

그래픽 유저 인터페이스를 지원하는 웹 기반 프로그래밍 환경 ‘봄밭’의 설계 및 구현

BomBart : Web-based Programming Environment Support to Graphic User Interface

천준석*, 송지원**, 우 균***

부산대학교 전기전자컴퓨터공학과/LG전자 스마트 제어 센터*, 부산대학교 컴퓨터 및 정보통신연구소**,
LG전자 스마트 제어 센터***

Junseok Cheon(jscheon@pusan.ac.kr)*, Jiwon Song(jiwon@pusan.ac.kr)**,
Gyun Woo(woogyun@pusan.ac.kr)***

요약

최근 전세계적으로 프로그래밍 교육에 대한 관심이 증가하고 있다. 하지만 대부분의 언어는 컴파일러와 통합 개발 환경을 컴퓨터에 설치하여야만 사용할 수 있는 문제가 있다. 이를 해결하기 위해 Eclipse Che나 JDOODLE과 같은 웹 기반 프로그래밍 환경이 다수 개발되었지만 대부분이 GUI 프로그래밍을 지원하지 않고, 특히 한글 프로그래밍 언어는 지원하지 않는다. 이 논문에서는 ‘봄밭’이라고 하는 웹 기반 프로그래밍 환경을 제안한다. 봄밭은 한글 프로그래밍 언어인 새싹을 지원하고, GUI 프로그래밍도 지원한다. 콘솔 기반의 입출력 또한 지원한다. 콘솔 및 GUI 인터페이스를 모두 지원하기 위해서 봄밭에는 두 개의 서브 컴파일 시스템을 설계하고 구현하였다. 우리는 봄밭의 GUI 지원의 성능을 측정하기 위해서 Java의 모든 GUI 튜토리얼 코드를 새싹으로 변환하여 봄밭에서 실행해 보았다. 그 결과, 81.42%의 코드를 정상적으로 실행할 수 있는 것을 확인하였다.

■ 중심어 : 웹 기반 프로그래밍 환경 | GUI 프로그래밍 | 한글 프로그래밍 언어 | 새싹 | 봄밭 |

Abstract

There has been a growing interest in programming education recently. However, to use most programming languages the corresponding compiler and IDE have to be installed on computers. To tackle this issue, though there developed several web-based programming environment including Eclipse Che and JDOODLE, most of them does not support GUI nor Korean programming languages. This paper proposes a web-based programming environment called Bombart, which supports Saesark, a Korean programming language, with GUI output. It also supports a console-based input and output. To support both kinds of interfaces, two compiling subsystems are designed and implemented. To test the effectiveness of the GUI support of Bombart, all the Java tutorial codes on GUI are translated into Saesark and executed on top of Bombart. According to this test, 81.42% of codes can be successfully converted and executed.

■ keyword : Web-based Programming Environment | GUI Programming | Korean Programming Language | Saesark | Bombart |

* 이 논문은 부산대학교 기본연구지원사업(2년)에 의하여 연구되었음.

접수일자 : 2017년 01월 18일

심사완료일 : 2017년 02월 10일

수정일자 : 2017년 02월 09일

교신저자 : 우 균, e-mail : woogyun@pusan.ac.kr

I. 서론

최근 소프트웨어의 개발 분야가 다양해지고 이윤 창출의 기회가 증가하면서, 컴퓨터 프로그래밍에 관한 관심이 증가하고 있다. 교육부와 미래창조과학부는 2016년도 제12차 사회관계장관회의를 통해 ‘소프트웨어 교육 활성화 기본 계획’을 발표하였다[1]. 즉, 소프트웨어 교육의 종합적 기반을 마련하고, 교육 강화를 통해 소프트웨어 중심 사회의 국가 경쟁력을 확보한다는 것이다.

모국어로 이루어진 프로그래밍 언어는 초보자가 쉽게 접근할 수 있는 프로그래밍 언어이다. 한국의 경우, 80년대 후반부터 한글 프로그래밍 언어가 개발되어 왔다. 최근에도 새싹을 비롯한 몇몇 언어가 출시되었다.

하지만, 대부분의 프로그래밍 언어는 컴파일러와 통합 개발 환경을 로컬 컴퓨터에 설치하여야 사용할 수 있다는 문제가 있다. 한 대의 컴퓨터에서 소프트웨어를 개발할 때는 사용자가 해당 컴파일러와 통합 개발 환경을 설치하면 된다. 하지만 여러 대의 컴퓨터에서 단일 소프트웨어를 개발할 때는 모든 컴퓨터에 컴파일러와 통합 개발 환경을 설치하여야 한다. 이러한 설치 과정은 초보자에게 또 다른 진입 장벽으로 작용하고 있다.

이러한 비효율적인 작업을 줄여주는 방안으로 가상 데스크톱 인프라(Virtual Desktop Infrastructure)가 있다. 가상 데스크톱 인프라는 중앙 서버에 데스크톱 가상 머신을 운영한다. 이는 서버에 접속해서 작업하는 사용자가 데스크톱 형태로 작업할 수 있도록 한다. 하지만 소규모의 소프트웨어 기업이나 교육기관에서 가상 데스크톱 인프라를 구축하기에는 너무나 큰 비용이 든다.

가상 데스크톱 인프라와 유사한 환경으로 웹 기반 통합 개발 환경이 있다. 웹 기반 통합 개발 환경은 기존의 통합 개발 환경을 웹 사이트로 제공한다. 소프트웨어 개발자는 웹에 접속할 수 있다면 언제든지 해당 통합 개발 환경을 사용하여 프로그래밍할 수 있다.

현재 여러 사이트에서 제공되는 웹 기반 통합 개발 환경은 콘솔 프로그램을 컴파일하고 실행할 수 있다. 하지만 현존하는 웹 통합 개발 환경은 대부분 웹 애플리케이션의 기술적인 문제 및 보안 문제로 GUI(graphic

user interface) 프로그램을 컴파일할 수 없다. 또한, 컴파일이 가능하더라도 컴파일은 할 수 있으나 해당 실행 결과를 웹 사이트에서 직접 확인할 수 없다.

이 논문에서는 GUI 프로그램을 실행할 수 있는 웹 기반 통합 개발 환경인 봄밭을 제안한다. 봄밭은 한글 프로그래밍 언어 중의 하나인 새싹의 통합 개발 환경을 웹서비스의 형태로 제공한다. 봄밭은 새싹의 콘솔 프로그램을 실행할 수 있고 새싹 언어의 GUI 프로그램의 결과를 웹 사이트에 출력할 수 있다.

이 논문의 구성은 다음과 같다. 1장은 서론으로, 봄밭의 개발 필요성을 소개한다. 2장에서는 관련 연구에 대해 설명한다. 3장과 4장에서는 각각 봄밭의 설계와 구현에 대해 설명한다. 5장은 봄밭의 커버리지 실험을 수행하였고, 6장에서는 실험에 대한 토의를 실시하였다. 마지막으로 7장에서 이 논문에 대한 결론을 맺는다.

II. 관련 연구

1. 웹 기반 통합 개발 환경

이 절에서는 기존의 웹 기반 통합 개발 환경을 소개한다. 웹 기반 통합 개발 환경은 기존의 통합 개발 환경을 웹 서비스로 제공한다는 특징이 있다[2]. 즉, 웹에 접속하기만 하면 되므로 로컬 PC에 설치해야 사용할 수 있는 기존 통합 개발 환경의 불편한 점을 개선할 수 있다.

Eclipse Che는 Eclipse 재단에서 개발한 웹 기반 통합 개발 환경이다[3]. Eclipse Che는 Java 기반 통합 개발 환경인 Eclipse를 웹 사이트의 형태로 구현한 것이다. Eclipse Che는 Java, C, C++, Python, Ruby 등을 지원하며 해당 언어로 개발된 프로그램을 작성하고 실행한다. 하지만 Eclipse Che는 GUI 코드를 컴파일하거나 실행할 수 없다.

The First Online Collaborate Compiler는 Fixoncloud에서 만든 웹 기반 통합 개발 환경이다[4]. 해당 사이트는 자신의 계정에서 만든 프로젝트를 공유하는 것이 가능하고 컴퓨터에서 작업한 코드를 업로드하여 이를 서버에 저장하는 것도 가능하다. 하지만 The First Online Collaborate Compiler는 GUI 코드의 컴파일 및 실행은

불가능하다.

JDOODLE은 Nutpan.com에서 개발한 웹 기반 통합 개발 환경이다[5]. JDOODLE은 단순히 코드 편집 창을 제공하는 것이 아닌 버튼을 사용하여 프로그램의 함수 및 클래스를 정의한다. 하지만 JDOODLE에서는 GUI 코드의 컴파일 및 실행이 불가능하다.

[표 1]은 이 절에서 언급한 웹 기반 통합 개발 환경과 그 외의 통합 개발 환경을 정리한 것이다. 콘솔 프로그램 컴파일을 지원하는 웹 기반 통합 개발 환경 중 GUI 프로그램을 컴파일할 수 있는 경우가 별로 없는 것을 확인할 수 있다. 우리가 조사한 범위 내에서는 .net Fiddle만이 일부에 한해서 GUI 프로그램을 컴파일하고 실행할 수 있는 환경이었다.

표 1. 웹 기반 통합 개발 환경 비교

웹 기반 통합 개발 환경	콘솔 프로그램		GUI 프로그램	
	컴파일	결과실행	컴파일	결과실행
Eclipse Che	가능	가능	불가능	불가능
The First Online Collaborate Compiler	가능	가능	불가능	불가능
JDOODLE	가능	가능	불가능	불가능
.NET Fiddle	가능	가능	일부 가능	일부 가능
paiza.io	가능	가능	불가능	불가능
compileonline.com	가능	가능	불가능	불가능
codepad	가능	가능	불가능	불가능
ideone.com	가능	가능	가능	불가능

2. 한글 프로그래밍 언어

한글 프로그래밍 언어는 영어로 작성하는 대부분의 프로그래밍 언어와는 달리 한글로 작성하는 프로그래밍 언어이다. 한글 프로그래밍 언어의 연구는 1980년대부터 이루어졌으며[6], 초창기에는 기존의 프로그래밍 언어에서 키워드만 번역하는 것이 전부였다. 하지만 연구가 계속되면서 더욱 한글에 가깝게 작성할 수 있는 프로그래밍 언어들이 개발되고 있다.

표 2. 한글 프로그래밍 언어 비교

언어명	통합 개발 환경	한글 오류 메시지	컴파일러 단독 사용 여부
씨앗	존재	출력	불가능
약속	존재하지 않음	미출력	가능
와글	존재	출력	불가능
엔트리	존재	미출력	불가능
새싹	존재	출력	가능

새싹은 객체 지향 프로그래밍을 지원하는 한글 프로그래밍 언어이다[7][8]. 새싹은 자바 가상 머신(Java Virtual Machine, JVM) 상에서 동작하며, 객체 지향 프로그래밍과 람다 식을 지원한다. 또한, 콘솔에서 실행 가능하며 Eclipse IDE를 기반으로 하는 통합 개발 환경을 제공한다.

[표 2]는 기존 한글 프로그래밍 언어를 비교한 것이다. 조사한 바에 따르면, 새싹은 통합 개발 환경을 제공하고, 오류 메시지를 한글로 출력한다. 새싹은 씨앗[9], 와글[10], 엔트리[11]와 달리 통합 개발 환경 안이 아니라도 컴파일러를 사용할 수 있다. 또한, 약속[12]과 달리 오류 메시지를 한글로 보여준다. 따라서 우리는 봄밭 구현에 새싹의 컴파일러를 사용하였다.

III. 봄밭의 설계

이 장에서는 한글 프로그래밍 언어 새싹을 제공하는 웹 기반 통합 개발 환경 봄밭의 설계에 대해 설명한다. 봄밭은 크게 코드 입력 모듈, 코드 생성 모듈, 코드 실행 모듈로 구성된다. 사용자는 웹 브라우저를 통해 봄밭에 접속하여 프로그래밍을 할 수 있다.

1. 봄밭의 전체 구조

이 논문에서 제안하는 봄밭의 전체 구조는 [그림 1]과 같다. 사용자는 코드 입력 모듈에 새싹 언어 코드를 입력하고 실행 옵션을 선택한다. 실행 옵션은 콘솔 실행과 GUI 실행이 있다. 사용자는 프로그램의 종류에 따라 콘솔 또는 GUI를 선택해서 결과를 확인할 수 있다.

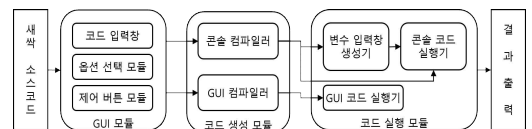


그림 1. 봄밭의 전체 구조

봄밭 서버는 코드 입력 모듈로부터 받은 새싹 코드를 입력받아 코드 생성 모듈을 실행한다. 입력받은 실행 옵션이 콘솔 실행이면 콘솔 컴파일러를, GUI 실행이면

GUI 컴파일러를 실행한 후 코드 실행 모듈에 실행 명령을 내린다. 만약 입력 코드를 실행할 수 없다면 오류 메시지를 출력한다.

코드 실행 모듈은 코드 실행 결과를 출력하는 기능을 수행한다. 콘솔 프로그램을 입력한 경우 코드 실행 모듈 내 콘솔 코드 실행기를 통하여 출력한다. 만약 콘솔 프로그램에서 표준 입력을 받는다면 콘솔 코드 실행기를 실행하기 전 변수 입력창 생성기를 사용하여 사용자에게 변수를 입력받고, 그 결과를 실행한다. GUI 프로그램을 입력한 경우 GUI 코드 실행기로 실행 결과를 사용자에게 출력한다.

[그림 2]는 봄발에 대한 유스케이스 다이어그램을 나타낸 것이다. 먼저 브라우저를 통해 봄발 서버에 접속한다. 이후에는 프로그램 설치 없이 봄발의 모든 기능을 쉽게 이용할 수 있다.

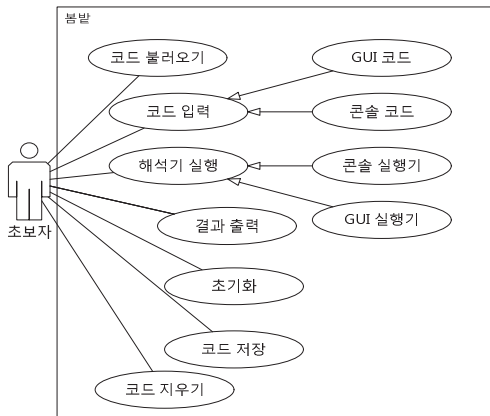


그림 2. 봄발의 유스케이스 다이어그램

2. 코드 입력 모듈 설계

코드 입력 모듈은 사용자로부터 새싹 언어 코드와 실행 옵션을 입력받아 이를 서버로 전송하는 역할을 수행한다. 코드 입력은 웹 페이지의 형태로 나타낸다. 코드 입력 모듈은 크게 코드 입력창, 옵션 선택 모듈, 제어 버튼 모듈로 구성된다.

코드 입력창은 사용자로부터 새싹 언어 코드를 입력받는다. 코드 입력창은 새싹의 예약어(keyword)를 강조하는 기능(syntax highlighting)을 지원한다. 옵션 선택 모듈의 경우 사용자가 콘솔 실행 모드와 GUI 실행

모드를 선택할 수 있게 한다. 제어 버튼 모듈은 코드 입력창에 입력한 새싹 코드를 제어하는 기능이다. 제어 버튼 모듈에는 실행, 초기화, 지우기, 저장하기, 불러오기 등의 기능이 있다.

3. 코드 생성 모듈 설계

코드 입력 모듈로부터 넘겨받은 새싹 코드는 코드 생성 모듈을 통해 컴파일된다. 콘솔 실행을 선택한 경우에는 콘솔 컴파일러를 실행한다. 또한, GUI 실행을 선택한 경우에는 GUI 컴파일러를 실행한다.

콘솔 컴파일러는 코드 생성 모듈에서 새싹의 콘솔 코드를 컴파일하고 실행하는 역할을 수행한다. 컴파일러는 먼저 코드 입력 모듈로부터 입력받은 새싹 코드를 분석하여 표준 입출력이 필요한지 판단한다. 만약, 입력 변수가 필요한 상황이면, 코드 실행 모듈 안에서 입력받은 코드를 분석하여 입력 변수의 프롬프트를 구별한 후 콘솔 코드 실행기를 수행한다.

GUI 컴파일러는 새싹 언어로 작성한 GUI 프로그램 코드를 컴파일하는 역할을 수행한다. 봄발에서 사용하는 GUI 컴파일러는 기존의 새싹 컴파일러에 GUI 패키지를 추가하여 GUI 코드를 실행할 수 있도록 하였다. 봄발에서 지원하는 주요 GUI 패키지는 [표 3]과 같다.

표 3. 봄발에서 지원하는 새싹 GUI 패키지 목록

새싹 패키지	설명
새싹.GUI,*	새싹 GUI 패키지
새싹.GUI,색	GUI 패키지 중 색 관련 패키지
새싹.GUI,글꼴	GUI 패키지 중 글꼴 관련 패키지
새싹.GUI,이미지	GUI 패키지 중 이미지 관련 패키지
새싹.GUI,이벤트	GUI 패키지 중 이벤트 관련 패키지
새싹.GUI,프린터	GUI 패키지 중 프린터 관련 패키지
새싹.GUI,버튼	GUI 패키지 중 버튼 관련 패키지
새싹.GUI,체크상자	GUI 패키지 중 체크상자 관련 패키지
새싹.GUI,그래픽	GUI 패키지 중 그래픽 관련 패키지
새싹.GUI,라벨	GUI 패키지 중 라벨 관련 패키지
새싹.GUI,텍스트창	GUI 패키지 중 텍스트창 관련 패키지

기존의 웹 기반 통합 개발 환경에서는 입력받은 코드를 웹 브라우저에서 그대로 실행할 수 없다. 하지만, JVM의 애플릿(Applet)을 이용하면 사용자의 허가 하에 웹 브라우저에 결과 코드를 출력할 수 있다[13]. 애플릿은 기존 데스크톱에서 지원하는 GUI 요소를 웹 브라우저에서 출력할 수 있게 한다. 새싹은 JVM을 기반

으로 하기 때문에 GUI용 새싹 컴파일러를 Java 애플릿을 사용하여 설계한다.

4. 코드 실행 모듈 설계

코드 실행 모듈은 코드 생성 모듈의 콘솔 또는 GUI 컴파일러에서 컴파일한 새싹 코드를 결과 출력 창에 출력하는 역할을 수행한다. 코드 생성 모듈에서 새싹 코드 컴파일러가 실패할 경우 코드의 오류 메시지를 결과 출력 창에 출력한다. 오류 메시지는 한글로 출력되며 새싹 컴파일러의 메시지를 출력한다.

콘솔 프로그램을 실행할 경우 모듈 내 콘솔 코드 실행기를 통하여 해당 결과를 결과 창에 출력한다. 표준 입력이 필요하면 변수 입력창 생성기를 통해 값을 입력 받는다. 이후 이를 이용해서 콘솔 코드 실행기를 실행하고, 그 결과를 결과 출력 창에 출력한다.

GUI 프로그램을 실행할 경우에는 GUI 코드 실행기를 사용한다. GUI 컴파일러를 통해 컴파일하면 .class 파일이 생성된다. 이를 웹 브라우저 상에서 실행하기 위해서는 Applet 내용을 포함하는 HTML 파일을 따로 생성해야 한다. GUI 코드 실행기는 .class 파일을 실행할 수 있는 HTML 파일을 생성하고 이를 실행하는 역할을 수행한다.

IV. 봄밭의 구현

봄밭은 한글 프로그래밍 언어 새싹을 지원하는 웹 기반의 실행 환경으로서 봄밭 내에서 새싹 소스코드를 작성하고 이를 실행할 수 있다. [그림 3]은 봄밭의 초기 실행 화면이다.

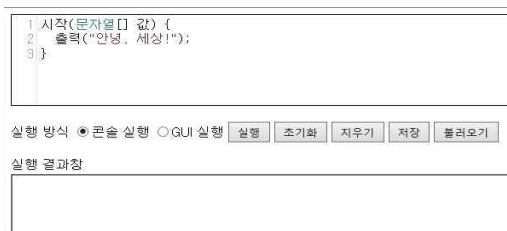


그림 3. 봄밭 초기 실행 화면

봄밭에서 사용한 새싹은 JVM을 기반으로 하고 있다. 따라서 JVM 기반의 웹 서버에서 구현하는 언어인 JSP와 해당 언어를 사용하는 웹 서버인 Apache Tomcat을 사용한다[14]. 웹 사이트는 HTML과 JavaScript 언어를 사용하여 구현하였다. 사용한 JavaScript 라이브러리는 파일 제어 라이브러리인 FileSaver.js와 텍스트 창 라이브러리인 CodeMirror이다[15][16].

1. 사용자 인터페이스 구성

봄밭은 코드 입력창, 제어 버튼 모듈, 결과 출력 창으로 구성된다. 코드 입력창은 사용자가 새싹 코드를 입력할 수 있는 창이다. 제어 버튼 모듈은 봄밭을 사용하기 위한 버튼이다. 코드 입력창으로부터 받은 실행 결과는 결과 출력 창을 통해 사용자에게 출력한다.

코드 실행은 코드 입력창에 입력한 새싹 코드를 실행한다. 초기화는 코드 입력창에서 입력한 새싹 코드를 초깃값으로 돌려놓는다. 지우기는 코드 입력창에서 입력한 새싹 코드를 지운다. 저장은 코드 입력창에 입력한 새싹 코드를 사용자의 컴퓨터에 저장한다. 불러오기는 사용자의 컴퓨터에 저장된 새싹 코드를 코드 입력창에 불러오는 역할을 수행한다.

2. 봄밭의 사용 시나리오

이 절에서는 이 논문에서 제안하는 봄밭의 사용 시나리오에 대해 설명한다. 봄밭은 크게 콘솔 프로그램 실행 기능과 GUI 프로그램 실행 기능으로 나눈다. 이 절에는 콘솔 프로그램 실행 시나리오, GUI 프로그램 실행 시나리오를 나누어 설명한다

사용자는 웹 브라우저를 통해 봄밭에 접속하여 코드를 입력한다. 콘솔 실행을 선택 후, 실행 명령을 내리면 인터페이스는 서버에게 사용자가 입력한 코드를 전송한다. 봄밭 해석기는 코드를 실행하고, 실행 결과를 저장한다. 입력 코드가 실행 가능할 경우에는, 서버는 인터페이스에게 입력창 출력 명령을 출력하고, 인터페이스는 사용자에게 입력창을 출력한다. 사용자가 변수를 입력하고 실행 명령을 내리면, 인터페이스는 서버를 통해 봄밭 해석기에게 입력 값을 전송한다. 봄밭 해석기에서 해당 입력 값을 사용한 프로그램의 결과를 실행한

후 그 결과를 사용자에게 출력한다. [그림 4]는 이를 시퀀스 다이어그램으로 나타낸 것이다.

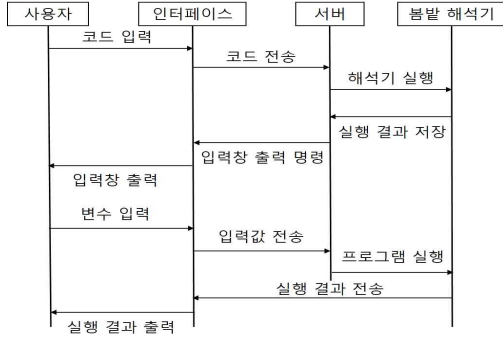


그림 4. bommal 콘솔 프로그램 실행 시나리오

[그림 5]는 bommal의 GUI 프로그램 실행 기능을 나타내는 시퀀스 다이어그램이다. 사용자는 인터페이스에 새싹 코드를 입력한다. 사용자가 인터페이스에게 실행 명령을 내리면 인터페이스는 서버에게 코드를 전송한다. 그 후 패키지 변환기에서 새싹 패키지를 Java의 패키지로 알맞게 변환한 후, bommal 해석기를 통해 해당 결과를 서버에 저장한다. 서버는 실행 결과를 인터페이스로 전송하여 사용자에게 출력한다.

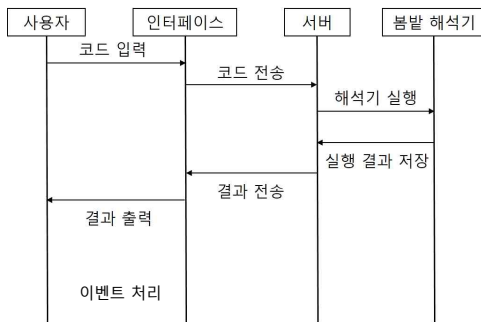


그림 5. bommal GUI 프로그램 실행 시나리오

3. bommal의 실행 결과

이 절에서는 이 논문에서 제안하는 bommal을 사용하여 새싹 코드를 실행한 결과에 대해 설명한다. [그림 6]은 표준 입력을 이용한 콘솔 프로그램을 실행한 결과이다.

해당 프로그램은 입력이 필요한 프로그램이기 때문에 실행할 경우 프로그램을 실행하기 전에 변수 입력 창이 실행 결과 창에 출력된다. 변수 입력창은 프롬프트와 변수를 입력하는 공간, 실행하기 버튼이 있다. 변수를 입력하고 실행하기 버튼을 누르면 입력된 변수에 따른 실행 결과를 실행 결과 창에 출력한다.

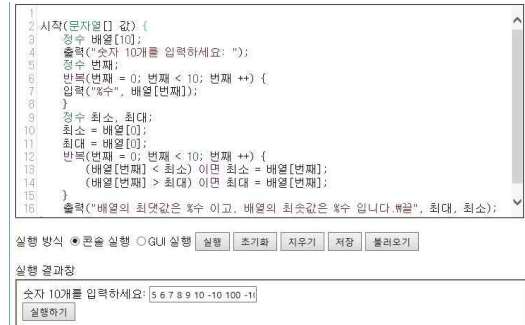


그림 6. 새싹 콘솔 프로그램의 실행 결과

[그림 7]은 GUI 프로그램 중 그래픽을 표현하는 새싹 프로그램을 실행한 결과이다. bommal에서는 해당 코드의 내용은 현재 시각을 아날로그 시계로 출력하는 내용이다. 실행 방식을 GUI 실행으로 설정하고 해당 코드를 실행하면 실행 결과 창에 아래와 같이 출력된다.



그림 7. 그래픽 출력 프로그램의 실행 결과

V. 커버리지 실험

이 장에서는 봄밭의 커버리지 실험에 대해 다룬다. 봄밭은 웹 기반 통합 개발 환경이므로 웹에서 GUI 코드가 실행된다. 따라서 웹에서 GUI를 실행하는 것을 중점으로 실험을 하였다. 실험에 사용할 예제 코드는 Java GUI와 2D 그래픽 예제를 새싹 코드로 변환한 것이다.

1. GUI 예제 실험

[표 4]는 Java GUI 예제 코드 중 봄밭으로 실행 가능한 새싹 코드의 비율을 보여준다. 실험에 사용된 코드는 Java 튜토리얼 사이트의 GUI 코드 156개를 새싹으로 구현한 것이다[17]. [표 4]에서 확인할 수 있듯이 Java GUI의 예제 코드 중 봄밭에서 실행할 수 있는 것은 총 126개이며, 원본 코드 중 총 80.77%를 봄밭에서 구현 가능한 것을 확인하였다.

표 4. 봄밭에서 실행 가능한 GUI 예제 코드

주제	코드 수	실행 가능 수	비율
컴포넌트	83	69	83.13%
레이아웃	23	19	82.61%
드래그 앤 드롭	8	6	75.00%
이벤트 리스너	16	13	81.25%
기타 GUI 기능	26	19	73.08%
계	156	126	80.77%

2. 2D 그래픽 예제 실험

이 절에서는 Java의 2D 그래픽 코드에 대한 실험을 수행하였다[17]. 2D 그래픽은 선과 면과 같은 2차원 요소를 출력하는 것을 가리킨다. Java로 작성된 2D 그래픽 코드를 새싹으로 변환한 것을 사용하였다.

표 5. 봄밭에서 실행 가능한 2D 그래픽 예제 코드

주제	코드 수	실행 가능 수	비율
도형 그리기	1	1	100.00%
텍스트 다루기	7	7	100.00%
이미지 다루기	6	5	83.33%
출력하기	4	3	75.00%
기타 예제	9	7	77.78%
계	27	23	85.19%

[표 5]는 Java 2D 그래픽 튜토리얼 코드 중 봄밭으로 실행 가능한 새싹 코드의 비율을 나타낸 것이다. [표 5]에서 확인할 수 있듯이 Java 2D 그래픽 예제 코드 중 봄밭으로 실행할 수 있는 새싹 코드는 총 23개로, 85.18%를 커버할 수 있다.

VI. 토의

두 실험을 진행한 결과, 총 183개의 예제 코드 중 149개의 코드(81.42%)가 봄밭에서 실행 가능한 것을 확인하였다. GUI 예제 코드의 경우에는 156개 코드 중 130개(80.77%)의 코드가 실행 가능하였다. 2D 그래픽 예제 코드의 경우에는 27개 코드 중 23개(85.19%)의 코드가 실행 가능한 것을 확인하였다.

GUI 예제 코드 중 제대로 실행되지 않는 코드에는 새 창을 띄우는 코드, 내부 프레임에 이용한 코드, 룩앤필(Look and Feel) 등이 있었다. 룩앤필은 프로그램의 외관이 어떻게 나타나며, 위젯이 어떻게 동작하는지를 가리킨다. 2D 그래픽 코드의 경우에는 새 창을 띄우는 코드와 내부 프레임에 사용하는 코드가 봄밭에서 작동하지 않았다.

새 창 띄우기, 내부 프레임 이용하기, 룩앤필 기능은 Java Swing 라이브러리를 일부 이용한다. 하지만, 현재 봄밭에서는 Java Swing 라이브러리를 지원하지 않는다. 그래서 해당 코드는 구현은 가능하지만, 컴파일 시 오류가 발생한다. 따라서 해당 결과를 브라우저를 통해 직접 확인할 수 없었다. 이는 앞으로 봄밭이 Java Swing 라이브러리를 지원하도록 만들어서 해결할 예정이다.

봄밭은 클라우드 환경이 아닌 웹 기반의 프로그래밍 환경을 제공한다. 클라우드 환경이나 웹 기반 환경은 둘 다 사용자 컴퓨터에 프로그래밍 개발 환경을 구축할 필요가 없다. 하지만 클라우드 환경을 구축하기 위해서는 여러 대의 서버가 필요하다. 또한, 네트워크 자원을 많이 요구하고, 경우에 따라서는 클라우드 환경에 대한 클라이언트를 사용자 컴퓨터에 설치해야할 수도 있다. 따라서 우리는 봄밭이 웹 기반의 프로그래밍 환경을 제공하도록 하였다.

VII. 결론

이 논문에서는 웹 기반 프로그래밍 환경인 봄밭을 제안하고 구현하였다. 봄밭은 한글 프로그래밍 언어인 새싹을 활용하여 콘솔 프로그램과 GUI 프로그램을 사용자에게 웹 브라우저를 통해 출력한다. 사용자는 로컬 PC에 프로그램 설치 없이 인터넷을 통해 쉽게 프로그래밍할 수 있다.

봄밭은 콘솔 프로그램과 GUI 프로그램의 출력을 지원한다. 기존의 웹 기반 통합 개발 환경에서는 웹 사이트에서 콘솔 프로그램을 실행할 수 있으나 대부분의 프로그래밍 언어는 GUI 프로그램을 실행할 수 없는 문제점이 있었다. 봄밭은 웹에서도 실행 가능한 GUI 요소를 한글 프로그래밍 언어 새싹을 사용하여 구현하였다. 봄밭은 새싹 언어의 콘솔 프로그램과 GUI 프로그램의 결과를 사용자에게 출력한다.

우리는 봄밭의 GUI 기능 커버리지를 측정하기 위하여 Java의 GUI 및 2D 그래픽을 구현하는 예제 코드를 새싹 코드로 변환하고 봄밭에서 동작하는지 실험하였다. 그 결과, 봄밭은 GUI 코드의 예제 코드 대부분(약 81.25%)과 2D 그래픽의 예제 코드 대부분(약 85.18%)을 제대로 실행할 수 있었다. 즉, 실험에서 사용한 183개의 예제 코드 중 약 81.42%에 해당하는 149개의 코드가 정상적으로 실행 가능한 것을 확인하였다.

향후 연구로는 GUI 패키지 이외에도 실제 Java에서 사용하는 패키지를 봄밭을 통하여 구현할 방안 등에 대해 생각해볼 것이다. 또한, 기존의 프로그래밍 언어 중 GUI를 지원하는 프로그래밍 언어를 봄밭으로 출력하는 방안을 연구할 예정이다. 봄밭은 GUI 프로그램을 웹에서 실행할 수 있으므로, 이를 활용하면 기존의 웹 기반 통합 개발 환경에서 GUI 프로그램 코드를 실행할 수 있을 것이라 기대한다.

참 고 문 헌

- [1] 교육부 홍보담당관, 「소프트웨어 교육 활성화 기본계획」 발표 - 초·중등 SW교육 필수화 준비 및 학교 중심의 SW교육 추진 -, 교육부, 2016.

- [2] T. Gaikwad, P. Dhavale, K. Gaware, and N. Shivale, "Web Based IDE," IJRIT International Journal of Research in Information Technology, Vol.2, No.4, pp.616-620, 2014.
- [3] <http://www.eclipse.org/che/>
- [4] <http://www.fixoncloud.com/Home/compiler/>
- [5] <https://www.jdoodle.com/>
- [6] 김영택, 김종상, 이석호, 조유근, 권혁철, "한글 프로그래밍 언어 설계에 관한 연구," 정보과학회 논문지, 제11권, 제2호, pp.81-101, 1984.
- [7] 천준석, 강도훈, 김건우, 우균, "간결한 한글 프로그래밍 언어 '새싹,'" 한국정보과학회논문지, 제42권, 제4호, pp.496-503, 2015.
- [8] 천준석, 우균, "새싹: 초보자를 위한 한글 객체 지향 프로그래밍 언어," 한국콘텐츠학회논문지, 제16권, 제3호, pp.288-295, 2016.
- [9] 임진희, *썬맛은 C가 아닙니다*, 성안당, 1995.
- [10] <http://waagle.org/>
- [11] <https://playentry.org/>
- [12] <http://yaksok.org/>
- [13] <http://www.oracle.com/technetwork/java/applet-s-137637.html>
- [14] <http://tomcat.apache.org/>
- [15] <https://github.com/eligrey/FileSaver.js/>
- [16] <https://codemirror.net/>
- [17] <https://docs.oracle.com/javase/tutorial/>

저 자 소 개

천 준 석 (Junseok Cheon)

정희원



- 2011년 : 동서대학교 컴퓨터공학과(학사)
- 2013년 : 부산대학교 전자전기컴퓨터공학과(석사)
- 2013년 ~ 현재 : 부산대학교 전기전자컴퓨터공학과(박사과정)

<관심분야> : 프로그램 시각화, 데이터베이스, 함수형 언어, 한글 프로그래밍 언어 개발

송 지원(Jiwon Song)

정회원



- 2014년 : 부산대학교 정보컴퓨터 공학부(학사)
- 2016년 : 부산대학교 전기전자컴퓨터공학과(석사)
- 2017년 ~ 현재 : 부산대학교 컴퓨터 및 정보통신연구소(전임연구원)

구원)

<관심분야> : 프로그래밍 언어, 함수형 언어, 웹프로그래밍, 한글 프로그래밍 언어 개발

우 균(Gyun Woo)

정회원



- 1991년 : 한국과학기술원 전산학(학사)
- 1993년 : 한국과학기술원 전산학(석사)
- 2000년 : 한국과학기술원 전산학(박사)

- 2000년 ~ 2004년 : 동아대학교 컴퓨터공학과 조교수
- 2004년 ~ 현재 : 부산대학교 전기컴퓨터공학부 교수

<관심분야> : 프로그래밍 언어 및 컴파일러, 함수형 언어, 로봇 프로그래밍, 소프트웨어 메트릭, 프로그램 분석, 프로그램 시각화, 한글 프로그래밍 언어 개발