

# 심화프로그래밍

실습강의 10주차

# 실습강의 소개

## ● 실습 진행 방법

- 간단한 이론 복습 및 해당주차 실습과제 설명
- 실습 후 보고서와 소스코드를 압축하여 **수요일 자정(23:59)까지**  
꼭!! 이클래스 제출(**이메일 제출 불가, 반드시 이클래스를 통해 제출**)
- 실습과제 제출기한 엄수(제출기한 이후로는 0점 처리)

## ● Q & A

- 이클래스 및 실습조교 이메일을 통해 질의응답
- **이메일 제목 : [심화프로그래밍\_홍길동] \*본인 과목명과 성명 꼭 작성!!**
- 실습조교 메일 주소 : [0hae@dgu.ac.kr](mailto:0hae@dgu.ac.kr)

# 실습 보고서 작성 방법 [1/2]

## ● 실습 보고서

- 문제 분석: 실습 문제에 대한 요구 사항 파악, 해결 방법 등 기술
- 프로그램 설계 및 알고리즘
  - 해결 방법에 따라 프로그램 설계 및 알고리즘 등 기술
  - e.g.) 문제 해결 과정 및 핵심 알고리즘 기술
- 소스코드 및 주석
  - 소스코드와 그에 해당하는 주석 첨부
  - 각각의 함수가 수행하는 작업, 매개변수, 반환 값 등을 명시
  - 소스코드 전체 첨부 (소스코드 화면 캡처X, 소스코드는 복사/붙여넣기로 첨부)
- 결과 및 결과 분석
  - 결과 화면을 캡처 하여 첨부, 해당 결과가 도출된 이유와 타당성 분석
- 소감
  - 실습 문제를 통해 습득할 수 있었던 지식, 느낀 점 등을 기술

# 실습 보고서 작성 방법 [2/2]

## ● 제출 방법

- 보고서, 소스코드를 1개의 파일로 압축하여 e-class “과제” 메뉴를 통해 제출
  - “이름학번실습주차.zip” 형태로 제출(e.g. :김동국19919876실습10.zip)
  - 파일명에 공백, 특수 문자 등 사용 금지

## ● 유의 사항

- 보고서의 표지에는 학과, 학번, 이름, 담당 교수님, 제출일자 반드시 작성
- 정해진 기한내 제출
  - 기한 넘기면 **0점** 처리
  - 이클래스가 과제 제출 마지막 날 오류로 동작하지 않을 수 있으므로, 최소 1~2일전에 제출
  - **과제 제출 당일 이클래스 오류로 인한 미제출은 불인정**
- 소스코드, 보고서를 자신이 작성하지 않은 경우 **실습 전체 점수 0점 처리**
- **Visual Studio 2019 또는 Sharstra 웹 IDE 기반 학습 프로그램 사용하여 실습 진행**

- 상속(Inheritance)

- 객체 지향 언어의 본질적인 특성이며, 상속 개념이 없다면 객체 지향 언어로 보기 어려움
- 소프트웨어의 재사용을 가능하게 해주고, 실행 도중에 동적 바인딩이 가능함

- 상속 관계

- 두 클래스(class) 사이에 **부모-자식**의 상속 관계를 선언함
- 부모 클래스를 **기본 클래스(base class)**, 자식 클래스를 **파생 클래스(derived class)**로 부름
  - 자바에서는 슈퍼 클래스(super class), 서브 클래스(sub class)
  - C#에서는 부모 클래스(parent class), 자식 클래스(child class)
- 여러 개의 클래스를 동시에 상속받은 **다중 상속(multiple inheritance)**을 허용함

## ● 상속의 목적과 장점

- 간결한 클래스 작성
  - 동일한 코드 중복 방지
- 클래스 간의 계층적 분류 및 관리의 용이함
  - 구조적인 관계 파악을 쉽게함
- 클래스 재사용과 확장을 통한 소프트웨어의 생산성 향상
  - 새로운 소프트웨어 출시나 업그레이드 주기가 가속화됨에 따라 상속을 통해 기능 확장이 필요함

## ● 상속 선언

- 상속은 class 선언 뒤에, **콜론(:)**과 **기본 클래스 이름**을 선언하여 사용 가능함
- 이때 기본 클래스 이름 앞에 반드시 **상속 접근(public, private, protected)**을 지정해야함

# 상속(Inheritance) [3/3]

- 상속 예제 소스

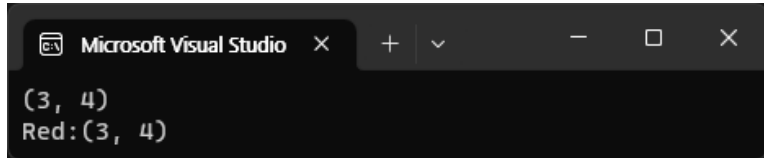
```
1. class Dog : public Animal {
2. public:
3.     Dog(const string& dogName) : Animal(dogName) {}
4.
5.     void makeSound() override {
6.         cout << name << ": 멍멍!" << endl;
7.     }
8. };
9.
10. class Cat : public Animal {
11. public:
12.     Cat(const string& catName) : Animal(catName) {}
13.
14.     void makeSound() override {
15.         cout << name << ": 야옹~" << endl;
16.     }
17. };
```

# 업 캐스팅(up-casting)

## ● 업 캐스팅

- 파생 클래스의 객체를 기본 클래스의 포인터로 가리키는 것을 의미함
- 파생 클래스의 객체를 기본 클래스의 객체처럼 다룰 수 있게 함

```
1. int main() {  
2.     ColorPoint cp;  
3.     ColorPoint *pDer = &cp;  
4.     Point* pBase = pDer; // 업 캐스팅  
5.  
6.     pDer->set(3, 4);  
7.     pBase->showPoint();  
8.     pDer->setColor("Red");  
9.     pDer->showColorPoint();  
10.    pBase->showColorPoint() // 컴파일 오류, Point의 멤버가 아님.  
11.}
```



- 업 캐스팅한 기본 클래스의 포인터로는 기본 클래스의 멤버만 접근 가능함
- 업 캐스팅 시 명시적 타입 변환은 필요없음



# 다운 캐스팅(down-casting)

- 다운 캐스팅

- 기본 클래스의 객체를 파생 클래스의 포인터로 가리키는 것을 의미함

```
1. int main() {  
2.     ColorPoint cp;  
3.     ColorPoint* pDer;  
4.     Point* pBase = &cp; // 업 캐스팅  
5.  
6.     pBase->set(3, 4);  
7.     pBase->showPoint();  
8.  
9.     pDer = (ColorPoint*)pBase; // 다운 캐스팅  
10.    pDer->setColor("Red");  
11.    pDer->showColorPoint();  
12.}
```

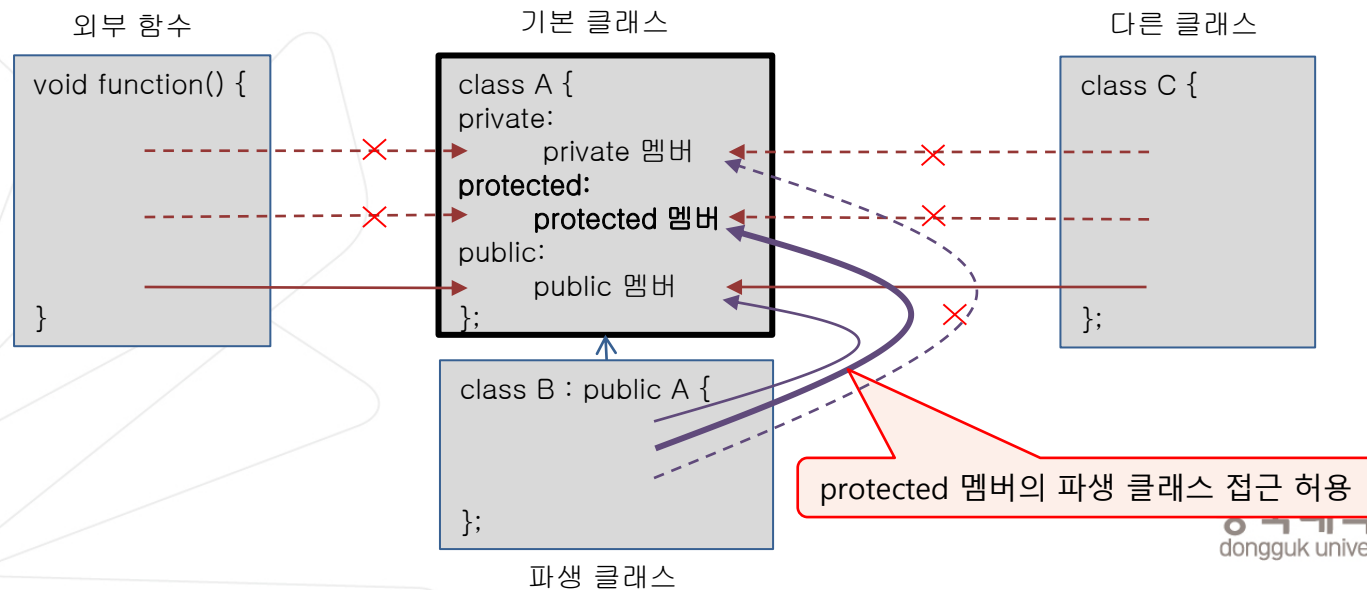
- 다운 캐스팅한 파생 클래스의 포인터로는 파생 클래스의 멤버만 접근 가능함
- 다운 캐스팅 시 명시적 타입 변환은 필요함

# 접근 지정자 protected 멤버

## ● protected 접근 지정

- 기본 클래스에 protected로 지정된 멤버는 **파생 클래스**에게 접근 허용함
- protected도 private처럼 다른 클래스나 외부 함수에서 접근할 수 없도록 숨겨짐
- protected를 사용하더라도 기본 클래스의 private 멤버는 접근 불가능함

	private 멤버	protected 멤버	public 멤버
멤버를 선언한 클래스	O	O	O
파생 클래스	X	O	O
다른 클래스나 외부 함수	X	X	O



1. 객체 지향 언어의 상속 개념과 가장 가까운 것은?

- ① 오리와 닭은 가축이다.
- ② 스마트폰은 일반 전화기처럼 전화를 할 수 있을 뿐만 아니라 사진도 찍을 수 있다.
- ③ 박 군의 집은 아버지가 주신 집이다.
- ④ 사람은 원숭이로부터 진화했다.

2. 객체 지향에서의 상속 개념과 가장 가까운 것은?

- ① TV와 스마트 TV
- ② 천 소파와 가죽 소파
- ③ CPU와 PC
- ④ 자동차 엔진과 변속기

3. C++의 상속 특징 중 틀린 것은?

- ① C++에서는 다중 상속을 허용하지만 다중 상속으로 문제가 발생하는 경우도 있다.
- ② C++에서는 상속의 횟수에 제한이 없다.
- ③ `protected` 멤버는 클래스 내에서 보호받기 때문에 상속되지 않는다.
- ④ 상속의 가장 큰 장점은 소프트웨어의 재사용에 있다.

4. 다음 코드에 대해 컴파일 오류가 발생하는 것은?

```
class A {  
    public: int w;  
};  
class B : public A {  
    public: int x;  
};  
class C : private A {  
    public: int y;  
};  
class D : protected B {  
    public: int z;  
};
```

① A a; a.w=10;

② B b; b.w=10;

③ C c; c.y=10;

④ D d; d.w=10;

5 다음 코드에서 컴파일 오류가 발생하는 곳을 있는 대로 골라라.

```
class A {  
    int s, x;  
protected :  
    void setX(int x) { this->x = x; }  
    void setS(int s) { this->s = s; }  
};  
  
class B : private A {    // ①  
    int y;  
public:  
    void setXY(int a, int b, int s) {  
        x = a;           // ②  
        y = b;           // ③  
        setS(s);         // ④  
    }  
};
```

6. 다음 코드에서 컴파일 오류가 발생하는 곳을 있는 대로 골라라.

```
class A {  
    int x;  
public :  
    void setX(int x) { this->x = x; }  
};  
  
class B : protected A {  
    int y;  
public:  
    void setXY(int x, int y) { setX(x); this->y = y; }  
};  
  
int main() {  
    A a;  
    B b;  
    a.x = 3;      // ①  
    b.y = 3;      // ②  
    a.setX(5);    // ③  
    b.setX(5);    // ④  
}
```

7. 다음 클래스 A, B와 변수가 선언되어 있을 때 물음에 답하라.

```
class A {  
    public: int x;  
};  
class B : public A {  
    public: int y;  
};  
A a, *p;  
B b, *q;
```

(1) 업 캐스팅과 다운 캐스팅을 골라라.

① `p = &a;`      ② `p = &b;`      ③ `q = (B*)&a;`      ④ `q = &b;`

(2) 다음 코드는 컴파일 오류는 없지만 실행 중에 오류가 발생한다. 그 이유는 무엇인가?

```
p = &a;  
q = (B*)p;  
q->y = 100; // 실행 중 오류 발생
```

8. 다음 클래스 A, B와 변수가 선언되어 있을 때 물음에 답하라.

```
class A {  
    public: int w;  
};  
class B : public A {  
    public: int x;  
};  
class C : public A {  
    public: int y;  
};  
class D : public B {  
    public: int z;  
};  
A a; B b; C c; D d; // 클래스마다 하나씩 객체 생성  
  
A *ap = &a;  
B *bp = &b;  
C *cp = &c;  
D *dp = &d;
```

(1) 업 캐스팅에 해당하지 않는 것은?

- ① `ap = bp;`    ② `ap = cp;`    ③ `bp = cp;`    ④ `bp = dp;`

(2) 객체 d의 멤버를 모두 나열하라.

(3) 객체 d의 멤버에 대한 접근 중에서 컴파일 오류가 발생하는 것은?

- ① `d.x = 10;`    ② `dp->x = 10;`  
③ `ap = dp; ap->x = 10;`    ④ `bp = dp; bp->x = 10;`

(4) 아래 두 라인을 작성하면 두 번째 라인에서 컴파일 오류가 발생한다. 수정하라.

```
ap = dp; // 컴파일 오류 발생하지 않음. 업 캐스팅  
dp = ap; // 컴파일 오류 발생. 다운 캐스팅
```



9. 다음 코드에 대한 물음에 답하라.

```
class A {  
public:  
    A() { cout << "생성자 A" << endl; }  
    A(int x) { cout << "생성자 A " << x << endl; }  
};  
class B : public A {  
public:  
    B() { cout << "생성자 B" << endl; }  
    B(int x) { cout << "생성자 B " << x << endl; }  
    B(int x, int y) : A(x + y + 2) { cout << "생성자 B " << x*y*2 << endl; }  
};
```

다음과 같이 객체 b가 생성될 때 화면에 출력되는 내용은?

- (1) B b;
- (2) B b(10);
- (3) B b(10, 20);

10. 다음 코드에 대해 물음에 답하라.

```
class A {  
public:  
    A(int x) { cout << "생성자 A " << x << endl; }  
};  
class B : public A {  
public:  
    B() { cout << "생성자 B" << endl; }  
    B(int x) { cout << "생성자 B " << x << endl; }  
};
```

(1) 위 코드에서 컴파일 오류가 발생하는 곳은 어디인가? 컴파일 오류의 내용은 무엇인가?

(2) 다음 코드와 실행한 결과가 아래와 같도록 생성자 B()를 수정하라.

B b;

→ 실행 결과

생성자 A 20

생성자 B

(3) 다음 코드와 실행한 결과가 아래와 같도록 생성자 B(int x)를 수정하라.

B b(15); → 실행 결과

생성자 A 35

생성자 B 15



# 실습과제(8장)

문제1에 적용되는 원을 추상화한 Circle 클래스가 있다.

```
class Circle {  
    int radius;  
public:  
    Circle(int radius=0) { this->radius = radius; }  
    int getRadius () { return radius; }  
    void setRadius (int radius) { this->radius = radius; }  
    double getArea() { return 3.14*radius*radius; }  
};
```

# 실습과제(8장)

1. 다음과 같이 배열을 선언하여 다음 실행 결과가 나오도록 Circle을 상속받은 NamedCircle 클래스와 main() 함수 등 필요한 함수를 작성하라.

```
NamedCircle pizza[5];
```

5 개의 정수 반지름과 원의 이름을 입력하세요

1>> 5 크림피자

2>> 8 치즈피자

3>> 25 대왕피자

4>> 30 블랙홀피자

5>> 15 마늘피자

가장 면적이 큰 피자는 블랙홀피자입니다

힌트

NamedCircle 클래스에 디폴트 매개 변수를 가진 생성자를 작성해야 한다.

# 실습과제(8장)

문제2에 적용되는 2차원 상의 한 점을 표현하는 Point 클래스가 있다.

```
class Point {  
    int x, y;  
public:  
    Point(int x, int y) { this->x = x; this->y = y; }  
    int getX() { return x; }  
    int getY() { return y; }  
protected:  
    void move(int x, int y) { this->x = x; this->y = y; }  
};
```

# 실습과제(8장)

2. 다음 main() 함수가 실행되도록 Point 클래스를 상속받는 ColorPoint 클래스를 작성하고, 전체 프로그램을 완성하라.

```
int main() {  
    ColorPoint zeroPoint; // BLACK 색에 (0, 0) 위치의 점  
    zeroPoint.show(); // zeroPoint를 출력한다.  
  
    ColorPoint cp (5, 5);  
    cp.setPoint (10, 20);  
    cp.setColor ( "BLUE");  
    cp.show(); // cp를 출력한다.  
}
```

BLACK색으로 (0,0)에 위치한 점입니다.  
BLUE색으로 (10,20)에 위치한 점입니다.

# 실습과제(8장)

문제3에 적용되는 BaseArray 클래스는 다음과 같다.

```
class BaseArray {  
private:  
    int capacity; // 배열의 크기  
    int *mem; // 정수 배열을 만들기 위한 메모리의 포인터  
protected:  
    BaseArray(int capacity=100) {  
        this->capacity = capacity; mem = new int [capacity];  
    }  
    ~BaseArray() { delete [] mem; }  
    void put(int index, int val) { mem[index] = val; }  
    int get(int index) { return mem[index]; }  
    int getCapacity() { return capacity; }  
};
```

# 실습과제(8장)

3. BaseArray 클래스를 상속받아 스택으로 작동하는 MyStack 클래스를 작성하라.

```
MyStack mStack (100);
int n;
cout << "스택에 삽입할 5개의 정수를 입력하라>> ";
for (int i=0; i<5; i++) {
    cin >> n;
    mstack.push(n); // 스택에 푸시
}
cout <<"스택용량 : " << mStack.capacity() <<" , 스택크기:"<<mStack.length() << endl;
cout << 스택의 모든 원소를 팝하여 출력한다>> ";
while(mStack.length() != 0) {
    cout << mStack.pop ( ) << ' '; // 스택에서 팝
}
cout << endl << "스택의 현재 크기 : "<< mStack.length() << endl;
```

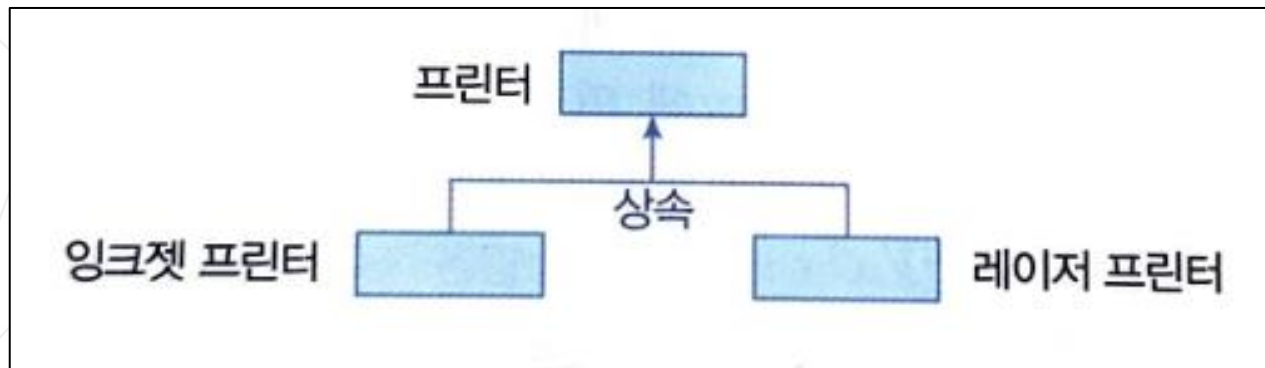
```
스택에 삽입할 5개의 정수를 입력하라>> 1 3 5 7 9
스택 용량: 100, 스택 크기:5
스택의 모든 원소를 팝하여 출력한다>> 9 7 5 3 1
스택의 현재 크기 : 0
```



# 실습과제(8장)

4. 다음 그림과 같은 상속 구조를 갖는 클래스를 설계한다.

모든 프린터는 모델명(model), 제조사(manufacturer), 인쇄 매수 (printedCount), 인쇄 종이 잔량(availableCount)을 나타내는 정보와 print(int pages) 멤버 함수를 가지며, print()가 호출할 때마다 pages 매의 용지를 사용한다. 잉크젯 프린터는 잉크 잔량(availableInk) 정보와 printInkJet (int pages) 멤버 함수를 추가적으로 가지며, 레이저 프린터는 토너 잔량(availableToner) 정보와 역시 printLaser(int pages) 멤버 함수를 추가적으로 가진다. 각 클래스에 적절한 접근 지정으로 멤버 변수와 함수, 생성자, 소멸자를 작성하고, 다음과 같이 실행되도록 전체 프로그램을 완성하라. 잉크젯 프린터 객체와 레이저 프린터 객체를 각각 하나만 동적 생성하여 시작한다.



# 실습과제(8장)

현재 작동중인 2 대의 프린터는 아래와 같다

잉크젯 : Officejet V40 ,HP ,남은 종이 5장 ,남은 잉크 10

레이저 : SCX-6x45 ,삼성전자 ,남은 종이 3장 ,남은토너 20

프린터(1:잉크젯, 2:레이저)와 매수 입력>>1 4

프린트하였습니다.

Officejet V40, HP ,남은 종이 1장 ,남은 잉크 6

SCX-6x45 ,삼성전자 ,남은 종이 3장 ,남은토너 20

계속 프린트 하시겠습니까(y/n)>>y

프린터(1:잉크젯, 2:레이저)와 매수 입력>>2 10

용지가 부족하여 프린트할 수 없습니다.

Officejet V40, HP ,남은 종이 1장 ,남은 잉크 6

SCX-6x45 ,삼성전자 ,남은 종이 3장 ,남은토너 20

계속 프린트 하시겠습니까(y/n)>>y

프린터 (1:잉크젯, 2:레이저)와 매수 입력>>2 2

프린트하였습니다.

Officejet V40 ,HP ,남은 종이 1장 ,남은 잉크 6

SCX-6x45 ,삼성전자 ,남은 종이 1장 ,남은토너 19

계속 프린트 하시겠습니까(y/n)>>n