5주차 예비보고서

전공 : 컴퓨터공학전공 학년 : 2학년 학번 : 20151616 이름 : 최승환

1.목적

C++는 객체지향 프로그래밍 개념을 도입하여 클래스(Class)를 확장하고 수정하는 수단으로 클래스 상속(Class Inheritance)을 제공한다. 이번 실험에서는 효율적인 프로그래밍을 위하여 클래스의 상속 방법을 익혀보도록 한다.

2.문제 설명

아래 부분에 추상화된 기초 클래스 Array는 new 연산자를 사용하여 정수 타입의 1차 배열을 동적으로 생성한다. 또한 배열의 인덱스를 나타내는 ‘[]’에 대하여 연산자 다중정의를 하여 Array클래스로 생성된 객체에서도 사용할 수 있도록 한다. 우리는 Array클래스를 상속받아 RangeArray를 추가로 구현하려 한다. RangeArray클래스의 특징은 배열이 인덱스 0에서 시작되는 것이 아니라, 생성자에 전달된 임의의 범위를 갖도록 한다. 한 예로 RangeArray A가 A(-10,10)로 생성되었다면, 인덱스 (-10 ~ 10)를 가지고, 그 배열은 21개의 원소를 저장할 수 있다. 즉 A[-10], A[-9],.......,A[0] , A[1], ….A[10]은 모두 올바른 배열 표현이다. 객체 지향 플고그래밍에 적합하도록 Array클래스로부터 가능한 많은 기능을 상속받아 RangeArray클래스를 구현한다.

|  |
| --- |
| 객체-지향 언어는 다음의 5가지 특성을 가진다.  1. Object  2. Class  3. Encapsulation  4. Inheritance  5. Polymorphism  객체의 속성(fields)와 행위(method)를 하나로 묶고, 실제 구현 내용의  일부를 외부에 감추어 은닉하는 성질을 캡슐화 라고 한다. private,  protected, public 과 같은 접근지정자를 사용하여 은닉한다.  프로그램 언어의 다형성(polymorphism)은 그 프로그래밍 언어의 자료형  체계의 성질을 나타내는 것으로, 프로그램 언어의 각 요소들(상수, 변수, 식,  오브젝트, 함수, 메소드 등)이 다양한 자료형(type)에 속하는 것이 허가되는  성질을 가리킨다.  ※ 객체, 클래스, 상속은 다음 질문에서 언급되므로 캡슐화, 다형성에  대하여만 정리하였다.  객체 : 객체 지향 프로그래밍에서 데이터(필드)와 그 데이터에 관련되는  동작(method)을 모두 포함한 개념.  클래스 : 유사한 특징을 지닌 객체(object)들의 속성을 묶어 놓은 집합체.  인스턴스 : 클래스를 실제로 사용할 수 있도록 메모리에 할당하는 것을  인스턴스 생성이라 하며, 생성된 객체를 인스턴스라 한다.  OOP를 쓰는 이유 :  1. 객체 지향 프로그래밍을 사용함으로써 캡슐화, 다형성, 상속을  이용하여 코드 재사용률을 증가 시키고, 유지보수를 감소시키는  장점을 얻을 수 있다.  2. 클래스는 관련된 데이터만 정의하기 때문에, 그 클래스의 인스턴스가  수행될 때 다른 프로그램에서 사용하는 데이터를 건드릴 수 없게 된다.  자료 숨김 특성은 높은 시스템 보안을 제공하고, 의도하지 않은 자료의  훼손을 방지한다.  3. 데이터 클래스 개념은 언어에 정의되지 않은 새로운 데이터 형식을  프로그래머가 임의로 정의할 수 있게 한다.  상속 : 기존의 클래스를 재사용하여 새로운 클래스를 작성하는 것. 상속의  개념을 사용하여 코드를 재사용 할 수 있으며, 클래스의 계층 구조를 표현할  수 있게 된다. |