhoneBook
name:kim      phone:01022222222
name:choi     phone:01033333333
name:lee      phone:01044444444
name:tom      phone:01055555555

-----After sorting-----
kim
Print all contents in the PhoneBook
name:choi     phone:01033333333
name:kim      phone:01022222222
name:lee      phone:01044444444
name:tom      phone:01055555555

```

3. Github 화면 (20점)

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

```

vkdrb2003@ubuntu:~/homework$ git clone https://github.com/datalab-pnu/unix-hw6-YeonggyuKwon.git
Cloning into 'unix-hw6-YeonggyuKwon'...
Username for 'https://github.com': vkdrb2003@naver.com
Password for 'https://vkdrb2003@naver.com@github.com':
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 4 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), done.
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ ls
delete.c hw6Main.c makefile phone.h print.c README.md register.c search.c sort.c
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ touch yeonggyu_kwon_201522505.txt
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ git add *.c *.md *.h *.txt makefile
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   delete.c
    modified:   hw6Main.c
    new file:   makefile
    new file:   print.c
    new file:   register.c
    new file:   search.c
    new file:   sort.c
    new file:   yeonggyu_kwon_201522505.txt

vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ █
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ git commit -m "final"
[main 800603f] final
 8 files changed, 262 insertions(+), 193 deletions(-)
 create mode 100644 delete.c
 rewrite hw6Main.c (88%)
 create mode 100644 makefile
 create mode 100644 print.c
 create mode 100644 register.c
 create mode 100644 search.c
 create mode 100644 sort.c
 create mode 100644 yeonggyu_kwon_201522505.txt

```

```

vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$ git push origin main
Username for 'https://github.com': vkdrb2003@naver.com
Password for 'https://vkdrb2003@naver.com@github.com':
Counting objects: 10, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (10/10), 2.70 KiB | 2.70 MiB/s, done.
Total 10 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/datalab-pnu/unix-hw6-YeonggyuKwon.git
   552a0d1..800603f  main -> main
vkdrb2003@ubuntu:~/homework/unix-hw6-YeonggyuKwon$

```

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

data-lab-pnu / unix-hw6-YeonggyuKwon (Private)
generated from TouchofEvil/2021unixhw6

<> Code ⓘ Issues 🔗 Pull requests ⌚ Actions 📁 Projects ⓘ Security 📊 Insights ⚙ Settings

main 1 branch 0 tags

Go to file Add file Code

YeonggyuKwon final 800603f 5 minutes ago 2 commits

README.md	Initial commit	2 days ago
delete.c	final	5 minutes ago
hw6Main.c	final	5 minutes ago
makefile	final	5 minutes ago
phone.h	Initial commit	2 days ago
print.c	final	5 minutes ago
register.c	final	5 minutes ago
search.c	final	5 minutes ago
sort.c	final	5 minutes ago
yeonggyu_kwon_201522505.txt	final	5 minutes ago

*pdf보고서는 스크린캡처 후 마무리하여 다시 제출하였습니다.

4. 논의 사항 (10점)

Code blocks로 먼저 구현하고, 우분투에서 실행해보았는데, Code blocks에서는 필요한 소스파일을 전부 include해야 실행되는 반면에, 우분투에서는 include하면 오히려 에러가 나서 그것을 찾느라 힘들었습니다.