

## HW7. 자기 참조 구조체

학과: 정보컴퓨터 공학부

학과: 본인의 학번. 2017-24486

이름: 배준석

Github ID: Bae-JunSeok

제출일: 2020-06-16

보고서에는 다음의 내용을 포함할 것.

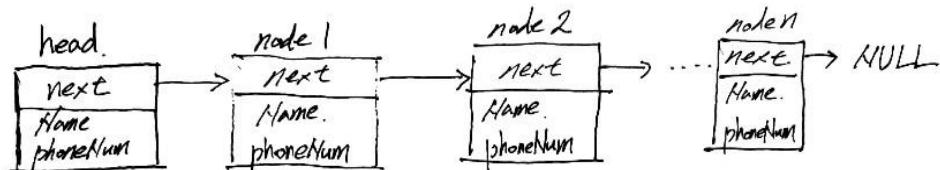
### 1. 구현 내용에 대한 설명 (60점)

#### (1) 주요 변수 설명

이번 과제는 hw6처럼 함수 포인터를 이용해서 함수를 호출하는 것이 아니라 hw5 처럼 함수 이름을 통해서 함수를 호출하였다. 주요 변수로는 선택을 하기 위해서 열거형을 정의했다. 2번 문항에서 자세히 설명하겠지만, 자기 참조 구조체(linked list)를 사용하기 위해서 이름과 휴대폰 번호를 저장할 수 있는 char 배열 2개와 다음 노드의 주소를 저장할 포인터를 멤버로 가지는 구조체를 생성한다. Linked list의 처음과 끝을 표현하기 위한 head와 tail도 있다. 또한 printArray, sortPhoneBook에서 사용하기 위한 구조체 배열도 선언한다.

#### (2) 주요 자료 구조 설명

- 자기 참조 구조체를 이용한 연결 리스트



지난 번까지의 과제들이 구조체 배열을 사용했다면, 이번 과제에서는 자기 참조 구조체(linked list)를 사용하였다.

Linked list는 node와 link를 구조화 시킨 것입니다. node는 데이터를 담는 그릇이고, link는 리스트의 순서를 유지할 수 있게 해주는 연결고리입니다.

배열과 연결 리스트의 차이는 배열은 배열 선언할 때 저장할 공간(메모리)를 미리 할당해야하지만, 리스트는 필요할 때마다(동적으로) 줄였다 늘였다 할 수 있다는 점입니다.(malloc과 free를 통해서)

node에는 데이터를 담고 link는 연결고리라고 했는데, 어떻게 연결을 시킬까? 그 방법은 바로 다음을 가리키는 포인터를 갖는 것이다. 위의 그림처럼 linked list는 다음을 가리키는 포인터를 갖고 있으며, 그를 통해서 linked list에 접근할 수 있다.

#### (3) 주요 함수 구현 방법 설명

주요 함수는 registerPhoneData, printAll, printArray, searchByName, deleteByName, sort, sortPhoneBook, 마지막으로 main 함수가 있다.

main 함수는 Exit를 선택하지 않는 이상 프로그램이 계속 실행되게끔 while문을 사용하였으며, 그 안에 switch 문을 통해서 어떤 기능을 수행할지 선택할 수 있도록 구현하였다.

registerPhoneData 함수에서 비밀번호와 관련한 기능은 HW5, 6과 동일하게 구현하였다. 이름과 전화번호를 저장하는 방식이 바뀌었는데, 이름과 전화번호 저장을 위한 배열을 선언하고, scanf를 통해서 입력한다. 이 후에 malloc을 통해서 구조체를 동적 할당하고, strcpy 함수를 통해서 배열에 입력된 값을 저장한다. 나중에 printArray 함수와 sortPhoneBook 함수 사용을 위해서 마찬가지로 구조체 배열의 요소에 strcpy 함수를 통해서 값을 저장한다.

printAll 함수는 printArray 함수를 호출하는 기능을 가진다.

printArray 매개변수로 void\* PhoneBook과 int size를 가진다. Contact\*를 사용해서 형 변환을 한 수 후에 값을 참조해서 이름과 전화번호를 출력한다.

searchByName 함수는 찾고자 하는 이름을 입력 받을 배열을 선언하고, scanf 함수를 통해서 값을 입력한다. 첫번째 노드를 가리키는 curr을 선언하고, curr이 끝을 가리킬 때 (curr != NULL) 까지 while문을 통해서 반복한다. 반복문 안에서 strcmp를 통해서 찾고자 하는 이름이 있는지 비교해서 같으면 break를 통해 반복문을 벗어나게 하고, 같지 않으면 curr이 다음 요소를 가리키게 한 후에 계속해서 반복한다. while문을 벗어났을 때, Curr이 끝을 가리키지 않으면 curr안에 저장되어 있는 이름과 전화번호를 출력하고, curr이 끝을 가리키면 이 이름은 전화번호부에 저장되어 있지 않다고 출력한다.

deleteByName 함수도 searchByName 함수와 마찬가지로 삭제할 이름을 찾는 과정을 비슷하게 구현하였다. 이번에는 첫번째 노드(head)를 가리키는 curr을 선언하고, 그 curr을 가리키는 previous를 선언했다. Curr에 저장된 이름과 삭제하고자 하는 이름이 같으면, previous->next = curr->next를 통해서 값을 당기고, free(curr)을 이용해서 메모리 공간을 해체하고, break문을 통해서 반복문을 빠져나간다. curr이 끝을 가리키면 (curr==NULL) 삭제하고자 하는 이름이 전화번호부에 저장되어 있지 않다고 출력한다.

sort 함수는 이름을 기준으로 오름차순 정렬을 하게 하는 sortPhoneBook 함수를 호출하고, 정렬되기 전과 정렬한 후의 결과를 출력하는 함수이다. 정렬되기 전에 printArray 함수를 호출해서, 정렬 전을 출력하고, sortPhoneBook의 호출을 통해서 버블 정렬을 시킨다. 이를 위해서 2가지 함수를 호출한다. 첫 번째로 contactCmpr 함수는 strcmp 함수를 통해서 구조체 배열에 저장된 이름을 비교하고, 앞에 저장된 이름의 아스키 코드 값이 클 경우에 1을 반환함으로써 contactSwap 함수를 호출한다. contactSwap 함수는 값을 바꾸는 함수로써 임시로 값을 저장할 구조체 변수 temp를 선언하고, temp를 통해서 값을 바꾼다. 이 두 함수는 void\* PhoneBook과 int i, int j를 매개변수로 가진다. i와 j는 배열의 몇 번째 요소를 비교할지를 위해 사용된다. 이 때 void형 포인터를 통해서 배열의 요소에 접근이 불가능하기 때문에 Contact\*를 사용해서 형변환 후에 값을 참조한다.

#### (4) Makefile 설명

Suffix rule(확장자 규칙)과 매크로 활용을 통해서 c 파일들(register.c, print.c, search.c, delete.c, sort.c, hw7Main.c)을 object 파일로 컴파일 되도록 한다. 만들어진 object 파일들을 링크해서 실행파일 phonebook이 생성되도록 문장을 구성한다.

```
.SUFFIXES = .c .o
$(CC) = gcc
OBJECTS = register.o print.o search.o delete.o sort.o hw7Main.o
CFLAGS = -Wall -g
TARGET = phonebook

phonebook : $(OBJECTS)
    $(CC) -o $(TARGET) $(OBJECTS)

%.o : %.c
    $(CC) -c -o $@ $(CFLAGS) $<

clean :
    rm -f $(OBJECTS) $(TARGET)
```

## 2. 실행 방법 설명 (10점)

- (1) 사용한 운영체제 및 컴파일러의 종류 Ubuntu 18.04.4, gcc compiler
- (2) 컴파일 방법 및 실행 방법  
Makefile을 이용해서 c 파일들을 object 파일로 만들고, 만들어진 object 파일들을 링크하여 실행파일을 생성한다.
- (3) 숙제에서 제시한 동작들이 제대로 동작한다는 것을 확인할 수 있도록, 프로그램을 실행하고 그 결과를 캡처하며 보고서에 포함함

```

seok@ubuntu:~/HW7$ make
cc -c -o register.o -Wall -g register.c
cc -c -o print.o -Wall -g print.c
cc -c -o search.o -Wall -g search.c
cc -c -o delete.o -Wall -g delete.c
cc -c -o sort.o -Wall -g sort.c
cc -c -o hw7Main.o -Wall -g hw7Main.c
cc -o phonebook register.o print.o search.o delete.o sort.o hw7Main.o
seok@ubuntu:~/HW7$ ./phonebook
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 1
Password:
1234
New User Name: aaa
PhoneNumber: 01011111111
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 1
Password:
1234
New User Name: bbb
PhoneNumber: 01022222222
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 1
Password:
1234
New User Name: ccc
PhoneNumber: 01033333333
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 1
Password:
1234
New User Name: ddd
PhoneNumber: 01044444444
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 2
Print all contents int the PhoneBook
name : aaa      phone : 01011111111
name : bbb      phone : 01022222222
name : ccc      phone : 01033333333
name : ddd      phone : 01044444444
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 1
Password:
1234
New User Name: zzz
PhoneNumber: 01055555555
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 1
Password:
1234
New User Name: ggg
PhoneNumber: 01099999999
Registered...
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 2
Print all contents int the PhoneBook
name : aaa      phone : 01011111111
name : bbb      phone : 01022222222
name : ccc      phone : 01033333333
name : ddd      phone : 01044444444
name : zzz      phone : 01055555555
name : ggg      phone : 01099999999
===== Telephone Book Management =====
<<<1. Register  2. Print All  3. Search      4. Delete      5. Sort 6. Exit>>>
Please enter your service number (1-5)> 3
>>Enter a name to search: a
Oops! a is not in the PhoneBook.

```

```

===== Telophone Book Management =====
<<<1. Register  2. Print All  3. Search  4. Delete  5. Sort 6. Exit>>>
Please enter your service number (1-5)> 3
>>Enter a name to search: a
Oops! a is not in the PhoneBook.
===== Telophone Book Management =====
<<<1. Register  2. Print All  3. Search  4. Delete  5. Sort 6. Exit>>>
Please enter your service number (1-5)> 3
>>Enter a name to search: aaa
aaa
01011111111
===== Telophone Book Management =====
<<<1. Register  2. Print All  3. Search  4. Delete  5. Sort 6. Exit>>>
Please enter your service number (1-5)> 4
>>Enter a name to delete: e
Oops! e is not in the PhoneBook.
===== Telophone Book Management =====
<<<1. Register  2. Print All  3. Search  4. Delete  5. Sort 6. Exit>>>
Please enter your service number (1-5)> 4
>>Enter a name to delete: aaa
aaa is deleted...
===== Telophone Book Management =====
<<<1. Register  2. Print All  3. Search  4. Delete  5. Sort 6. Exit>>>
Please enter your service number (1-5)> 5
Sort function is called
Before sorting----
name : aaa      phone : 01011111111
name : bbb      phone : 01022222222
name : ccc      phone : 01033333333
name : ddd      phone : 01044444444
name : zzz      phone : 01055555555
name : ggg      phone : 01099999999

After sorting----
name : aaa      phone : 01011111111
name : bbb      phone : 01022222222
name : ccc      phone : 01033333333
name : ddd      phone : 01044444444
name : ggg      phone : 01099999999
name : zzz      phone : 01055555555

===== Telophone Book Management =====
<<<1. Register  2. Print All  3. Search  4. Delete  5. Sort 6. Exit>>>
Please enter your service number (1-5)> 6

```

### 3. Github 화면 (20점)

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



```

seok@ubuntu:~/HW7$ ls
delete.c  hw7Main.c  makefile      phonebook  print.c  register.c  search.c  sort.c
delete.o  hw7Main.o  osp2020-hw7-Bae-JunSeok  phone.h    print.o  register.o  search.o  sort.o
seok@ubuntu:~/HW7$ touch junseok_bae_201724486.txt
seok@ubuntu:~/HW7$ ls
delete.c  hw7Main.o  osp2020-hw7-Bae-JunSeok  print.c  register.o  sort.c
delete.o  junseok_bae_201724486.txt  phonebook  print.o  search.c  sort.o
hw7Main.c  makefile      phone.h      register.c  search.o
seok@ubuntu:~/HW7$ git init
Initialized empty Git repository in /home/seok/HW7/.git/
seok@ubuntu:~/HW7$ git init
Reinitialized existing Git repository in /home/seok/HW7/.git/
seok@ubuntu:~/HW7$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        delete.c
        delete.o
        hw7Main.c
        hw7Main.o
        junseok_bae_201724486.txt
        makefile
        osp2020-hw7-Bae-JunSeok/
        phone.h
        phonebook
        print.c
        print.o
        register.c
        register.o
        search.c
        search.o
        sort.c
        sort.o

```

```

seok@ubuntu:~/HW7$ git add .
warning: adding embedded git repository: osp2020-hw7-Bae-JunSeok
hint: You've added another git repository inside your current repository.
hint: Clones of the outer repository will not contain the contents of
hint: the embedded repository and will not know how to obtain it.
hint: If you meant to add a submodule, use:
hint:
hint:   git submodule add <url> osp2020-hw7-Bae-JunSeok
hint:
hint: If you added this path by mistake, you can remove it from the
hint: index with:
hint:
hint:   git rm --cached osp2020-hw7-Bae-JunSeok
hint:
hint: See "git help submodule" for more information.
seok@ubuntu:~/HW7$ git add.
git: 'add.' is not a git command. See 'git --help'.

The most similar command is
add
seok@ubuntu:~/HW7$ git add .
seok@ubuntu:~/HW7$ git commit -m "201724486_Bae_JunSeok_HW7"
[master (root-commit) ac73dbd] 201724486_Bae_JunSeok_HW7
17 files changed, 266 insertions(+)
create mode 100644 delete.c
create mode 100644 delete.o
create mode 100644 hw7Main.c
create mode 100644 hw7Main.o
create mode 100644 junseok_bae_201724486.txt
create mode 100644 makefile
create mode 160000 osp2020-hw7-Bae-JunSeok
create mode 100644 phone.h
create mode 100755 phonebook
create mode 100644 print.c
create mode 100644 print.o
create mode 100644 register.c
create mode 100644 register.o
create mode 100644 search.c
create mode 100644 search.o
create mode 100644 sort.c
create mode 100644 sort.o
seok@ubuntu:~/HW7$ git remote add origin https://github.com/Bae-JunSeok/201724486_Bae_JunSeok_HW7.git

```

```
seok@ubuntu:~/HW7$ git remote add origin https://github.com/Bae-JunSeok/201724486_Bae_JunSeok_HW7.git
seok@ubuntu:~/HW7$ git push origin master
Username for 'https://github.com': Bae-JunSeok
Password for 'https://Bae-JunSeok@github.com':
Counting objects: 18, done.
Compressing objects: 100% (17/17), done.
Writing objects: 100% (18/18), 24.65 KiB | 2.74 MiB/s, done.
Total 18 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/Bae-JunSeok/201724486_Bae_JunSeok_HW7.git
 * [new branch]      master -> master
```

## 4. 논의 사항 (10점)

- 숙제를 하면서 새로이 알게 된 내용(수업 시간 이외의 내용),
- 숙제를 하는 중에 어려웠던 점 등을 기술

새로운 자료형을 이용해서 프로그램 작성하는 것이 어려웠다. 특히 연결 리스트의 연결 관계 부분이 어려웠다.