4주차 결과보고서

전공: 컴퓨터공학과 학년: 4학년 학번: 20192135 이름: 윤영인

**1.**

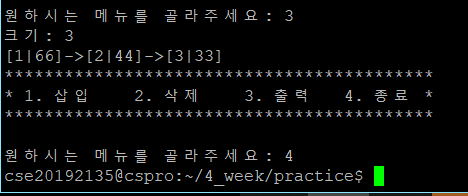
LinkedList.h에는 연결리스트를 구성하는 Node와 LinkedList 클래스가 정의되어 있다. 템플릿 자료형을 사용한 확장형 클래스를 구현하기 위해 클래스 정의 전에 template<typename T>를 써 주었다. 연결리스트를 구성하는 Node 클래스는 생성자로 자료형이 T인 element를 갖고, 다음 노드의 주소를 저장하는 포인터인 link를 가진 Node 인스턴스를 생성한다. LinkedList 클래스는 생성자에 의해 첫 번째 주소를 담지 않은 포인터인 first와 연결리스트의 사이즈를 나타내는 변수인 current\_size에 0을 담아 인스턴스를 생성한다. GetSize() 멤버 함수는 연결리스트의 사이즈를 반환하며, Insert(T element) 멤버 함수는 자료형이 T인 element를 가진 노드를 할당하여 연결 리스트의 맨 앞에 삽입하고 첫 번째 주소를 삽입한 노드로 설정하기 위해 first에 이 노드를 넣어준다. Delete 멤버 함수는 연결리스트를 link를 따라가면서 순환하여 마지막 노드를 찾아 메모리를 할당 해제하고 data 값을 리턴하기 위해 element 변수에 넣어준다. Print 멤버 함수는 link로 연결리스트를 돌면서 노드의 data 값을 모두 출력한다.

Stack.h 헤더에는 LinkedList 클래스의 상속을 받는 Stack 클래스가 템플릿 확장으로 정의되어 있다. 스택은 Last-In-First-Out으로 가장 나중에 들어간 원소가 가장 먼저 나오므로, Delete 멤버함수를 LIFO형식으로 재정의해야 한다. 따라서 Insert 멤버 함수로 삽입한 노드를 첫 번째 노드인 first로 가리키게 하므로 Delete 함수는 first가 가리키고 있는 노드를 삭제해야 한다. 따라서 삭제할 노드인 first를 this 포인터로 가져와 curNode라는 노드형 포인터 변수에 넣어준다. 삭제하는 노드의 값을 리턴하기 위해 element에 삭제할 노드인 curNode의 data를 넣어주고 연결 리스트의 크기를 하나 줄인 뒤 이 노드를 delete로 할당 해제한다.

**2.**

아래의 사진은 make로 실행 파일인 a.out을 생성하여 Stack 클래스로 연결리스트가 만들어졌는지 확인한 결과이다.

텍스트이(가) 표시된 사진

자동 생성된 설명

아래의 사진은 LinkedList 클래스로 만든 연결리스트를 확인한 결과이다.

텍스트이(가) 표시된 사진

자동 생성된 설명