

2021 소프트웨어프로젝트 리포트 4

Generic Stack

1. 내용

주어진 GenericStackLimitedCapacity 소스를 이용해서 아래 스택들을 만들고 테스트한다.

- 1) Generic Stack with limited capacity (GenericStackLimitedCapacity)
 - 예년에는 스스로 작성하던 것. 올해는 기본 제공.
 - ArrayList 등을 사용하지 않고, 순수 array 만을 사용 (용량은 상수로 한정).
 - 모든 type의 값을 넣을 수 있음.
 - push, pop, isEmpty 만 지원. (top 변수 반드시 사용)
- 2) Generic Stack with unlimited capacity (GenericStack)
 - GenericStackLimitedCapacity를 super class로 하고, extension으로 구현 (**반드시**)
 - * 용량을 초과하는 경우, array를 2배로 증가시키는 기능만 넣으면 됨.
 - (강의 노트에 유사 코드 있음)
 - * 핵심은 어디를 재정의해야 하는가?
 - super class의 저장 공간은 **visible하게 수정**하여야 한다.
- 3) Parameterized Generic Stack with unlimited capacity (ParaStack)
 - 2)의 코드를 참조하여 **type-safe** 한 동작을 지원하는 ParaStack 구현 및 실험
 - 예) ParaStack<String> s; // String 만을 지원하는 Stack
 - Object 배열을 사용하고, Casting을 해서 type-safe를 구현.
- 4) 다양한 GenericStackLimitedCapacity/GenericStack/ParaStack 테스트
 - 예를 들어, int형과 String형 데이터를 같은 스택에 넣는 것 실험
 - 이 기능은 장점? 단점?

2. 목적

- Java polymorphism, parameterized type 및 generic programming 이해 및 연습
- Inheritance를 이용한 incremental programming 이해 및 연습

3. 추진 방법

- **강의 내용의 이해가 필수.** (강의 내용을 이해 못하면, 과제 내용도 이해할 수 없음).
- 강의에서 배우는 코드를 참조. (배운 내용들을 조합하여 복습하는 과제)
- 이해를 확인하기 위한 **충분한 실험**

4. 평가항목

- 프로그램의 동작 여부 (100%)
 - * 각 Stack 에 대한 각각의 요구사항을 검증할 수 있는 결과
 - * 어떤 테스트를 해야 하는지 스스로 생각해내야 함.
- 요구 사항 만족 필수
 - * 다른 stack 작성 시, copy 간주

5. 리포트 제출

- 1) 기한 : **5/9(일)** 까지
 - 각 Stack 프로그램에 대해
 - * 소스 프로그램 및 설명 + 테스트 과정 화면
 - 하나로 묶어서 리포트로 eClass로 제출

```

class GenericStackLimitedCapacity {

    private int top = -1;
    private final int INIT_CAPACITY = 2;
    private Object[] list;

    GenericStackLimitedCapacity() {
        list = new Object[INIT_CAPACITY];
    }

    GenericStackLimitedCapacity(int capacity) {
        list = new Object[capacity];
    }

    public boolean isEmpty() {
        return (top < 0);
    }

    public Object pop() {
        if (top >= 0)
            return list[top--];
        else
            return null;
    }

    public void push(Object o) {
        list[++top] = o;
    }

    /** Override the toString in the Object class */
    public String toString() {
        String ss = "";
        for(int i=0; i<=top; i++) ss += list[i] + " ";
        return ss;
    }

}

```