

Peter O'Hearn

CAPSULE INTRODUCTION

SNU CSE DISTINGUISHED LECTURE SERIES

Peter O'Hearn 교수는 소프트웨어 오류 검증에서 많은 기대를 모으고 있는 “따로 논리”(Separation Logic)라는 논리를 발명하고 그 실용성을 보여주고 있는 리더이다.

논리란 어떤 사실들을 확인해가는 방법이다. “논리를 발명한다”는 것은 목표로 하는 종류의 사실들을 특히 편하게 확인할 수 있는 방법을 고안한다는 뜻이다. “발명한 논리의 실용성을 보인다”는 것은 그 논리의 방식대로 따지고 들면 목표로 하는 성질들을 편하게 확인할 수 있게 됨을 실제적인 케이스를 통해 확인한다는 뜻이다.

“따로 논리”(Separation Logic)는 컴퓨터 프로그램의 실행 성질을 미리 확인하는 방법으로 고안되었다. 특히, C로 짜여진 시스템 프로그램같이, 메모리 주소를 자유롭게 다루면서 변화무쌍하게 메모리를 다루는 경우(pointers as values + dynamic memory allocations)에 유용하도록 고안되었다. “따로 논리”가 주목받는 이유는, C에서 같이 복잡하게 메모리 주소를 다루는 프로그램의 성질을 확인하는 데 “따로 논리”가 편리하게 사용될 수 있다는 것을 보여주기 때문이다.

이 논리가 “따로 논리”라고 이름 붙은 이유는, 확인방법이 “따로따로”이기 때문이다. 프로그램이 다루는 전체 메모리를 조각 조각 나누어 “따로따로” 확인할 수 있기 때문이다. 그리고, 따로 확인한 것을 모아 전체를 확인한 것으로 확장하기 편리하기 때문이다.

따라서 “따로 논리”는 컴퓨터과학에서 늘 하는 접근 방법을 논리에 적용한 것이다: 전체의 답은 부분들의 답들로 만들도록 하고, 부분들의 답을 만들때는 전체를 신경쓰지 않기.

“따로 논리”의 방식이 편리한 이유는, 부분들의 답을 만들때 전체를 신경쓰지 않기 때문에, 살피고 기록해야 하는 사실들이 주변만으로 한정될 수 있기 때문이다. 그래서 “따로 논리”로 검증하는 과정을 컴퓨터로 구현할때 소모하는 메모리와 시간(계산 비용)이 많이 줄어든다.

사족 하나. “따로 논리”를 지금까지 만들고 다듬은 핵심에는 CMU의 John Reynolds 교수, U of London의 O'Hearn 교수, 그리고 O'Hearn 교수와 같은 학교에 있는 양홍석(Hongseok Yang) 교수가 있다. 이 세 명은 “따로 논리”의 공헌으로 장차 컴퓨터과학에서 큰 일을 낼 것 같다는 것이 나의 주관적인 직감이다. 컴퓨터 분야는 아직 역사가 짧기 때문에 (혹시, 비유하자면 지금이 물리학이나 생물학의 19세기 초가 아닐까?) 우리와 지금 함께 살고있는 사람중에 “막스웰”도 있고 “퀴리부인”도 있는 것 같다. 장차 그런 사람들 중에 우리나라 사람들이 많으면 우리의 자부심은 더 근거있지 않을까. 보기도 좋을 것이다.

이 광근, 2009년 5월