4주차 예비보고서

전공: 아트엔테크놀로지 학년: 3학년 학번: 20191172 이름: 함승우

1.

우선, User의 첫번째 입력값 1 또는 2로 stack으로 구현할지, LinkedList로 구현할지 결정한다. 그 후 User의 두번째 입력값인 1, 2, 3, 4를 통해 삽입 삭제 출력 종료할 지 결정한다. 이에 따라 Print()함수를 만들어야 한다.

LinkedList로 소개된 부분을 다형성을 추구하는 템플릿 클래스(정수형 , int형이 아닌 double, float, string형을 따로 저장하지 않고 변경하면서 사용을 해야 하지만 이 다형성을 추구하는 템플릿 클래스로 구현한다면 모든 자료형을 저장할 수 있다)로 구현하기 위해서는 소개된 Node Class 및 LinkedList class 모두 바꿔야 한다. 이 때 유의해야 할 점은 클래스 인자를 명시해 줘야 한다는 점이다. typename T(template class T)를 통해 임의의 타입 T로 정의해 줘야 한다.

또한 초기 클래스의 설정은 큐의 정의를 따른다고 나와있다. Delete 함수를 개조해서 Stack의 정의를 따르게 하려면, 삭제하는 순서만 바꾸면 된다. 큐는 FIFO(First-int-First-out)구조이고, stack은 LIFO(Last in First out) 구조이므로 기존의 Delete 멤버함수가 삭제하는 인자의 순서를 바꾸면 된다.

마지막으로, 삽입, 삭제, 출력, 종료 중 LinkedList와 Stack 구현의 가장 다른 점은 두 번쨰 요소인 삭제다. 따라서, LinkedList로부터 Stack을 구현받아 delete 함수를 개조하고, 연결리스트를 템플릿 class로 변환한다면 어떤 자료형이든 처리할 수 있게 된다.