4주차 예비보고서

전공: 경제학과 학년: 4학년 학번: 20180501 이름: 김연수

첫 번째로, LinkedList.h를 구현한다. LinkedList.h file에서 Node class와 LinkedList class를 정의한다. 이 때 둘 모두 템플릿클래스로 선언해서 데이터타입으로 float, int, double, string 등 다양하게 들어올 수 있도록 한다. 즉 파라미터적 다형성을 만족시켜주도록 한다.

서브타입 다형성을 만족시켜서 더 효율적으로 작동하도록 할 수 있다. 서브타입 다형성을 이용하면, 자료형 T로 선언한 변수에 자료형 T의 객체 뿐만 아니라 서브타입 자료형의 객체도 사용할 수 있다. 서브타입 클래스는 기본클래스에서 파생된 클래스를 말한다. 서브 타입 다형성을 만족시키기 위해서 이 프로그램에서는 일단 LinkedList 자료형을 가지는 변수를 선언한다. 사용자의 입력에 따라서 파생 클래스인 Stack을 할당할 수 도 있고, 기본클래스인 LinkedList를 할당할 수도 있다. 사용자는 안내문에 따라서 1또는 2를 입력한다. 1을 입력하면 stack 자료형을 사용하는 것이고, 2를 입력하면 LinkedList를 사용하는 것이다. 이와 같은 방식으로 서브타입 다형성을 만족시킬 수 있다. 서브타입 다형성을 이용하면 다양한 이익을 볼 수 있다. 코드 양을 대거 줄일 수 있고, 프로그램을 유연하게 만들어준다. 프로그램이 유연하면 유지보수하기가 편리해진다.

두 번째로, Stack 클래스를 구현한다. Stack 자료형은 LIFO(last in first out)이다. 즉 마지막으로 들어간 데이터가 처음으로 나온다. 이를 구현하기 위해서는 LinkedList와 다른 delete함수를 써야한다. 다른 변수와 함수는 다 수정없이 활용할 수 있다. 이 때 LinkedList 복사해서 붙여넣기 하는 방식이 아니라, 상속받아서 파생클래스를 만들어주면 효율적인 프로그램을 만들 수 있다. 불필요한 코드가 줄고, 유지보수하기도 편리해진다. 가독성도 향상시킬 수 있다. 이 때 주의해야할 점은 delete함수를 선언할 때 virtual로 선언되어야 한다. 그렇지 않으면 정확한 작동을 보장하기 어렵다.