

프로그래밍 초보자를 위한 비주얼 기반의 C 프로그래밍 통합 환경

김행임[†] · 박은경^{††} · 김현주^{†††} · 배종민^{††††}

요 약

프로그래밍 초보자는 C언어를 처음 배울 때 여러 가지 면에서 어려움을 겪는다. 그 이유 중의 하나는 C 언어 문법 이해에 대한 학생들의 학습 부담이 크고, 문제 해결보다는 문법에 치중하는 경향이 있다. 또한 소프트웨어 개발에 대한 흥미를 유발시킬 수 있는 프로그래밍 도메인이 필요하다. 본 논문에서는 문제 해결 중심의 C언어 교육을 지원하는 프로그래밍 환경을 제시한다. 이를 위하여 첫째, NXT 로봇 프로그래밍용 간단한 비주얼 프로그래밍 언어 Tiny-VPL을 정의하고, Tiny-VPL로 로봇프로그래밍을 할 수 있는 환경을 제시하며, 둘째 간단한 C 언어인 Mini-C언어를 사용하여 NXT 로봇 프로그래밍을 할 수 있는 환경을 제시한다. 그리고 Tiny-VPL을 Mini-C로 변환하며, 변환결과를 대화형으로 제공하는 그래픽 인터페이스를 제시한다. 개발된 프로그래밍 환경은 로봇 프로그래밍을 통하여 학생들의 흥미를 유발할 수 있고, 비주얼언어와 C언어 사이의 관계를 그래픽으로 제공하여 문제 해결 중심의 C 언어 교육에 효과적으로 활용될 수 있다.

주제어 : C 프로그래밍, NXT, Tiny-VPL, Mini-C, Eclipse, 통합 환경

An Integrated C Programming Environment for Novices Based on Visuals

Haeng-Im Kim[†] · Eun-Kyoung Park^{††} · Hyun-Ju Kim^{†††} · Jong-Min Bae^{††††}

ABSTRACT

Programming novices have various difficulties when they learn C language for the first time. Novices have a large burden for understanding of a C language grammar, and have a tendency to focus on the grammar rather than problem solving. Moreover, it requires programming domains to arouse student's interest for software development. This paper presents a programming environment for C languages education focusing on a problem solving. To this end, this paper defines Tiny-VPL that is a simple visual programming language for NXT robot programming and presents robot programming environment using Tiny-VPL. This paper also presents an environment for NXT robot programming using Mini-C language which is a subset of C language. For the purpose of helping to understand the C syntax and semantics, the visual and interactive conversion system of Tiny-VPL to Mini-C is provided. Our programming environment can arouse student's interest through robot programming and can be used effectively for C language education focusing on problem solving with graphical and interactive conversion of the visual language Tiny-VPL to the textual language Mini-C.

Keywords : C Programming, NXT, Tiny-VPL, Mini-C, Eclipse, Integrated Environment

† 정 회 원: 경상대학교 컴퓨터과학과 †† 정 회 원: 경상대학교 컴퓨터과학과
 ††† 정 회 원: 경남과학기술대학교 컴퓨터융합공학과 교수 †††† 정 회 원: 경상대학교 컴퓨터과학과 교수 (교신저자)
 논문접수: 2013년 07월 19일, 심사완료: 2013년 10월 24일, 게재확정: 2013년 10월 31일
 * 본 논문은 2012년도 경상대학교 연구년제연구교수 연구지원비에 의하여 수행되었음.

1. 서론

프로그래밍 언어 교육은 컴퓨터의 기본적인 동작원리를 학습하는 데에 도움을 주고, 논리적 사고력과 문제 해결력을 길러주기 때문에 컴퓨터 교과과정에서 필수 영역으로 자리 잡고 있다. 그럼에도 불구하고 C언어와 같은 고급언어를 처음 접하는 초보자들은 여러 가지 면에서 어려움을 겪는다. 그 중에 하나는 문법에 대한 기계적인 암기와 함께 코딩방법을 익히는 것에만 치중하기 때문에 시간이 지날수록 프로그래밍 학습에 대한 부담감이 커지고, 결국 학습 의욕을 떨어뜨리게 하는 결과를 초래한다. 또한 교수자의 일방적인 강의식 방식은 물론 단순한 코딩결과확인 수준의 실습방식도 문제이다. 이는 알고리즘 개발에 필요한 문제분석 능력과 논리적 사고력, 그리고 절차적인 문제 해결 능력을 기르는 데에 적합한 방법이 되지 못하고 있다[1].

이에 따라 문법에 치중하지 않고 알고리즘 개발에 집중하게 하여 프로그래밍 교육을 좀 더 쉽게 학습할 수 있는 교육환경에 대한 연구가 활발히 진행 중에 있다. 대표적으로 Scratch와 NXT-G 등과 같은 비주얼 기반의 프로그래밍 언어를 이용한 학습방법을 예로 들 수 있다[2][3][4]. Scratch와 NXT-G 프로그래밍 언어는 블록(아이콘)들을 화면에 배치하여 프로그램을 작성하는 그래픽 환경을 제공한다. 학습자는 세세한 문법에 신경 쓰지 않아도 되기 때문에 문제 해결 과정에 좀 더 집중할 수 있다[1][2][4]. 그러나 그래픽 기반의 환경에 의존해서 프로그래밍 언어 교육을 받는 것에는 한계가 있다. 블록배치에 의해 자동으로 생성되는 내부적 코드가 사용자의 의도와 다르게 실행될 수 있으며, 규모가 큰 프로그램일수록 블록배치만을 이용하여 해결할 수 없기 때문이다. 따라서 그래픽 기반의 환경과 함께 텍스트 기반의 프로그래밍 환경도 필수적으로 익혀야 한다.

초보자들에게 학습효과를 높이기 하기 위해서는 프로그래밍 환경에 대한 흥미와 관심을 유발하게 하고 이를 지속시키게 하는 것이 무엇보다도 중요하다. 그러나 프로그램을 코딩하고 이를 실행하여 단순히 텍스트 기반의 실행결과를 확인

하는 기존의 메커니즘은 학습자의 흥미를 떨어뜨릴 수 있는 요인이 된다. 이를 해결하기 위하여 레고 마인드스톰 로봇을 활용하는 사례들이 늘고 있다. 본인이 작성한 프로그램이 로봇을 통해서 실제로 어떻게 동작되어지는지 확인해 봄으로써 프로그램에 대한 관심과 함께 학습동기를 유발시킬 수 있다.

이에 따라 본 연구에서는 초보자가 쉽게 프로그래밍 언어를 학습할 수 있는 비주얼 기반의 통합 환경을 개발하였다. 본 논문에서 제시하는 통합 환경은 다음과 같은 기능을 가지고 있다.

첫째, 언어의 문법적 어려움을 덜기 위하여 Scratch와 비슷한 형태이지만 기능을 간소화한 Tiny-VPL을 정의하였고, 이를 사용하여 프로그래밍 할 수 있는 환경을 제공한다.

둘째, 프로그래밍 학습에 대한 관심과 흥미를 유발할 수 있도록 NXT 로봇 응용 프로그램을 개발할 수 있는 환경을 갖추고 있다.

셋째, 텍스트 기반 언어인 C언어를 사용하여 NXT 로봇 프로그래밍을 할 수 있도록 기능을 간소화한 Mini-C를 정의하였고 이에 대한 컴파일러를 지원한다.

넷째, 그래픽 기반의 언어인 Tiny-VPL을 텍스트 기반의 언어인 Mini-C로 변환되는 변환기를 제공한다.

다섯째, C언어 문법 이해에 도움이 될 수 있도록 Tiny-VPL의 각 컴포넌트가 실행될 때마다 그에 대응되는 Mini-C언어 코드를 대화형으로 보여주는 인터페이스를 개발하였다.

여섯째, 이 모든 기능이 이클립스 플러그인 기능을 통해 이클립스 환경에서 통합되어 사용하게 함으로써 시스템을 보다 편리하게 사용할 수 있도록 하였다.

본 논문의 구성은 다음과 같다. 2장에서 관련된 연구를 살펴보고, 3장에서 전체 시스템 구조에 대해서 설명한다. 4장에서는 레고 마인드스톰 XNT를 소개하고, 5장에서는 프로그램 생성도구의 설계 및 구현을 제시한다. 마지막으로 6장에서는 결론 및 향후 과제에 대해서 논한다.

2. 관련연구

교육용 프로그래밍 언어는 비주얼 기반 프로그래밍 언어와 텍스트 기반 프로그래밍 언어로 나눌 수 있다. 비주얼 기반 프로그래밍 언어의 대표적인 예로 초등교육 수준에서 사용할 수 있는 Scratch, Etoy Alice가 있고, 중등교육 수준에서의 NXT-G, Robolab 그리고 고등교육 수준에서 사용할 수 있는 NXT-MSRDS, Webot, Sinulink 등이 있다. 한편 텍스트기반 프로그래밍 언어는 Greenfoot, Phrogram 등과 같은 초보자용과 NXJ, NXC, RobotC와 같은 보다 고학년에서 사용할 수 있는 언어가 있다. 특히 NXC는 대학의 입문과정에서 널리 사용되는 C언어와 유사한데 이는 프로그래밍 초보자들에게 다소 어려울 수 있다.

프로그래밍 초보자들에게 C언어가 어려운 것 이외도 기존의 프로그래밍 응용영역은 흥미를 끌기 어려웠다. 따라서 프로그래밍 초보자들이 학습 동기를 부여하고, 학습 의욕을 높이기 위한 목적으로 프로그래밍 입문 교육에서 레고 마인드스톰 로봇을 이용하여 교육방법을 개선하기 위한 많은 연구가 있다. <표 1>은 로봇을 이용한 프로그래밍 교육 관련 주요 연구 결과이다[1][2][5][6][7].

<표 1> NXT를 이용한 C 프로그래밍 교육 연구

연구자	연구 분야	주요 연구 내용
김승환	레고 마인드스톰을 이용한 C언어 교육	레고 마인드 스톰 기반의 Robotic Invention Systems을 이용하여 학생들에게 C언어를 교육하기 위한 아이디어를 제안한다. 대학 신입생들에게 프로그래밍을 위한 기본적인 능력을 키우는 데 효과적임을 입증함.
배상용	로봇을 이용한 C 프로그래밍 교육	프로그래밍 학습에서 로봇을 이용한 수업이 전문계 고등학생들의 학업 성취도에 미치는 영향을 알아보고 프로그래밍 학습에 로봇의 적용 가능성을 검증함
조용만	레고 로봇을 활용한 프로그래밍 교육 모형	학생들의 효과적인 프로그래밍 교육을 위해 Robolab 프로그램과 C언어의 제어문을 이용하여 프로그래밍 학습 활동을 위한 새로운 학습 모형을 개발함
홍기천	프로그래밍 언어 수업 방안	레고 마인드스톰 NXT를 활용하여 프로그래밍 수업을 재미있게 진행할 수 있는 예비교사의 프로그래밍 언어 수업 방안에 대한 고찰

한편 프로그래밍에 대한 흥미를 돋우기 위하여 로봇을 활용하여 교육하는 연구 뿐 만 아니라 로

봇을 이용하여 초보자들이 프로그래밍을 잘 할 수 있도록 도와주는 프로그래밍 환경 개발에 대한 연구가 있다.

Robotran은 초보자도 쉽고 편리하게 NXT 로봇 프로그래밍을 할 수 있는 프로그래밍 환경이다[8]. Robotran은 C언어를 사용하는 로봇 프로그래밍이 프로그래밍 초보자들에게 어렵다고 판단하여, Robolang이라는 텍스트 기반의 간단한 언어를 정의 하였다. 그리고 고학년 학생들을 위하여 자바 기반의 NXT 로봇을 동작할 수 있는 언어인 Lejos로 프로그래밍을 할 수 있도록 지원한다. 또한 2-D 그래픽 로봇 시뮬레이터를 제공한다.

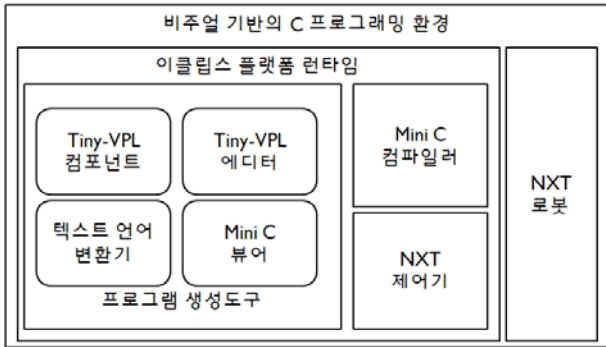
또 다른 교육용 프로그래밍 환경으로서 컴파일러를 보다 쉽게 개발할 수 있는 환경이 있다 [9][10][11]. 그 중에 [9]는 간단한 C언어에 대한 레퍼런스 컴파일러를 제공하며, 뿐만 아니라 컴파일러 제작도구, 목적 언어 테스트 도구, 코드 생성 시각화 도구가 포함되어 있다. 이때 사용자의 흥미를 돋우기 위하여 NXT 로봇을 사용하였는데, 레퍼런스 컴파일러의 목적코드는 로봇을 동작시키는 NBC 코드를 사용한다. 또한 이 프로그래밍 환경은 이클립스 플러그인 기반으로 개발되어 편의성과 확장성이 뛰어나다.

본 논문에서 제시하는 C언어 프로그래밍 환경에서는 비주얼 언어인 Scratch와 유사한 언어인 Tiny-VPL을 정의하고 이 비주얼 언어를 사용하여 프로그래밍을 할 수 있는 프로그램 생성도구를 제공한다. 특히, Tiny-VPL으로 작성된 프로그램을 C언어 코드로 변환하여 사용자에게 대화형으로 보여줌으로써 텍스트 언어에 대한 이해를 돕는다. 그리고 초보자들의 흥미를 높여주기 위해 NXT 로봇 프로그래밍 환경을 제공하여, 프로그래밍 결과는 NXT 로봇을 이용하여 확인할 수 있다. 또한 제시된 프로그래밍 환경은 이클립스 플러그인 기반으로 개발하여 사용의 편의성과 시스템 확장성을 중시하였다.

3. 시스템 구조

<그림 1>은 본 논문에서 제시하는 프로그래밍 환경의 전체적인 시스템 구조를 간략하게 제시한 것이다. 본 시스템은 크게 프로그램 생성도구,

Mini-C컴파일러, NXT제어기, NXT 로봇으로 구성된다.



<그림 1> 시스템 구조

프로그램 생성도구는 본 논문에서 Scratch와 유사하게 정의한 비주얼 기반 프로그래밍 언어인 Tiny-VPL을 이용하여 프로그램을 작성하는 도구이다. 이는 Tiny-VPL 컴포넌트, Tiny-VPL 에디터, Mini-C 뷰어, 텍스트 언어 변환기로 구성된다.

Tiny-VPL 컴포넌트는 Tiny-VPL의 문법을 컴포넌트 단위로 표현한 것이며, Tiny-VPL 에디터는 Tiny-VPL 컴포넌트로 프로그램을 편집 할 수 있는 작업영역이다. Tiny-VPL 에디터로 작성된 프로그램은 텍스트언어 변환기에 의해 Mini-C로 변환되고, 이는 Mini-C 뷰어를 통해 나타난다.

Mini-C 컴파일러는 프로그램 생성 도구로부터 생성된 Mini-C 코드를 로봇이 동작할 수 있는 중간 코드인 Mini NBC 코드로 변환한다.

NXT 제어기는 Mini NBC를 NXT 로봇이 실제로 사용하는 바이트코드로 변환한다. 또한 변환된 바이트 코드를 NXT 로봇에 업로드 하여 로봇을 동작하도록 한다.

NXT 로봇은 로봇 프로그램을 실행하는 소형 컴퓨터인 인텔리전트 브릭과 입력을 담당하는 센서들, 출력을 담당하는 모터, 그리고 골격을 만드는 레고 블록들로 구성되며, 이를 이용하여 로봇을 직접 설계 및 조립하여 만든다.

본 논문에서 제시하는 시스템은 NXT 제어기와 Mini-C 컴파일러는 기존에 연구된 결과[9]를 그대로 활용하며, NXT 제어기, Mini-C 컴파일러, 프로그램 생성도구는 이클립스 플러그인 프레임

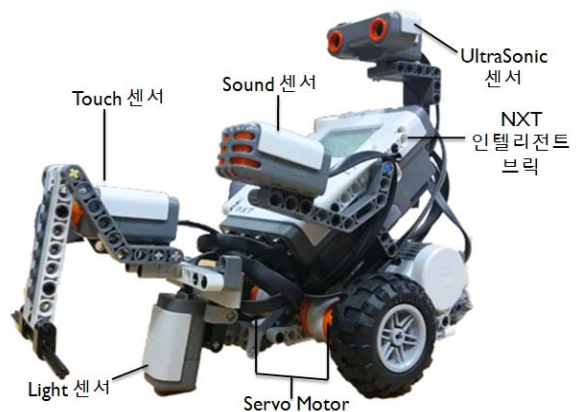
워크를 기반으로 구현되어 이클립스 플랫폼에서 수행된다.

4. 레고 마인드스툼 NXT 개요

레고 마인드스툼은 레고 부품으로 여러 가지 로봇을 만들고 로봇프로그램을 통하여 로봇을 다양하게 동작시킬 수 있다[12]. 레고 마인드스툼 NXT는 인텔리전트 브릭과, 4개의 센서, 3개의 서보모터, 그리고 로봇의 형태를 만들 수 있는 레고 블록들로 구성된다.

레고 마인드스툼 NXT의 인텔리전트 브릭은 로봇의 두뇌 역할을 하며 프로그래밍 된 내용에 따라 각종 센서에서 수집된 정보를 분석하여 로봇의 행동을 제어한다.

센서는 초음파, 빛, 사운드, 터치 센서 등이 있다. 초음파 센서는 로봇과 물체와의 거리를 측정할 수 있다. 빛 센서는 빛의 세기를 측정할 수 있고 서로 다른 색상을 구별할 수 있다. 사운드 센서는 소리의 패턴과 톤을 인식할 수 있다. 터치센서는 로봇의 장애물이나 물체와의 접촉 상태를 감지하는 센서이다[13]. 또한 NXT에서 사용되는 모터는 내장 센서를 통해 속도를 제어할 수 있는 서보 모터가 사용된다.



<그림 2> 레고 마인드스툼 NXT 로봇

<그림 2>는 본 프로그래밍 환경에서 사용된 NXT 로봇의 모습이다. 프로그래밍 환경을 사용하는 대상이 초보자이기 때문에 다양한 프로그램을 작성할 수 있도록 4가지 센서를 모두 사용하였고, 로봇을 다루는 데에 어렵지 않도록 로봇의

동작은 전진, 후진, 회전, 대기, 멈춤이 가능하도록 설계하였다.

NXT로 만든 로봇은 중간코드인 NBC 코드를 이용하여 동작할 수 있는데, 본 시스템은 그 기능을 축소한 Mini NBC를 사용한다. Mini NBC는 로봇을 제어하기 위하여 제시된 프로그래밍 환경에서는 <표 2>와 같이 시스템 함수와 상수로 구성된 API를 이용하여 로봇 하드웨어를 제어한다.

<표 2> Mini NBC API

기능	시스템 함수	시스템 상수
센서 제어	SetSensorLight, SetSensorSound, SetSensorTouch, SetSensorUltrasonic, ReadSensor, ReadSensorUS	IN_1,IN_2,IN_3,IN_4
서보모터 제어	Off, OnFwd, OnRev	OUT_A, OUT_B, OUT_C, OUT_AB ,OUT_AC, OUT_BC, OUT_ABC

5. 프로그램 생성 도구 설계 및 구현

5.1 시스템 설계

5.1.1 Tiny-VPL

본 프로그램 생성 도구에서는 간단한 비주얼 기반 프로그래밍 언어를 사용한다. 이 프로그래밍 언어의 이름은 Tiny-VPL이라고 한다. Tiny-VPL은 google blockly[14]의 일부 기능을 사용해서 정의하였다.

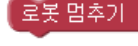



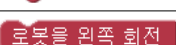



Tiny-VPL은 컴포넌트를 이용하여 문법을 표현한다. 컴포넌트의 종류는 기본 컴포넌트, 변수 컴포넌트, 사칙연산 컴포넌트, 논리연산 컴포넌트, 제어 컴포넌트, 로봇제어 컴포넌트가 있다. 정의된 Tiny-VPL의 모든 컴포넌트는 한글로 표현함으로써 학생들이 컴포넌트에 대한 이해를 도울 수 있도록 설계하였다.

<표 3>은 기본 컴포넌트이다 여기에는 프로그램 시작 컴포넌트와 프로그램 끝 컴포넌트가 있다. 프로그램 시작 컴포넌트는 프로그램을 작성할 때 가장 먼저 배치하여 프로그램 작성을 시작하는 컴포넌트이고 프로그램 끝 컴포넌트는 가장 마지막에 배치한다.

<표 3> Tiny-VPL 기본 컴포넌트

컴포넌트 모양	설명
	프로그램의 작성을 시작하는 컴포넌트
	프로그램의 작성을 종료하는 컴포넌트

<표 4> Tiny-VPL 로봇제어 컴포넌트

컴포넌트 모양	설명
	로봇을 멈추게 하는 함수를 호출하는 컴포넌트
	로봇을 대기시키는 함수를 호출하는 컴포넌트
	로봇을 전진시키는 함수를 호출하는 컴포넌트
	로봇을 후진시키는 함수를 호출하는 컴포넌트
	로봇을 왼쪽으로 회전시키는 함수를 호출하는 컴포넌트
	로봇을 오른쪽으로 회전시키는 함수를 호출하는 컴포넌트
	터치센서를 NXT 입력 포트와 연결하는 함수를 호출하는 컴포넌트
	터치센서로부터 데이터를 읽어오는 함수를 호출하는 컴포넌트

<표 4>는 로봇의 행동을 제어하는 컴포넌트들이다. 로봇 멈추기 컴포넌트는 로봇을 멈추는 내장 함수를 호출하는 컴포넌트으로써 로봇 모터의 동작을 멈추게 한다. 로봇 대기 컴포넌트는 로봇의 동작을 일정시간 멈추는 내장 함수를 호출한다. 이때 로봇 대기 컴포넌트는 다른 컴포넌트를 연결하여 멈추는 시간을 설정할 수 있는데, 숫자 상수 컴포넌트와 변수호출 컴포넌트를 로봇대기 컴포넌트에 연결할 수 있다. 로봇 전진 컴포넌트와 로봇 후진 컴포넌트는 로봇의 전진 또는 후진을 제어하는 내장 함수를 호출한다. 이 컴포넌트들도 숫자상수 컴포넌트, 변수지정 컴포넌트를 연결하여 전진 또는 후진 속도를 설정할 수 있다. 또한 로봇 오른쪽 회전 컴포넌트와 로봇 왼쪽 회전 컴포넌트는 로봇을 오른쪽 또는 왼쪽으로 회전하도록 제어하는 내장 함수를 호출한다.

입력센서 연결 컴포넌트는 입력센서인 터치 센서, 사운드 센서, 빛 센서, 초음파 센서를 NXT

인텔리전트 브릭의 입력포트와 연결하는 내장 함수를 호출한다. 이때 입력센서 연결 컴포넌트의 입력센서와 포트번호를 클릭하여 다른 입력센서나 포트 번호를 선택할 수 있다.

입력센서 컴포넌트는 입력센서로부터 입력된 정보의 값을 읽어오는 내장 함수를 호출하는 컴포넌트이다. 이때 입력된 자료의 값을 저장할 목적으로 사용될 매개변수 이름을 지정하는 컴포넌트를 포함할 수 있다.

기본 컴포넌트와 로봇제어 컴포넌트를 제외한 컴포넌트들은 google blockly[14]에서 정의된 컴포넌트의 일부를 사용하며, 이때 사용되는 컴포넌트는 <표 5>와 같다.

<표 5> 그 외 컴포넌트

컴포넌트 모양	설명
	이름이 item인 변수를 선언하고 값을 설정하는 컴포넌트
	이름이 item인 변수를 반환하는 컴포넌트
	수식의 값을 정수형으로 설정하는 컴포넌트
	두 피연산자 컴포넌트의 덧셈 연산을 하는 컴포넌트
	두 피연산자 컴포넌트의 뺄셈 연산을 하는 컴포넌트
	두 피연산자 컴포넌트의 곱셈 연산을 하는 컴포넌트
	두 피연산자 컴포넌트의 나눗셈 연산을 하는 컴포넌트
	두 피연산자 컴포넌트의 값이 같은지 비교하는 컴포넌트
	두 피연산자 컴포넌트의 값이 다른지 비교하는 컴포넌트
	앞의 피연산자 컴포넌트의 값이 뒤의 피연산자 컴포넌트 값 보다 크거나 같은지 비교하는 컴포넌트
	앞의 피연산자 컴포넌트의 값이 뒤의 피연산자 컴포넌트 값 보다 작거나 같은지 비교하는 컴포넌트

	㉠에 연결된 컴포넌트의 값이 참일 때 ㉡과 연결된 컴포넌트를 수행하는 컴포넌트
	㉠에 연결된 컴포넌트의 값이 참일 때 ㉡과 연결된 컴포넌트를, 거짓일 때 ㉢과 연결된 컴포넌트를 수행하는 컴포넌트
	㉠에 연결된 컴포넌트의 값이 참일 동안, ㉡에 연결된 컴포넌트를 반복 수행하는 컴포넌트

5.1.2 Tiny-VPL 에디터

Tiny-VPL 에디터는 Tiny-VPL 컴포넌트를 이용하여 프로그래밍을 할 수 있는 작업 영역이며, 작업 화면은 <그림 3>과 같다. Tiny-VPL 에디터는 컴포넌트 영역과 프로그램 편집 영역으로 구성되어 있다. Tiny-VPL에서 제공하는 모든 컴포넌트들이 성격별로 분류되어 왼쪽의 컴포넌트 영역에 배치되어 있다. 사용자는 원하는 컴포넌트들을 오른쪽의 프로그램 편집 영역으로 가지고 와서 원하는 위치에 배치 또는 제거하면서 프로그램을 작성해 나간다.



<그림 3> Tiny-VPL 에디터

5.1.3 Mini-C 문법

본 논문에서 정의한 Mini-C언어는 NXT C언어의 문법을 바탕으로 정의되었으며 C언어와 유사한 문법을 가지고 있는 텍스트 기반 프로그래밍 언어이다. <그림 4>는 Mini-C언어의 문법을 BNF로 표현한 것이다.


```

<program> ::= <nxt_control>
<nxt_control> ::= <var_decl> int main() <comp_stmt>
<var_decl> ::= <dcl_list> | ε
<dcl_list> ::= <dcl_list> <decl> | <decl>
<decl> ::= int <var_list>
<var_list> ::= <var_list> , NAME | NAME
<stmt> ::= <asgn_stmt> | <if_stmt> | <call_stmt>
| <while_stmt> | <comp_stmt>
<asgn_stmt> ::= NAME = <exp>;
<if_stmt> ::= if ( <cond> ) <stmt> | if ( <cond> ) <stmt> else <stmt>
<while_stmt> ::= while ( <cond> ) <stmt>
<call_stmt> ::= NAME( <param_list> );
<param_list> ::= <exp_list> | ε
<exp_list> ::= <exp_list> , <exp> | <exp>
<comp_stmt> ::= { stmt_list }
<stmt_list> ::= <stmt_list><stmt> | <stmt>
<cond> ::= <exp> >= <exp> | <exp> <= <exp>
| <exp> != <exp> | <exp> == <exp>
<exp> ::= <exp> + <term> | <exp> - <term> | <term>
<term> ::= <term> * <factor> | <term> / <factor> | <factor>
<factor> ::= NAME | NUMBER | HEXA | (<exp>)
    
```

<그림 4> Mini-C BNF

5.1.4 텍스트 언어 변환기

텍스트 언어 변환기는 비주얼 기반 프로그래밍 언어인 Tiny-VPL을 텍스트 기반 프로그래밍 언어인 Mini-C 코드로 변환한다. 각각의 Tiny-VPL 컴포넌트는 컴포넌트 목적에 맞게 정의된 코드 변환 메소드를 호출하면서 Mini-C 코드로 변환한다.

5.1.5 Mini-C 뷰어

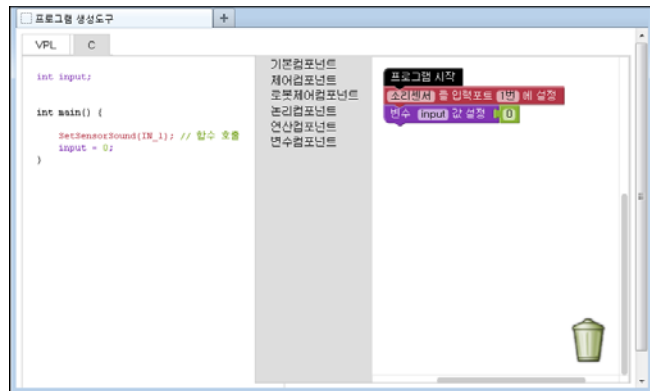
Mini-C 뷰어는 Mini-C 코드를 보여주는 영역이다. 이때 학습자의 이해를 돕기 위하여, Tiny-VPL과 Mini-C 코드는 대화형으로 변환될 뿐만 아니라 Tiny-VPL 컴포넌트와 변환된 Mini-C 코드는 같은 색상으로 일치하여 보여준다. 또한 변환된 Mini-C 코드는 가독성을 높이기 위하여 들여쓰기를 하여 나타난다.

5.2 구현

제시된 프로그램 생성도구를 이용하여 실제 프로그램을 작성하는 과정을 살펴보자. 작성할 프로그램은 “로봇이 30의 속도로 전진하다가 소리가 감지되면 30의 속도로 후진하라”는 기능을 수행하는 프로그램이다. 이때 소리센서로부터 입력받은 소리의 크기 값이 50 이하이면 소리를 감지하지

않은 것으로 가정한다.

프로그램 작성을 시작하기 위해 Tiny-VPL 에디터의 “기본컴포넌트” 메뉴에서 “프로그램 시작” 컴포넌트를 선택하여 프로그램 편집 영역에 배치한다. 이때, <그림 5>와 같이 Mini-C 뷰어에는 자동으로 “int main() { }” 코드가 검정색으로 나타난다.

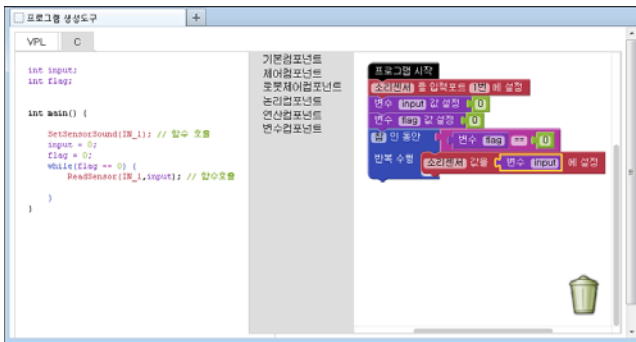


<그림 5> 프로그램 시작과 변수 선언

계속해서 소리센서를 입력포트 1번에 설정하고, 소리센서를 통해 입력받은 소리 값을 저장하기 위한 변수 input 설정하고 0으로 초기화한다. 이 과정을 수행하기 위해 사용자는 “로봇제어컴포넌트” 메뉴에서 입력센서를 포트에 설정하는 컴포넌트를 가지고 와서 배치하고 입력센서의 종류를 소리센서로, 그리고 입력포트는 1번으로 설정한다. 설정과 동시에 Mini-C 뷰어에는 “SetSensorSound(IN_1)” 코드가 빨간색으로 나타난다. 그 다음으로 “변수컴포넌트” 메뉴에서 변수설정 컴포넌트를 가지고 와서 배치시키고, 변수의 이름을 “input”으로 입력한다. 이 과정이 끝나면 Mini-C 뷰어에는 “int input”이라는 변수 선언문이 보라색으로 나타난다. 그리고 “연산컴포넌트” 메뉴에서 수식 컴포넌트를 가지고 와서 변수 설정 컴포넌트의 오른쪽에 연결시키고, 값을 0으로 설정한다. 이 과정이 끝나면 Mini-C 뷰어에는 “input=0”이라는 코드가 생성되며, 이때 “input=”은 보라색으로, “0”은 연두색 값으로 나타난다.

<그림 6>은 로봇이 계속해서 소리 감지를 시도하기 위하여 반복문을 작성하는 과정을 나타낸다. “제어컴포넌트” 메뉴에서 반복문 컴포넌트를

가지고 와서 배치시키고, 반복문 조건을 설정하기 위하여 논리연산 컴포넌트를 연결시킨다. 반복문의 조건이 참일 동안에 수행해야 할 수행문으로서 소리 센서로부터 입력받은 정보를 "input" 변수에 저장하기 위해 소리 센서 컴포넌트와 변수 지정 컴포넌트를 차례로 배치시켰다. 이와 같은 배치과정이 끝이 나면 Mini-C 뷰어에는 파란색으로 "while() { }" 반복문 코드가 나타난다. 그리고 변수명인 "flag", 논리연산자인 "==", 상수인 "0" 은 각각 보라색, 분홍색, 연두색으로 while문의 괄호 () 안에 나타난다. 반복문이 참일 때 수행되는 코드인 "ReadSensor(IN_1,input);"는 input은 보라색으로, 이외의 코드는 빨간색으로 while문의 중괄호 {}안에 나타난다.



<그림 6> 반복문 설정

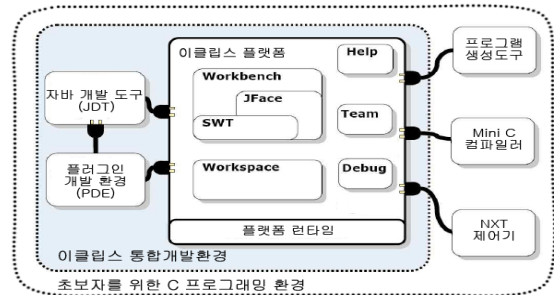
이와 같은 과정을 통해 완성된 프로그램은 <그림 7>과 같다. 사용자가 각 컴포넌트들을 Tiny-VPL 에디터의 컴포넌트 영역에 하나씩 가지고 와서 배치할 때마다 텍스트 변환기의 수행을 통해 대화형으로 그에 대응하는 C 언어의 문법 구문이 Mini-C 뷰어에 순차적으로 나타난다. 이때 Mini-C 코드는 배치된 컴포넌트와 같은 색상으로 나타나며, 가독성을 높이기 위하여 들여쓰기를 하여 나타난다.

예제 프로그램에서 보는 바와 같이 본 프로그래밍 환경은 프로토타입으로 구현되었다. 여기에는 식, 변수, 제어문, 함수호출 등의 기능이 포함되어 있는데, 배열, 포인터, 레코드 등과 같은 자료구조와 함수정의 기능을 지원하지 않은 한계가 있다.



<그림 7> 완성된 프로그램

본 논문에서 제시한 비주얼 기반 C 프로그래밍 환경은 이클립스 플러그인 프레임워크를 적용하여 개발되었다. JAVA 모듈 형식인 Jar 파일을 이클립스에 플러그인하여 이클립스 통합개발환경과 하나의 프로그램으로 통합할 수 있다. 제시하는 C 프로그래밍 환경의 구성요소 중 프로그램 생성도구는 Mini-C컴파일러, NXT 제어기와 함께 <그림 8>과 같이 이클립스 통합개발환경과 하나의 시스템으로 통합된다.



<그림 8> 이클립스와 통합된 C 프로그래밍 환경

5. 결론 및 향후 연구

프로그래밍 초보자가 C언어 프로그래밍을 배울 때 C언어 자체의 문법에 어려움을 느끼며 그 결과 알고리즘 개발에 집중하기 어렵다. 또한 대부분의 프로그래밍 도메인은 초보자들의 흥미를 유발하기 어렵다.

이러한 어려움을 덜기 위하여 본 논문에서는 초보자를 위한 비주얼 기반의 C언어 프로그래밍 환경을 개발하였다. 제시된 C 프로그래밍 환경은 간단한 비주얼 기반 프로그래밍 언어를 제공하고,

비주얼 언어의 컴포넌트가 추가될 때 마다 그에 대응되는 C언어 코드를 대화형으로 변환하여 컴포넌트마다 색깔을 달리하여 보여준다. 또한 로봇 프로그래밍 환경을 구축하여 소프트웨어 개발에 관한 동기유발 가능성을 높였다.

제시된 프로그래밍 환경은 이클립스 플랫폼에서 플러그인 형태로 개발되었기 때문에 기능 확장에 용이하고 배포가 쉬워 지속적인 유지, 보수가 가능하다.

앞으로 배열, 포인터, 구조체 등을 포함하는 C 언어의 더 많은 기능을 구현하고 수업에 활용하여 그 교육적 효과를 분석할 필요가 있으며, 또한 자바와 같은 객체지향형 언어 프로그래밍 환경을 지원하도록 그 기능을 확장할 필요가 있다.

참 고 문 헌

- [1] 김태희 · 강문설 (2010). 레고 마인드스톰 로봇을 이용한 프로그래밍 입문 교육의 효과 측정. 한국인터넷정보학회논문지, 11(4), 159 - 173.
- [2] 홍기천 (2009). 레고 NXT 로봇을 활용한 예비교사의 프로그래밍 언어 수업 방안. 한국정보교육학회논문지, 13(1), 71-78.
- [3] 조성환 · 송정범 · 김성식 (2008). 스크래치를 이용한 프로그래밍 수업 효과. 한국정보교육학회논문지, 12(4), 375-383.
- [4] 양권우 (2010) 스크래치가 프로그래밍 교육에 대한 학습동기 및 학업성취도에 미치는 영향. 정보교육학회논문지, 14(4), 547-553.
- [5] Kim S. H., & Jeon, J. W. (2006). Educating C Language using LEGO Mindstorms Robotic Invention System 2.0. Proceedings of the 2006 IEEE International Conference on Robotics and Automation(ICA 2006).
- [6] 배상용 (2009). 로봇을 이용한 C 프로그래밍 교육이 학업 성취도에 미치는 영향. 석사학위 논문, 한국교원대학교.
- [7] 조용만 (2007). 레고(LEGO)로봇을 활용한 컴퓨터 프로그래밍 교육모형 개발. 석사학위논문, 연세대학교.
- [8] Meyer, R. M., & Burhans. D. T. (2007). Robotran: A Programming Environment for Novices Using LEGO Mindstorms Robots. *Florida Artificial Intelligence Research Society conference(FLAIRS-07)*.
- [9] 성우경 · 강현석 · 배종민 (2011). 이클립스 기반의 교육용 컴파일러 통합개발환경, 컴퓨터교육학회논문지, 14(5), 9-18.
- [10] Li Xu, & Fred G. Martin (2006). Chirp on Crickets: Teaching Compilers Using an Embedded Robot Controller. SIGCSE '06.
- [11] Li Xu (2007). RobotStudio: A Modern IDE-based Approach to Reality Computing. SIGCSE '07.
- [12] Lego, Inc.. "8547 LEGO® MINDSTORMS® NXT 2.0". <http://mindstorms.lego.com>
- [13] 문외식, 유승한, 성영훈 (2008). LEGO MINDSTORMS NXT로 로봇 만들기. ALCO.
- [14] Google Inc (2012). google blockly A visual programming editor. <http://code.google.com/p/blockly/>



김행임

2011 경상대학교
컴퓨터과학과(학사)
2013 경상대학교
컴퓨터과학과(석사)

관심분야: 프로그래밍언어, 컴퓨터교육, 로봇교육
E-Mail: kimhangim@naver.com



배종민

1980 서울대학교
수학교육과(학사)
1983 서울대학교
계산통계학과(석사)

1995 서울대학교 계산통계학과(박사)
1997~1998 Virginia Tech. 객원연구원
1984~현재 경상대학교 컴퓨터과학과 교수
관심분야: 컴파일러, 프로그래밍언어, 컴퓨터교육
E-Mail: jmbae@gnu.ac.kr



박은경

1999 경상대학교
컴퓨터과학과(학사)
2002 경상대학교
컴퓨터과학과(석사)

2007 경상대학교 컴퓨터과학과 (박사)
관심분야: 컴퓨터교육, XML, DB 통합
E-Mail: pek1028@daum.net



김현주

1988 경상대학교
컴퓨터과학과(학사)
1990 숭실대학교
전자계산학과(석사)

2000 경상대학교 컴퓨터과학과 (박사)
2002~현재 경남과학기술대학교 컴퓨터융합공학과 교수
관심분야: 정보검색, 모바일 프로그래밍
E-Mail: khj@gntech.ac.kr