

프로그램 분석과 죽음의 성물

김태은*

2022.05.19

Abstract

프로그램 분석의 자동성, 종료성, 정확성은 동시에 쟁취할 수 없는 꿈의 성질이다. 따라서 실제 프로그램 분석은 이 셋 중 하나를 포기함으로 가능해진다. 연구자들은 자신이 풀고자 하는 문제의 성질이 어떠한지를 바탕으로 어떠한 성질을 포기할 지 잘 선택해야 한다.

예로부터 지금까지 소설이나 전설에서 뭔가 좋은 것은 꼭 세 가지가 묶여서 전해진다. 해리포터의 죽음의 성물, 일본 신화의 삼신기, 단군신화의 천부인 등이 그렇다. 이러한 물건들은 보통 다 모으면 굉장한 힘을 발휘하지만 다 모으기가 굉장히 어려운 성격을 가진다. 프로그램 분석의 성질에도 이러한 3종 세트가 존재한다. 그것은 바로 자동성(Automatic), 종료성(Terminating), 그리고 정확성(Exact) 이다.

프로그램 분석은 현실의 문제지만 그 세가지 성질을 다 모으기가 아예 불가능하다는 점에서는 전설보다 더 전설같다. Rice의 정리에 의하면 프로그램의 모든 중요한 성질은 결정 불가능(Undecidable)하기 때문이다. 여기서 결정 불가능하다는 것은 프로그램을 자동으로 정확하게 분석한 후 종료하는 분석이 불가능하다는 것이다. 이 세가지가 동시에 가능하다면, 우리는 멈춤 문제(Halting Problem)를 비롯해서 수많은 난제들을 해결할 수 있을 것이다. 그야말로 컴퓨터 과학계의 죽음의 성물인 셈이다.

따라서 프로그램 분석이 필요한 경우 세 가지 중 하나를 포기한 방법을 택한다. 자동성을 포기하고 수동 작업으로 증명을 보조하거나, 종료성을 포기하고 무한히 테스트를 돌린다. 또는 정확성을 포기하고 그 의미를 뭉뚱그려 프로그램을 분석하기도 한다.

프로그램을 연구하는 입장에서는 자신이 풀고자 하는 문제의 성질을 잘 이해하고, 필요에 맞게 위의 세 가지 중 무엇을 포기해야할지 결정해야 한다. 프로그램의 의미가 결정 불가능하기 때문에 결국 사람에게 그 선택이 넘어오는 것이다. 예를 들어, 사용자가 전문성이 있다면, 수동 작업으로 증명을 보조하는

*KAIST 프로그래밍 시스템 연구실 석사과정

방법을 선택할 수 있다. 혹은 항상 종료한다는 보장이 없어도 괜찮거나, 경험적으로 그럴 가능성이 매우 작은 경우에는 종료성을 포기하고 모델 체킹을 할 수도 있다. 마지막으로 가끔 거짓 정보가 나오더라도 큰 비용 없이 확실한 결과가 필요한 경우는 정적 분석을 사용할 수 있다.

비록 프로그램을 자동으로 정확하게 분석한 후 종료하는 분석은 불가능하지만, 연구자들은 위와 같이 다양한 대안들을 제시하였다. 이렇게 제약 조건의 틈바구니에서 살 길을 도모하는 것이 바로 공학의 참 아름다움이 아닐까 한다.