

## 1. 내용

- 실행시간을 측정하는 프로그램을 작성하고,
- 주어진 두 자연수의 공약수 개수를 알려주는 `numCommonDivisor(int i1, int i2)`을 작성하고,
- 그 코드 실행시간을 측정한다.
- 또한, Java 실행 모델의 확인을 위해서, 두 가지 방법으로 실행 절차 및 실행 과정에서 생성되는 파일들을 파악해본다.
- 실행시간보다는 실행 모델이 더 중요한 주제. (즉, 실행시간을 줄이는 것은 중요하지 않음)
- 시간 측정 방법
  - \* 계산 전에 시간 측정 (시작 시간)
  - \* 계산 후에 시간 측정 (종료 시간)
  - \* 두 시간의 차이가 계산에 소요된 시간
- 수행시간이 짧으면, (예, nano-second 보다 작다면) 어떻게 측정?
  - \* 반복 실행하고 (예, 100만 번), 소요 시간을 반복 횟수로 나누어서 단일 실행 시간 계산
- 먼저 실행시간을 측정하는 프로그램 작성하고, `numCommonDivisor()`을 다음에
  - \* 과제보고서에는 `numCommonDivisor(20349, 2747115)`의 실행시간 측정값을 반드시 제출.
- (주의) 이번 학기 프로그램 성능에 관심을 두는 것은 이번이 마지막. 이번 학기의 주제는 프로그램의 구조
- 이 프로그램을 두 방법으로 각각 구현/실험하여야 함.
  - 1) Java 프로그램 실행 방식에 따라서 단계별로,
    - + notepad 등의 editor를 이용하여 프로그램 작성하고,
    - + command 창에서 단계적으로 실행
    - + 각 단계에서 생성되는 중간 결과물을 반드시 확인해서 리포트에서 설명
  - 2) Eclipse를 사용하여 개발
- 관련 Java 지식
  - \* 프로그램 시작 함수: `public static void main(String[] args)`
  - \* `main()`을 포함하는 class로 프로그램을 구성하여야 함.
  - \* 현재 시간: `System.nanoTime()` - 시스템의 nano 단위 현재 시간을 long 타입으로 알려줌.
  - \* 화면으로 결과 출력: `System.out.println()`
  - \* 스트링 변수형: `String` 예) `String s;`
  - \* 스트링 concatenation operator '+' 예) `s = "hello" + s;`
  - \* 나머지는 C 언어와 유사

## 2. 목적

- Java 프로그램 입문: 프로그램 작성, 컴파일, 실행 실습
  - \* 자바 실행 모델을 중간결과물과 함께 확인
- Eclipse 사용법
- 리포트 작성 및 제출 방법 확인

### 3. 추진 방법

- 1) 1장 강의 노트 내용을 참조해서 에디터로 MeasureExecTime.java 작성
- 2) javac로 compile한 뒤, 결과물 확인
- 3) java 인터프리터로 실행

-----

#### 4) eclipse 실행

- 프로젝트 생성, 클래스 생성, 코드 추가, 실행 과정을 연습

### 4. 평가항목

- 프로그램의 동작 여부 (40%)
  - \* 동작 여부를 확연히 알아볼 수 있는 결과
- 리포트 (60%)
  - \* 각각의 방법에 따라 구분하여 표시/설명
    - 각 방법에서 단계별로 표시/설명
  - \* 평가자 입장에서 리포트 작성.
    - 자신이 중요하게 생각하는 것이 아니고, 평가자가 원하는 것을 제시/설명
    - 즉, 평가자가 보고자 하는 부분을 명확히 표시/설명
  - \* 자체평가표 참조 및 빈칸 채워서 리포트 끝에 첨부
    - 평가표 기재 내용은 채점 대상이 아님. 자체 평가를 했는가만 체크.
  - \* 본인이 실제 수행했음을 증명(?)하는 자료 포함
  - \* 자신의 노력을 평가자가 오해하는 것을 방지하려면??
    - + 평가자가 원하는 것? 과제 목적!
    - + 평가자가 찾기 쉽게! (양이 너무 많으면?)
    - + 평가자는 항상 copy를 의심! 자신의 결정/설명 -> 남과는 다른 리포트
    - + 리포트 설명서를 꼼꼼히 읽고 요구 사항에 맞게!!!

### 5. 리포트 제출

- 1) 기한 : 3/17(일) 24:00 까지
    - 테스트 과정 화면,
    - 소스 프로그램
    - 자체평가표
    - 모든 제출 자료를 하나의 pdf 문서로 묶어서 eClass로 제출
- \* 평가 시 동작 점검 안 함. (필요한 경우, 별도 지시가 있을 것임.)