

C 해석기를 통한 C 프로그램 실행의 시각화

최근수, 하상호
순천향대학교 정보기술공학부
e-mail:keunsu77@hotmail.com, hsh@sch.ac.kr

Visualization of the execution of C programs using the C Interpreter

Keunsu Choi, Sangho Ha
Division of Information Technology Engineering,
Soonchunhyang University

요 약

최근 인터넷과 웹의 확산과 더불어 많은 교육용 웹 콘텐츠들이 개발되고 있다. 웹 기반의 교육은 언제 어디서나 교육을 받을 수 있다는 장점을 제공하나 실습이 중요한 프로그래밍 과목의 경우에 적합하지 못하다고 지적되어 왔다. 논문에서는 C 프로그램 실행의 시각화와 실행환경을 제공하여 C 언어를 쉽게 이해할 수 있게 하는 웹 콘텐츠 상의 C 프로그램의 실행의 시각화기를 개발한다. 이 시각화기는 사용자가 입력한 프로그램에 대해 구문 분석을 통해서 해석할 수 있는 C 언어 해석기와 해석된 정보를 사용자에게 시각화 하여 보여주는 시각화기로 구성된다. 논문의 결과로 기존의 프로그래밍 교육의 이론과 실습을 통합하여 제공함으로써 새로운 교육 환경이 구축될 수 있다.

1. 서론

최근 인터넷과 웹의 확산과 더불어 많은 대학에서 학내가상대학[1]을 운영하고 정식 학위를 수여하는 사이버대학[2]이 등장하여 많은 교육용 웹 콘텐츠를 제공하고 있다. 그러나 프로그래밍 관련 웹 콘텐츠 중 대부분의 강의 교안은 문법적인 부분에 대한 설명과 예제 중심으로 대부분의 교육이 진행되고 있는 현실이다.

학습자는 프로그래밍 실습 환경이 제공되지 않는 상황에서 이론만을 암기식 위주로 학습하고 있다. 이론적인 교육이 중요한 분야가 있지만 프로그래밍 관련 분야를 교육하는데 있어서는 프로그램 작성에 대한 실습이 무엇보다 중요하다. 그렇기 때문에 이론과 실습을 동시에 병행할 수 있는 교육 환경이 무엇보다 필요하다.

프로그래밍을 교육하는 콘텐츠중 C 언어는 대부분의 대학에서 첫 번째 프로그래밍 언어로 선택되고 있으나 C 언어는 시스템 프로그래밍 목적으로 개발되었기 때문에 시스템 세부 사항을 다룰 수 있고 포인터 등 낮은 수준의 기능을 포함하고 있어서 초보자가 배우기에는 어려운 언어이다.

본 논문에서는 웹 환경에서 학습할 수 있는 방법으로 C 프로그램 실행을 시각화하여 보여줄 수 있는 시각화기를 설계하고 구현한다. 웹상에서 학습자와 상호 응답하며, C 프로그램 실행의 시각화기를

통하여 학습자는 프로그램을 작성하고 실행의 시각화를 통하여 주어진 문제에 대하여 프로그래밍 할 수 있는 능력을 갖추도록 하는 것이 개발 목표이다. 인터넷과 웹 환경을 위해 개발된 Java Applet을 이용하여 학습자에게 익숙한 인터페이스를 제공하고 학습자가 C구문을 입력하여 실행하게 되면 입력된 C구문은 **Java Cup**[4]과 **JFlex**[4]를 통해 생성된 Parser(구문분석기)와 Lexer(어휘분석기)를 통하여 유효한 구문인지를 파싱(Parsing)하게 된다. 파싱과정을 통하여 얻어진 입력구문에 대한 정보를 시각화기를 통해 학습자의 실습 화면에 입력구문을 시각적으로 표현하여 나타나게 된다. 학습자는 실습 통하여 프로그램을 작성, 확인하게 됨으로써 이론교육을 뒷받침하게 되어 흥미롭고 효과적인 교육을 할 수 있다.

2. C 프로그램 실행의 시각화 방법

논문에서 고려하고 있는 C 프로그램 실행의 시각화 대상은 변수선언문/배정문/식, 제어문, 함수호출이다. 다음은 각 대상에 대해서 실행의 시각화 방법에 대해서 설명한다.

변수선언문/배정문/식

C언어에서 기본이 되는 구문은 각각 자료형과 이에 해당하는 변수를 선언하고 이에 값을 지정하는

것이다. 그림 1은 변수선언문, 식, 배정문에 대한 시각화를 보여준다. 사용자가 C문장 입력창에서 문장 입력시마다 입력된 문장에 대한 시각화가 즉각적으로 시각화창에 나타난다.

C 문장 입력창을 통해 변수가 선언될 때마다 시각화 창 하단에서처럼 변수가 자료형에 따라 기억공간의 크기가 다르게 할당되며 변수 이름이 레이블되어 시각화된다. 배정문의 경우 오른쪽의 식, 변수, 값이 평가되어 왼쪽의 변수의 기억공간의 값이 갱신되어 시각화 된다. 식의 경우 사용자는 연산자 우선순위에 의해 식이 평가되는 순서를 사용자의 실행요청에 의해 단계별로 평가되어 시각화한다. 예를들어, $x = a+b*c/3$;와 같은 배정문은 1단계[$b*c$]가 먼저 평가된 후 ①에서처럼 시각화되고 2단계[1단계/3]가 ②에서처럼 평가되어 1단계의 시각화에 추가되어 나타난다. 3단계[$a+1,2$ 단계]가 추가되어 ③에서처럼 시각화되고 마지막으로 식의 평가된 결과가 ④에서처럼 변수 x에 배정된다. 이를 사용자는 시각화창을 통해 확인할 수 있다.

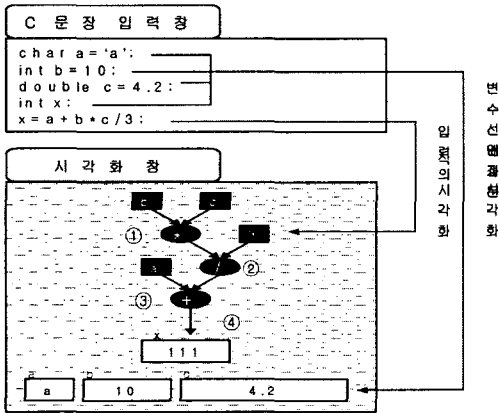


그림 1. 변수선언/배정문/식의 시각화

선택구조

선택구조는 일반적으로 조건식, then부분, else부분으로 구성된다. 실행의 의미는 조건식이 평가되고 그 결과에 따라 then 부분이나 else 부분의 문장을 실행하게 된다. 선택구조의 시각화는 사용자에게 익숙한 순서도의 도형을 이용하여 이루어진다. 가령, 그림 2는 입력된 if/else문에 대해 시각화창에서 순서도의 도형을 사용하는 것을 나타낸다. 이 문장에 대한 시각화는 다음과 같이 이루어진다.

사용자는 C문장 입력창을 통해 선택구조문장을 입력하게 되면 시각화창에는 입력문장과 그에 해당하는 순서도의 도형을 시각화하여 나타낸다. 화살표는 실행될 문장을 가리키게 된다. ①에서처럼 화살표는 실행될 문장과 순서도를 동시에 가리키며 선택구조의 조건문($n\%2 == 0$)을 평가하게 된다. 사용자의 다음 단계 실행 요청에 의해서 시각화 창 순서도에는 조건문의 평가 결과가 레이블되어 나타나며 ②에서처럼 조건문이 거짓일 경우 화살표가 입력문장에서는 else부분을 순서도에서의 오른쪽(else부분)을 동시에 가리키게 된다. 사용자의 다음 단계 실행 요청에 의해서 ③에서처럼 화살표는 해당 부분의 내부 문장을 가리키게 되고 해당문장을 실행하게 된

다. 내부문장 수행 종료시 ④와 같이 선택구조의 종료 표시를 나타낸다. 이처럼 입력문장과 순서도를 동시에 매칭 시켜가며 선택구조의 실행순서를 확인한다.

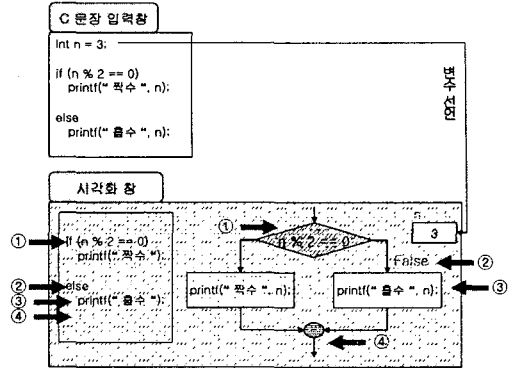


그림 2. 선택구조의 시각화

반복 구조

반복구조는 선택구조와 마찬가지로 조건식이 평가되고 평가결과가 거짓일 때까지 반복하여 입력문장과 순서도를 동시에 매칭 시켜 가던 실행을 확인한다. 그림 3에서 조건식 평가가 거짓일 때까지 ①, ②를 반복 수행하게 된다.

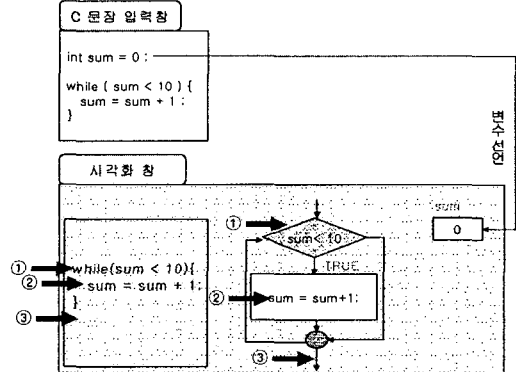


그림 3. 반복구조의 시각화

참조호출의 시각화

함수란 특정 작업을 수행하도록 짜여진 독립된 프로그램이다. 특정 작업을 반복적으로 수행해야 할 경우 사용자는 함수를 정의하여 필요할 때마다 호출하여 사용한다. 포인터 변수는 기억공간의 다른 변수의 번지를 갖는 변수이다. 함수호출과 포인터에 대한 시각화는 다음과 같이 이루어진다.

사용자는 함수 정의하는 창을 통해 함수를 정의하게 된다. C문장 입력창에서 정의한 함수에 대한 호출을 실행하게 되면 그림 4의 시각화 창에서 ①에서처럼 함수의 전체적인 구조를 시각화 창을 통해 시각화하여 나타나게 되며 인자로 a, b의 주소값이 전달될 경우 ②에서처럼 함수내의 포인터 변수가 인자의 기억공간의 번지를 화살표가 가리키며 시각화되어 나타난다. 사용자의 함수내부 문장의 단계별 실행 요청에 의해서 실행될 문장을 화살표를 통해 가리키며 실행되는 문장의 결과를 시각화 창을 통해 나타낸다. 이처럼 사용자는 함수 내부구조를 단계별로

실행하며 결과를 확인할수 있다.

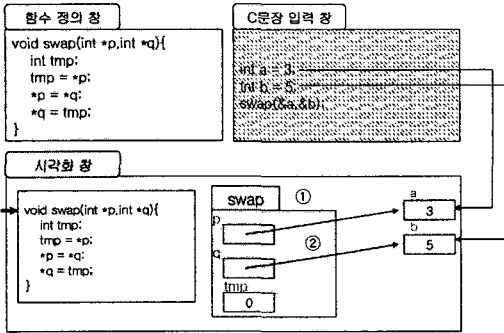


그림 4. 참조호출의 시각화

3. C 프로그램 실행 시각화기의 구조

그림 5는 프로그램 실행을 시각화 시키는 시각화기의 전체적인 구조를 보여준다.

C 프로그램 실행 시각화기는 C 해석기, 시각정보 클래스, 시각화기로 구성된다.

C 해석기는 Lexer와 Parser로 구성되어있다. 해석기는 사용자가 입력한 C 언어의 구문을 분석하기 위해서 스캐너생성기(scanner generator)인 JFlex와 파서생성기(parser generator)인 Java CUP을 사용하는데 JFlex에 의해서 생성된 스캐너(Lexer)는 입력 C 구문에 대한 문법적으로 의미를 가진 최소단위인 토큰을 생성해 주는 기능을 한다. 정규표현과 정규표현이 매칭 되었을 때 처리를 나타내는 액션코드로 작성된 입력 명세를 받아 상태전이 테이블과 그 테이블을 참조하는 입력 스트림을 처리하는 구동 루틴을 java 소스파일로 생성해준다. Java CUP은 문법 규칙에 대한 액션코드를 Java 프로그램 언어로 기술할 수 있도록 만들어졌고 BNF로 기술된 context-free 문법표현을 사용하여 파서의 기능기진 java 파일을 생성해 준다. 사용자로부터 C 구문을 입력받아 Lexer를 통해 토큰을 분리하고 Parser에 의해 구문분석과정을 거친다.

구문분석과정의 Action을 통하여 입력구문에 해당하는 시각 정보 클래스를 생성하고 시각화 하기위한 정보를 저장하고 있는 이 클래스들은 시각화기에 입력구문에 대한 정보를 전달한다.

시각화기는 사용자와의 상호응답을 위한 인터페이스와 사용자의 입력을 C해석기로 전달하는 인터페이스, 시각 정보 클래스의 정보를 추출하고 이를 Drawing모듈에서 처리하기 용이한 형태로 가공하는 Process모듈과 사용자에게 Process모듈을 거친 정보를 시각적인 표현으로 변환하여 시각화해주는 Drawing모듈로 구성되어 있다.

그림 5는 C구문 입력에서부터 사용자에게 시각화된 정보가 전달되는 과정을 나타내고 있다.

- ① 학습자는 C 구문을 시각화기를 통해 입력
- ② C구문은 구문분석을 위해 C해석기로 전달
- ③ 해석기 내부의 Lexer는 입력구문에 대하여 최소 의미를 가진 토큰으로 분리
- ④ 분리된 토큰은 Parser에 의해서 구문분석을 거치면서 구문의 유효성을 검사
- ⑤ 구문분석 또는 어휘분석시 에러(유효하지 않은

- 구 문)가 발생하면 이를 시각화기에 보고
- ⑥ Parser는 Action을 통하여 시각정보 클래스 생성
- ⑦ 시각정보 클래스는 시각화기로전달
- ⑧ Process모듈은 시각정보 클래스로부터 정보 추출하여 Drawing모듈로 전달
- ⑨ Drawing 모듈은 Process모듈을 통한 정보를 이용하여 입력구문에 대한 시각적인 표현을 생성
- ⑩ 학습자의 실습창으로 입력 구문에 대한 시각화 클래스(시각정보)

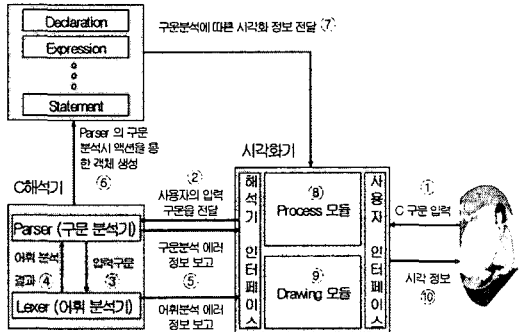


그림 5. C프로그램 실행의 시각화기 전체구조

4. 실행의 예

C 프로그램 실행의 시각화를 구조별 실행 예를 통해 구현 내용을 살펴본다.

변수 선언문 및 배정문

그림 6에서 학습자는 ②의 C문장 입력창을 통하여 실행하고자 하는 C 구문을 한 문장씩 입력, 실행하게 되면 ①의 History창에 실행된 문장은 이전에 실행된 문장에 추가되고 ④의 시각화 창에 이전에 시각화 되어진 정보에 추가되어 나타난다. 입력구문에 대한 에러 발생시 ③의 에러창을 통하여 에러메시지를 나타낸다.

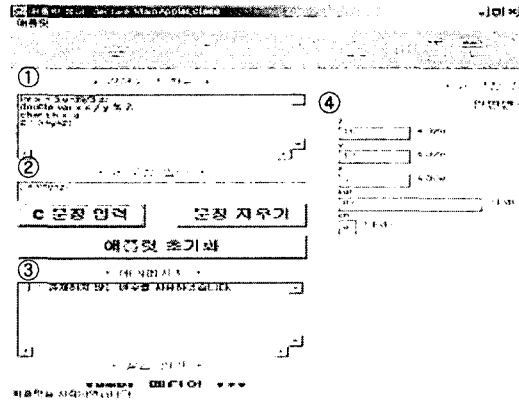


그림 6. 변수선언 및 배정문 실행 화면

식

그림 7에서 사용자는 식에 필요한 변수를 ①의 변수 선언창을 통하여 선언하게 되면 시각화 창 하단의 ⑤에서처럼 사용자에게 시각화 되어 나타나게 된다. 사용자가 식이 평가되는 순서를 단계별로 알아보고자 한다면 ②의 수식 입력창을 통하여 식을 입

력하며 시각화 창 상단의 ④에서처럼 입력한 식이 나타나고 시각화 창의 ⑥의 단계별 실행버튼을 통하여 입력수식에 대한 평가순서를 확인할 수 있다. 변수선언이나 입력 수식에 대한 ③의 에러창을 통하여 나타내게 된다. 학습자는 다양한 식을 입력하여 단계별로 평가되는 순서를 확인할 수 있다.

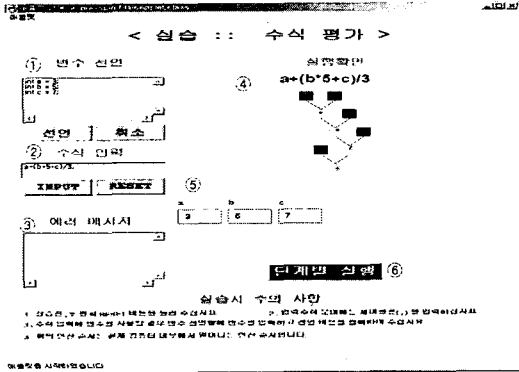


그림 7. 식 실행 화면

입출력에 대한 시각화

그림 8에서 학습자는 ①의 C 프로그램 입력창을 통하여 프로그램을 작성하여 입력하게 되면 시각화 창의 ②에서처럼 입력한 프로그램에 대한 시각화가 나타나고 ⑥의 단계별 실행 버튼으로 프로그램을 한 문장씩 실행해가면 실행결과를 확인한다. 시각화기 내부에 입출력 함수에 대한 처리모듈을 구성하여 scanf()함수는 시각화 창에 ④와같이 별도의 창을 통하여 사용자의 입력을 받고 printf()함수는 ⑤를 통해 출력결과를 나타낸다. 이와 같은 입출력 함수는 C프로그램에서 라이브러리로 제공되지만 본 논문에서는 입출력함수를 reserved word로 정의하여 해석기에서 인식하여 처리하였다.

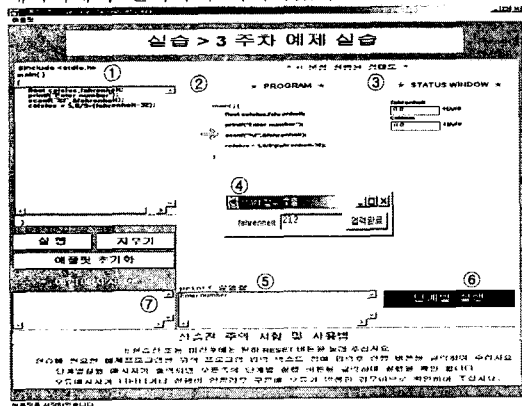


그림 8. 입출력에 대한 시각화 실행 화면

제어구조의 시각화

그림 9에서 사용자는 ①을 통하여 C구문을 입력, 실행하고 ②의 제어구조 입력창을 통하여 제어문을 입력 ③의 버튼을 통하여 제어구조를 입력한다. 입력한 제어문은 그림 10의 ④의 입력제어문과 함께 순서도가 시각화되어 나타나며 단계별로 ①의 조건이 평가된후 then 부분, else 부분의 해당 문장을 수

행하게 된다. 이를 제어문장과 순서도를 화살표를 통해 동시에 매칭 시켜 가며 나타낸다. 문장이 실행된 결과가 그림 10의 ⑤에서 보는바와 같이 출력됨을 확인할 수 있다.

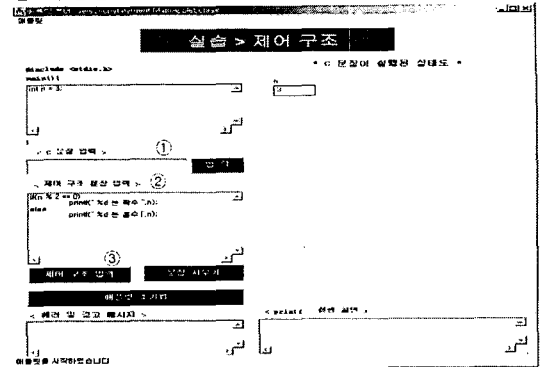


그림 9. 제어구조 입력 화면

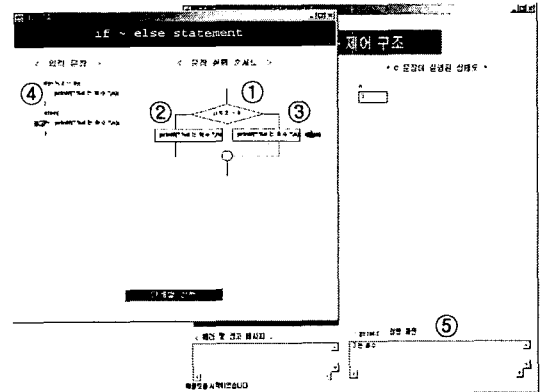


그림 10. 제어구조 입력에 따른 실행 화면

5.결론

본 논문에서 웹기반의 텍스트위주의 프로그램에 대한 교육내용과 실행환경이 분리되어 있는 문제점을 해결할 수 있는 C 프로그램의 시각화기를 설계하고 구현했다. 학습자는 학습 자료를 통하여 이론을 습득하고 습득한 이론에 대하여 실습을 통하여 이론과 실습을 병행해 가며 입력구문에 대하여 표현되어지는 시각화를 확인하는 과정에서 학습의 흥미와 교육효과를 높일 수 있다. 본 논문의 시각화는 모든 C구문에 대한 시각화를 제공하지는 않지만 C언어 초보학습자에게 C언어가 제공하는 기본적인 구문과 개념을 시각화를 통해 효과적인 학습이 이루어질 수 있도록 하기 위함이다.

참고문헌

- [1]학내가상대학 "http://sow.sch.ac.kr"
- [2]사이버대학 "http://www.ocu.ac.kr"
- [3]Kelley/Pohl "A Book On C" 3rd Ed. Addison-Wesley,1997
- [4]Robert W.Sebesta "Concepts of Programming Languages" Addison-Wesley,1999
- [5]Laura Lemay & Rogers Cadenhead "Teach Yourself JAVA 1.2" SAMS,1998
- [6]http://white.icu.ac.kr/cc/CUP0.10/cup_manual.htm