

## 2025 소프트웨어프로젝트 프로젝트 2

### Java 및 C 프로그램 스타일 비교

#### 1. 내용

주어진 C 프로그램, numarray.c와 같은 기능을 갖는 Java 프로그램을 Java 스타일에 맞게 작성한다. 프로그램은 정수형 array에서 최대값을 찾고, 평균을 계산하며, 두 array를 합치는 기능을 제공한다. C 프로그램은 예시이며, 실제 Java class의 정의 및 내부 구현은 본인의 스타일로 일관성 있게 구현한다. 단, Java 프로그램은 반드시 **Java 스타일에** 맞게 작성하여야 한다.

Java의 array 기능을 main()으로 동작시키는 것은 과제의 목적이 아님. 즉, 주어진 C 프로그램에 대응하는 Java 프로그램을 C 스타일로 짜는 것은 의미가 없음. array 데이터 공간과 관련 동작을 하나로 묶은 class를 작성하고 사용해보면서, class 도입의 의미를 확인하는 것이 프로그램 목적.

덧붙여, **이번 프로그램 작성에 기초해서, 비교 항목을 자체적으로 설정**하고 C와 Java 프로그램 스타일을 **비교**한다. (비교에 있어서 정답은 없음. 개인적인 의견을 논리적으로 설명하면 됨. 메소드 내부 구현은 비교 대상이 아님.)

(주의) 과제 2부터는 Eclipse 등의 개발 도구에서만 과제 수행을 하면 됨.

Java 프로그램은 다음 사항을 **반드시** 지켜야 함.

- **Java Style 프로그램**이어야 함.
  - \* 과제 1의 Java 프로그램은 C style의 프로그램. 즉, 올바른 예가 아님.
  - \* **C와 동일한 스타일로 프로그램을 하는 것은 의미가 없음.**
  - \* 이해를 돕기 위해서 첨부되는, circle 관련 C 프로그램과 Java 프로그램을 참조.
- class NumArray를 별도로 만들어서, array와 관련 동작을 encapsulation 하여야 한다.
  - \* 즉, data fields와 **생성자를 포함한** 모든 관련 메소드들을 class 안에 포함.
- 이때, 메소드는 **위치만이 아니라 의미면에서도 객체에 부속되어야** 한다는 점을 주의.
  - \* 예를 들어서, C의 findMax()처럼, 독립적으로 존재하면서, 인자로 주어진 모든 array에 동작하는 형태이어서는 안 됨.
  - \* arr1.findMax() 방식으로 사용하면서, **배열 arr1의 최대값을 처리**하여야 함.
- 프로그램의 테스트 동작은 **반드시** 별도의 **class TestNumArray**를 만들어서 수행.
  - \* public static main() 메소드는 여기에
- 정의된 method는 class를 통해서 **접근해서는 안 됨**. 반드시 객체 변수를 통해서 접근.
- 참고) Java에서는 array 선언할 때, “new”를 앞에 써야 함.
  - 예) C에서의 “int A[100];”는 “int[] A = new int[100];”
- **주의)** Java에 대한 강의가 많이 진행되지 않아서, 과제 2 자체는 상대적으로 간단함.
  - 과제 3부터는 본격적으로 class를 사용하면서 복잡해짐. 따라서 과제 2부터 class의 의미를 제대로 이해하면서 프로그램을 작성하는 것이 중요.

#### 2. 목적

- C 프로그램 스타일과 Java 프로그램 스타일의 차이를 체득.
- 실전적 경험을 통해 Java 프로그램의 장단점의 예측
- Java의 장점을 살리는 프로그래밍 방식에 대한 **목표 의식 설정**

#### 3. 추진 방법

- 1) 주어진 C 프로그램을 이해.
- 2) 주어진 Circle 관련 C 프로그램과 Java 프로그램을 분석.
- 3) C로 작성된 프로그램을, 요구 사항에 맞게, Java 프로그램으로 재작성

4) 인터넷 자료 등을 검색해서 Java 스타일에 맞는지 확인

- 주의) 주어진 C 프로그램과 동일한 수준의 프로그램만 작성. 과도하게 코드를 작성하면 copy 의심

5) 자기 나름대로, 두 스타일의 프로그램을 비교.

- 비교 항목을 정하고,
- 각 항목에 대해, **과제 프로그램에 입각해서**, 같은 점과 다른 점을 설명하고,
- 프로그래밍 관점에서 장단점을 논리적으로 설명.
- **틀려도 좋음.** 사실 정답이 없는 것임.
- 인터넷에서 검색해서 복사하지 말고, 본인의 생각을 기술.
- 앞으로 배울 Java 프로그래밍의 장단점을 이해할 때 도움이 되면 충분

5. 평가항목

- 프로그램의 동작 및 스타일 적용 여부 (80%)
  - \* **요구사항 준수 여부**
  - \* 일반적인 Java 스타일 적용 여부
- 스타일 비교에 대한 **노력** (20%)
  - \* 설계 및 구현 시 고민했던 사항과 결정 내용을 간략히 기술
    - + 본인이 직접 프로그램을 했음을 자연스럽게 입증
  - \* 평가자 입장에서, 평가사항을 찾기 쉽게 작성하였는가? 즉, 확연하게 비교하는 방법은?

5. 리포트 제출

1) 기한 : 3/30(일) 까지

- 최종 테스트 과정 화면, 설계 노트, 소스 프로그램, 자체평가표
- 모든 내용을 하나로 묶은 리포트를 pdf로 만들어서 eClass로 제출