

## 난독화기 제작 프로젝트

CDA0119

### 프로젝트 내용

#### ▶ 코드난독화의 구현

- ▶ 입력: Java class 파일
- ▶ 출력: 코드난독화가 적용된 class 파일

#### ▶ 프로젝트 진행 과정

1. 난독화 기법을 선정 (==> 5/26 발표)
2. BCEL을 이용하여 class 파일 변경 (==> 6/2 구현)
3. 제공되는 test.class가 잘 실행되는 것을 확인
4. 역컴파일이 잘 안되는 것을 확인
5. 프로젝트 최종 발표 (==> 6/9 발표)

## 코드난독화 프로젝트

---

### ▶ 채점 기준

- ▶ 난독화의 정도
- ▶ 난독화 알고리즘의 난이도
- ▶ 보고서의 충실도

### ▶ 보고서 내용

- ▶ 난독화 방법의 설명
- ▶ 구현에 대한 설명
- ▶ test.class 및 본인이 작성한 class에 대한 실행 결과
- ▶ 제작 후기



## 프로젝트 일정

---

- ▶ 5/26(목) 계획 발표
- ▶ 6/2(목) 프로젝트 데이
- ▶ 6/9(목) 프로젝트 발표회
- ▶ 6/10(금) 보고서 제출
  
- ▶ 6/16(목) 기말고사 (15:00~16:00)



## 프로젝트 제출 세부사항

- ▶ 팀 과제: 2인 1조
- ▶ 최종 제출물
  - ▶ 소스코드 (파일)
  - ▶ 보고서 (파일)
  - ▶ 발표자료 (파일)
- ▶ 제출 장소
  - ▶ 창원대학교 수업커뮤니티 내 "과제물"



## TestRSA.java

```
public class TestRSA {  
    public static void main(String[] args) {  
        long a = 65, b = 17, c = 3233;  
        System.out.println(a + "^" + b + " mod "  
            + c + " = " + FE.fastExp(a, b, c));  
  
        a = 2790; b = 2753; c = 3233;  
        System.out.println(a + "^" + b + " mod "  
            + c + " = " + FE.fastExp(a, b, c));  
    }  
}
```



## FE.java

---

```
public class FE {  
    // 빠르게  $m^e \bmod n$  을 구하기 위한 알고리즘  
    public static long fastExp(long m, e, n) {  
        long z = 1;  
        while (e != 0) {  
            while (e % 2 == 0) {  
                e = e / 2;  
                m = (m * m) % n;  
            }  
            e--;  
            z = (z * m) % n;  
        }  
        return z;  
    }  
}
```

---

