

## Homework 7

SNU 4541.664A, 2011 봄

**Due: 05/27(Fri) 24:00**

Kwangkeun Yi

이번 숙제의 목표는:

- T라는 하위의 언어를 대상으로 분석기를 디자인하고 구현해 본다.
- 실행과정 의미구조(trace semantics)에 기반해서 연습해 본다.
- 지난 숙제에서 정의한 모듈함수(functor)들을 이용해서 필요한 요약도메인을 간편하게 구현한다.

T 언어의 정의는 과목 홈페이지에 있다.

### Exercise 1 “daVinci Code Detector: 분석기 디자인과 구현 IV”

다빈치가 심혈을 기울여 T로 개발한 소프트웨어. 그 소스에 다빈치가 원저자임을 식별할 수 있는 고유의 식별 코드 조각이 뿌려져있다고 알려져 있다. 최근에 다빈치 코드의 성질은 다음과 같은 것임이 밝혀졌다:

어떤 변수가 있어서, 소프트웨어가 실행중에 그 변수가 가지는 정수값은 항상 1867(다빈치의 탄생년도 1452 와 탄생월일 0415를 더한 값)으로 나누어 나머지가 415(다빈치의 탄생월일)이다.

그러나 주어진 소프트웨어가 다빈치의 진품인지 아닌지를 확인하려면 간단치가 않다. 소프트웨어들을 모든 입력의 경우마다 일일이 돌려보고 그런 성질의 변수가 있는지를 찾아내야 하는데, 이 과정이 비용이 많이 들거나 심지어는 아예 불가능하게 된다. 입력의 경우가 무한하거나 너무 많은 경우도 그렇고, 소프트웨어가 사용하는 많은 변수들의 값들을 실행을 통해서 일일이 추적해야 하는 어려움도 그렇고.

좋은 방법은 정적분석 기술을 이용하는 것이다. 정적분석기술을 이용하면 주어진 소프트웨어가 위와 같은 변수를 가지고 있는지를 자동으로 안전하게 판

단할 수 있게 된다. 정수 변수들이 실행중에 가지는 값들이 1867로 나누어 어  
면 나머지 값들을 가지는지를 분석하면 된다.

이번 숙제는 그런 분석기를 만드는 것이다. 제출할 것은 디자인 리포트(PDF)와  
구현된 코드이다

- 디자인: 분석기에 해당하는 “요약 의미구조”(abstract semantics)를 정의  
하고 그 안전성을 증명한 것을 리포트로 제출한다.
- 구현: 분석기 프로그램

`daVinciCode : T.cmd → answer`

을 구현해서 제출한다. 분석기 `daVinciCode`는 주어진 `T` 프로그램을 분  
석한 후 그 결과를 낸다. 분석결과는 “네, 다빈치코드 입니다”(yes), “아  
니오, 다빈치코드가 아닙니다”(no), 혹은 “잘모르겠습니다”(dontknow).

- `T` 언의 구조를 구현한 모듈 `T`와 데이터타입 `answer`는 TA가 제공할 것이  
다.

□

### Exercise 2 “interval analyzer: 분석기 디자인과 구현 V”

이번에 분석할 것은 메모리가 가지는 정수들의 범위를 분석하는 것이다. 제  
출할 것은 디자인과 코드만 제출하면 된다.

- 디자인: 분석기에 해당하는 “요약 의미구조”(abstract semantics)를 정의  
하고 그 안전성을 증명한 것을 리포트로 제출한다.
- 구현: 분석기 프로그램

`intervalT : T.cmd → unit`

을 구현해서 제출한다. 분석기 `intervalT`는 주어진 `T` 프로그램을 분석  
한 후 그 결과를 출력한다. 분석결과는 프로그램 코드의 처음부터 끝까  
지 일렬로 훑으면서 각 명령문 실행 직전의 메모리를 예측한 결과를 출력  
하도록 한다.

□