

«C++ 연습 과제»

■ 제3장 함수

학습활동1: 함수의 정의 및 호출

인수로 전달되는 int형 배열에 저장된 값 중 가장 큰 값을 구하는 함수를 정의하고, 이를 호출하여 사용하는 프로그램을 작성하라.

※ Check list

- 함수 정의 문장 및 호출 문장의 형식
- return 명령을 이용한 결과값 반환
- 함수 원형의 사용 - 함수가 호출 위치에 비해 뒤에 정의되어 있거나, 다른 파일에 정의되어 있을 때는 함수 원형을 선언해야 한다.

학습활동2: 참조 호출

교재 89쪽 [소스코드 3-4]의 SwapValues 함수를 다음과 같이 호출한 후 다음과 같이 호출해 본다.

```
SwapValues(a, a); // (가)
```

함수 SwapValues를 아래와 같이 수정한 후 위의 (가) 명령을 다시 실행하여 결과를 비교해 본다.

```
void SwapValues(int& x, int& y)
{
    x += y;
    y = x - y;
    x -= y;
}
```

※ Check list

- 수정된 함수는 SwapValues(a, b); 와 같은 형식으로 호출할 때는 문제가 없으나, (가)와 같이 호출할 경우 형식매개변수 x와 y가 동일한 변수의 참조가 됨으로 인해 잘못된 결과를 만든다.

학습활동3: 함수 다중정의

[학습활동1]의 함수를 float형 배열에 대하여 다중정의하라.

※ Check list

- 함수 다중정의를 할 때는 각각의 함수가 인수의 자료형이나 개수로 구분할 수 있어야 한다.

■ 제4장 클래스와 객체

학습활동4: 클래스의 선언

교재 121쪽의 [소스코드 4-6]에 하향계수(계수기 값을 1씩 감소시킴) 멤버함수 `countDown()`을 추가하고, 이를 사용하는 프로그램을 작성하라.

※ Check list

- ☐ 생성자 선언 방법을 익힌다.
- ☐ `const` 데이터멤버는 초기화 리스트를 통해서 초기화한다.
- ☐ `getValue()`에 지정된 `const`의 의미를 이해한다. (왜 생성자는 `const`로 지정하지 않는가?)

학습활동5: static 멤버의 활용

이름(string으로 표현)과 일련번호를 저장하는 객체를 정의할 수 있는 클래스 `S`를 선언하라. 일련번호는 프로그램 시작 후 현재 시점까지 생성된 `S` 클래스 객체 수이다.

※ Check list

- ☐ `static` 데이터멤버를 선언하는 방법
- ☐ `static` 데이터멤버를 일반 멤버함수에서 액세스할 수 있는가?
- ☐ 일반 데이터멤버를 `static` 멤버함수에서 액세스할 수 있는가?

■ 제5장 연산자 다중정의

학습활동6: 연산자 다중정의

1개의 `int`형 데이터 멤버로 구성되며, 전위표기 및 후위표기 `++` 연산자와 `+=` 연산자를 포함하는 클래스를 선언하라.

※ Check list

- ☐ 전위표기 및 후위표기 연산자의 다중정의 구문 형식
- ☐ 이항 연산자의 다중정의 구문 형식
- ☐ 다중정의를 피연산자와 형식매개변수의 관계
- ☐ 연산자 다중정의를 연산자의 의미 고려

■ 제6장 상속

학습활동7: 파생클래스의 선언

교재 231쪽 [소스코드 6-9] `Person` 클래스의 파생 클래스인 `Athlete`을 선언하라. 운동 선수를 나타내는 클래스인 `Athlete`에는 소속팀을 나타내는 데이터멤버 `team`을 포함하며, 팀 이름을 지정하거나 저장된 팀 이름을 알리는 기능, 콘솔에 “...팀 소속 선수 ...입니다”라는 형식으로 출력하는 기능(`print`)이 포함된다.

※ Check list

- ☐ 파생 클래스 선언문 형식
- ☐ 파생 클래스와 기초 클래스의 생성자
- ☐ 가시성 상속 지시어의 기능

학습활동8: 가상함수

학습활동7에서 선언한 Person 클래스의 포인터로 Person 객체와 Athlete 객체를 가리키게 한 후 print를 호출하여 본다. 그리고 Person 클래스를 수정하여 동일한 처리가 해당 클래스의 print를 호출하도록 한다.

※ Check list

- 가상함수를 통해 멤버함수의 동적연결을 구현한다.

학습활동9: 추상 클래스

도형을 나타내는 추상 클래스 Shape을 선언하라. Shape의 멤버함수에는 면적을 구하는 area()와 도형의 내용을 콘솔에 출력하는 print()가 포함되며, 이들은 순수 가상함수이다. 그리고 이의 파생 클래스인 Circle과 Rectangle을 선언하라. 이때 Circle과 Rectangle이 상세 클래스가 되도록 하라.

※ Check list

- 순수 가상함수의 선언 방법
- Circle이나 Rectangle이 상세 클래스가 되려면 어떻게 해야 하는가?
- Shape, Circle, Rectangle 클래스의 객체를 정의할 수 있는가?

■ 제7장 템플릿

학습활동10: 클래스 템플릿의 활용

지정된 자료형의 데이터를 지정된 개수만큼 저장할 수 있는 배열 형태의 컨테이너 클래스를 템플릿으로 선언하라. 이 클래스 템플릿에는 데이터를 액세스하기 위한 [] 연산자가 포함된다. 교재 168쪽의 Pencils 클래스의 객체를 5개 저장할 수 있는 컨테이너를 정의하여 사용하는 예를 프로그램으로 작성하라.

※ Check list

- 클래스 템플릿의 선언 형식
- 템플릿 인수의 활용
- [] 연산자의 다중정의

학습활동11: vector의 활용

입력된 정수형 데이터를 vector에 저장한다. 데이터의 개수는 정해져 있지 않으며, 음수가 입력되면 데이터의 끝으로 간주한다. vector 객체의 값 중에서 최대값을 찾는다. 반복은 반복자를 이용한다. vector 객체에 저장된 값들을 정렬한 후 순서대로 콘솔에 출력한다.

※ Check list

- vector 객체의 선언
- 반복자의 활용
- 알고리즘의 활용

■ 제8장 예외처리

학습활동12: 예외 객체의 활용

연, 월, 일을 입력하는 함수 getDate()를 정의하라. getDate()는 2000년 1월 1일보다 앞서는 날이 입력된 경우와 2100년 12월 31일이 넘는 날이 입력된 경우 각각을 구분하는 예외 객체를 던진다. 이 예외 객체들은 exception 클래스로부터 상속을 받으며, 입력된 날짜를 what()을 통해 알린다. 이 함수를 사용하는 예를 프로그램으로 작성하라.

※ Check list

- ☐ exception으로부터 상속을 받는 예외 객체의 선언
- ☐ try 및 catch 블록의 작성