전공 : 컴퓨터공학과 학년 : 4 학번 : 20212020 이름 : 박민준

1. 주어진 문제 및 3-1의 문제 해결에 관한 내용을 이해하고 이 문제를 효율적으로 해결하기 위한 방법을 생각하여 이를 1쪽 이내로 요약하여 제출하시오. 문제 해결을 위한 간단한 단계별 수행 내용, 자료구조 등을 기술하시오

4주차 실습은 LinkedList 클래스를 template 클래스로 확장하고, 이를 상속받아 Stack 클래스를 구현함으로써 객체지향 프로그래밍의 특성 중 하나인 다형성에 대해 이해할 수 있다.

먼저, 기존의 LinkedList 클래스는 int 타입의 데이터만을 저장할 수 있었다. 그러나 template을 도입함으로써 LinkedList<T>와 같이 선언하여 어떠한 타입의 데이터도 저장할 수 있도록 만들 수 있다. 이러한 과정에서 Node 클래스 또한 template를 사용하여 다양한 타입의 데이터를 저장하도록 확장할 수 있다.

또한, LinkedList는 Print 함수를 제공하지 않기 때문에 이를 직접 구현해야 한다. Print 함수를 구현하기 위해서는 LinkedList를 순회하면서 각 노드의 데이터를 출력하는 방식을 사용해야 한다.

이렇게 template를 사용하여 확장된 LinkedList 클래스를 상속받아 Stack 클래스를 구현할 수 있다. Stack 클래스는 LinkedList의 모든 기능을 상속받되, Delete 함수만을 Overriding하여 Stack의 동작 방식(LIFO)에 맞게 맨 뒤가 아닌 맨 앞의 원소를 삭제하도록 구현해야 한다.

이후 최종적으로 제공된 테스트 코드를 사용하여 Stack과 확장된 LinkedList 클래스가 정상적으로 동작하는지 확인해야 한다. 해당 과정에서, 다형성을 통해 Parent 클래스의 포인터를 사용해 Child 클래스의 인스턴스에 접근하는 것을 확인할 수 있다.

해당 실습에는 3가지 자료구조가 포함된다. 먼저, template 기반 Node 클래스는 다양한 타입의 데이터를 저장할 수 있도록 구현된 노드이다. 두 번째로 template 기반 LinkedList 클래스는 다양한 타입의 데이터를 연결 리스트 형태로 관리한다. 기본적인 연산으로는 삽입, 삭제, 출력이 있다. 마지막으로 Stack 클래스는 LinkedList 클래스를 상속받아 구현된 스택이다. 스택의 특성에 맞게 Delete 함수만을 Overriding하게 된다.