

이해와 구현이 용이한 C 언어 시퀀스 포인트 모델

전웅^o 한동수
한국정보통신대학교
{woong^o, dshan}@icu.ac.kr

Comprehensive and Algorithmical Sequence Point Model for C

Woong Jun^o Dong-soo Han
Information and Communications University

요 약

이식성과 성능 사이의 타협을 위해 수식 평가 순서를 부분적으로 정의할 수 밖에 없는 C 언어는 표준이 사용하는 일반 영어의 모호성으로 일부 복잡한 수식의 유효성(validity)을 판별하기 어려운 문제를 안고 있다. 그동안 의존 관계나 언어 형식화(formalization)를 이용해 일관되게 유효성을 판별하려는 시도가 있었으나 이해와 구현 모두가 용이해야 한다는 표준의 요구를 만족하지 못했다. 본 연구에서는 구현을 고려해 AST(Abstract Syntax Tree)에 변수의 참조·수정 정보를 덧붙여 수식 부작용(side effect)과 관련된 시퀀스 포인트(sequence point) 문제를 해결하는 효율적인 모델을 제안한다.

1. 서 론

C 언어는 언어 정의를 일정 수준으로 규제해 고수준의 이식성을 제공하면서 동시에 기반 실행 환경의 특성을 최대한 활용할 높은 성능을 끌어낼 수 있도록 설계된 고급 언어로, 언어 사용자와 언어 구현자(implementer) 사이의 조율을 담당하는 주체는 ISO/IEC에 의해 제정된 언어 표준[1, 2]이다. 표준은 접근성(accessibility)을 위해 언어를 형식화하는 대신 BNF와 일반 영어를 사용해 문법과 의미를 정의하기 때문에 일부 경우에 표준이 제공하는 정의가 불분명한 경우가 발생하고 있다[3]. 일부 복잡한 형태를 갖는 수식의 유효성(validity) 판별 역시 그와 같은 경우에 해당한다.

명령형 언어(imperative programming language)인 C 언어로 작성된 프로그램은 본질적으로 실행 과정에서 변수값을 변화시키는 행동(action)을 명시하는 수식으로 구성된다. 하지만, 수식 평가에 일정 순서를 엄격하게 강요하는 경우 성능 저하가 불가피하기 때문에 표준은 문법적으로 정의되는 우선순위(precedence)와 결합방향(association)으로 수식의 부분적인 평가 순서만을 규정하고, 그 외에 피연산자 평가 순서와 부작용(side effect) 발생 순서에 대해서는 정의하지 않는다. 특히, 수식 내 피연산자나 부분 수식(sub-expression)의 평가가 부작용의 발생을 수반하지 않아야 한다.

단, 프로그램 내 일정한 지점에서 수식의 부작용이 완료되도록 하기 위해 이전 수식이 포함하는 부작용이 완료되고 다음 수식이 포함하는 부작용이 시작되지 않는 지점인 시퀀스 포인트(sequence point)를 적절한 위치에 배치하고 있다. 또한, 시퀀스 포인트를 통해 유효한 수식과 그렇지 않은 수식을 구분하기 위한 규칙(두 시퀀스 포인트 사이에서 한 변수는 최대 한번만 수정될 수 있으며, 수정되는 변수의 값은 저장될 새 값을 결정하기 위해서만 사용되어야 한다)을 제공하고 있지만, 영어 표현의 모호성으로 실제 프로그램에서 사용되는 일부 수식의 유효성을 올바르게 규정짓지 못하는 문제를 안고 있다.

시퀀스 포인트 모델(sequence point model)은 수식에 강제되

는 부분적인 순서만을 바탕으로 수식의 유효성을 명확하게 결정할 수 있는 규칙의 집합을 말한다. 보다 명확하게 정의된 규칙을 통해 컴파일러나 정적 분석 도구가 잠재적으로 이식성과 관련된 문제 혹은 직접적인 오류가 되는 부분을 사전에 지적할 수 있도록 하는 것이 가능하다.

초기 C 언어 수식의 유효성 판별에 대한 연구는 대부분 시퀀스 포인트에 대한 이해 부족과 명확한 규칙 부재로 기본적인 형태의 수식도 올바르게 판별하지 못했다[4]. 시퀀스 포인트 문제에 대한 인식으로 1999년 표준 개정 과정에서 수식의 가능한 평가 순서를 바탕으로 의존 관계를 파악하는 접근법을 시퀀스 포인트 모델로 제시했으나 모델이 갖는 문제로 최종 표준안에서는 배제되었다[5]. 또한, C 언어의 언어 정의 일부를 형식화하기 위한 접근으로 HOL (high-order logic)을 통해 C 언어의 데이터형 체계(type system)와 수식 체계를 형식화하는 시도도 있었다[6, 7]. 하지만 두 방법 모두 표준이 요구하는 중요한 두 특징인 이해의 용이성(특정 기술에 대한 전문 지식 없이도 이해가 가능해야 함)과 구현의 용이성(알고리즘의 형태를 갖추되 기존의 컴파일러나 분석 도구에 어렵지 않게 수용될 수 있어야 함)을 부분적으로 만족하지 못하고 있다.

본 연구에서는 수식의 AST(Abstract Syntax Tree) 표현을 바탕으로 일정 규칙을 기계적으로 따라 수식의 유효성을 판단할 수 있는 효율적인 접근법을 제안한다.

2. 꼬리표 AST(tagged AST)를 이용한 시퀀스 포인트 모델

2.1 가정

모델 제안에 앞서 문제를 간략화하고 중심이 되는 시퀀스 포인트 문제에 집중할 수 있도록 몇 가지 가정을 명시한다.

가정 1. 수식을 분석하는 과정에서 피연산자 및 연산 결과의 데이터형(type)은 고려하지 않는다. 데이터형 역시 수식의 유효성에 영향을 주지만 이는 시퀀스 포인트 모델과는