

객체지향프로그래밍 및 실습

OOP 2-3 과제

학과 : 컴퓨터정보공학부

학번 : 2021202043

이름 : 이은서

제출일 : 2022.04.27

문제 1.

1. 문제 설명

private으로 선언한 int type 멤버변수를 가지는 클래스를 구현한다. 생성자와 소멸자의 함수 내용으로 각각 저장된 멤버변수를 출력하는 코드를 넣으면 멤버변수에 접근할 수 있다.

2. 결과 화면

```
"/Users/ieunseo/Downloads/practice C++/cmake-build-debug/2-3-1"
Enter Any Number :18
Class A is Created with 18
Class A is Destroyed with 18

종료 코드 0(으)로 완료된 프로세스
```

3. 고찰

생성자와 소멸자에 대한 개념 이해가 부족하여 처음에 조금 헤맸었다. 생성자와 소멸자는 type 없이 class가 사용될 때 실행이 된다는 것을 알았다.

문제 2.

1. 문제 설명

첫 문자열을 입력 받고, 그 문자열을 가지고 길이를 구하거나 다른 문자열로 교체하거나, 붙여넣거나 출력하게끔 하는 명령어를 입력받아서 실행하는 프로그램을 구현한다.

2. 결과 화면

```
"/Users/ieunseo/Downloads/practice C++/cmake-build-debug/2-3-2"
First String : happy
Please Enter Command(strlen, strcpy, strcat, print, quit) :strlen
String Length :5
Please Enter Command(strlen, strcpy, strcat, print, quit) :strcpy hello
strcpy is done
Please Enter Command(strlen, strcpy, strcat, print, quit) :print
Current String : hello
Please Enter Command(strlen, strcpy, strcat, print, quit) :strcat myfriends
strcat is done
Please Enter Command(strlen, strcpy, strcat, print, quit) :strlen
String Length :14
Please Enter Command(strlen, strcpy, strcat, print, quit) :print
Current String : hellomyfriends
Please Enter Command(strlen, strcpy, strcat, print, quit) :quit
```

3. 고찰

생성자에서 문자열을 입력받아서 각각의 함수마다 적절한 코드를 구성하였고, 필요한 코드는 지난 과제에서 했던 코드들을 참고하여 직접 구현하였다. strtok 함수의 경우에는 '/'가 아닌 '공백'을 기준으로 문자열을 잘라서 따로 만든 문자열에 저장을 하고, 이를 이용해 정보를 저장하는 식으로 구현하였다.

문제 3.

1. 문제 설명

Student info를 저장하는 클래스를 구현하여 최대 10명의 학생을 저장, 탐색, 변경, 출력하는 프로그램을 설계하는 것이다. strcmp, strtok, stoi 등의 함수를 사용하여 명령어를 나누고 각 명령어에 맞는 클래스 멤버함수가 실행되게 하여 프로그램을 구현했다.

2. 결과 화면

```
practice C++ · OOP Assignment · 2-3 · 2-3-3.cpp
실행: 2-3-4 x 2-3-2 x 2-3-3 x
"/Users/ieunseo/Downloads/practice C++/cmake-build-debug/2-3-3"
Please Enter Command(insert, find, change, print, exit) :insert 이은서 21 광운대 컴공과
Please Enter Command(insert, find, change, print, exit) :insert 홍길동 24 광운대 법학과
Please Enter Command(insert, find, change, print, exit) :insert 고길동 28 광운대 경영학과
Please Enter Command(insert, find, change, print, exit) :print
-----print-----
Name : 이은서
Age : 21
University : 광운대
Major : 컴공과
-----
Name : 홍길동
Age : 24
University : 광운대
Major : 법학과
-----
Name : 고길동
Age : 28
University : 광운대
Major : 경영학과
-----
Please Enter Command(insert, find, change, print, exit) :find 이은서
-----find-----
Name : 이은서
Age : 21
University : 광운대
Major : 컴공과
-----
Please Enter Command(insert, find, change, print, exit) :change 홍길동 고길동 26 광운대 국문과
Please Enter Command(insert, find, change, print, exit) :print
-----print-----
Name : 이은서
Age : 21
University : 광운대
Major : 컴공과
-----
Name : 홍길동
Age : 26
University : 광운대
Major : 국문과
-----
Name : 고길동
Age : 28
University : 광운대
Major : 경영학과
-----
Please Enter Command(insert, find, change, print, exit) :exit
종료 코드 0(으)로 완료된 프로세스
```

3. 고찰

멤버변수와 멤버함수의 매개변수로 들어간 변수의 이름이 같아서 어떻게 정의를 해야하는지 고민을 많이 하였다. 그러던 와중 this의 사용이 가능하다는 것을 깨닫고 this->age 이런 식으로 멤버변수와 매개변수의 이름을 구분하여 정의할 수 있었다. 또, change 명령어에서 해맸는데, find에서 사용한 코드를 이용하여 저장된 모든 클래스의 이름을 각각 비교하여 같은 경우의 index에 저장된 정보를 바꾸어 주고 바로 break가 되게 하여, 이름이 같은 경우의 충돌 문제를 해결하였다.

문제 4.

1. 문제 설명

동물 정보를 저장하고 출력하는 프로그램을 구현한다. animal class는 zoo class를 통해 관리하는데 각 class는 주어진 멤버변수만 사용이 가능하다. 멤버함수는 자유롭게 구현할 수 있다.

2. 결과 화면

```
Please Enter Command(new_animal, print_all, print_species, exit) : new_animal 백구 4 개
Please Enter Command(new_animal, print_all, print_species, exit) : new_animal 흑구 3 개
Please Enter Command(new_animal, print_all, print_species, exit) : new_animal 나비 8 고양이
Please Enter Command(new_animal, print_all, print_species, exit) : new_animal 만두 7 고양이
Please Enter Command(new_animal, print_all, print_species, exit) : new_animal 야옹이 6 고양이
Please Enter Command(new_animal, print_all, print_species, exit) : new_animal 상아 11 코끼리
Please Enter Command(new_animal, print_all, print_species, exit) : print_all
-----print_all-----
-----0-----
Name : 백구
Year : 4
Species : 개
-----
-----1-----
Name : 흑구
Year : 3
Species : 개
-----
-----2-----
Name : 나비
Year : 8
Species : 고양이
-----
-----3-----
Name : 만두
Year : 7
```

```
practice C++ · OOP Assignment · 2-3 · 2-3-4.cpp
실행: 2-3-4 x
Species : 고양이
-----3-----
Name : 만두
Year : 7
Species : 고양이
-----4-----
Name : 야옹이
Year : 6
Species : 고양이
-----5-----
Name : 삼아
Year : 11
Species : 코끼리
-----
Please Enter Command(new_animal, print_all, print_species, exit) : print_species 개
-----print_species-----
-----0-----
Name : 백구
Year : 4
Species : 개
-----1-----
Name : 흑구
Year : 3
Species : 개
Number of Species is 2
-----
Please Enter Command(new_animal, print_all, print_species, exit) : print_species 고양이
-----print_species-----
-----2-----
Name : 나비
Year : 8
Species : 고양이
-----3-----
Name : 만두
Year : 7
Species : 고양이
-----4-----
Name : 야옹이
Year : 6
Species : 고양이
-----
Number of Species is 3
-----
Please Enter Command(new_animal, print_all, print_species, exit) : exit
종료 코드 0(으)로 완료된 프로세스
```

3. 고찰

animal class에 friend class를 두어 animal class에서 private으로 선언한 변수에 대한 접근을 가능하게 했다. species가 같은 것들을 출력하기 위해 변수에 접근하는 과정에서 여러번 오류가 났었고, friend class에 대한 개념 이해도 부족해서 여러번 시행착오를 겪었지만, 모든 animal list의 species들을 각각 비교하여 같은 문자열이면 해당 번호의 정보가 저장된 부분을 출력하게끔 하여 성공하였다.