한국컴퓨터정보학회 동계학술대회 논문집 제20권 제1호 (2012. 1)

LabVIEW OOP를 활용한 프로그래밍 교육의 적용 방안

정민포°, 조혁규*, 정덕길**
°*영산대학교 사이버경찰학과, **동의대학교 컴퓨터과학과
e-mail:{minpo, hgcho3}@ysu.ac.kr, dgjung@deu.ac.kr

The Application to the Programming Education Using LabVIEW OOP

Min-Po Jung°, Hyuk-Gyu Cho*, Deok-Gil Jung**
°*Dept. of CyberPolice, YoungSan University
**Dept. of Computer Science, DongEui University

● 요 약 ●

일반적으로 프로그램을 배우는 초보자들은 텍스트 기반으로 프로그램을 하는 것이 매우 어렵다. 이러한 문제를 쉽게 풀기 위한 한 방법이 Visual C++, Visual Basic, Delphi와 같은 비주얼 프로그램으로 구현하는 것이다. 그러나 이러한 프로그램을 배우는 학습자들 역시, 비주얼 컴포넌트 작동에 대한 텍스트 프로그래밍을 어려워하고 있다. 논문에서는 이러한 문제를 해결하기 위해 논리적 사고를 표현하고 객체지향을 지원하기 위해 UML을 도입하고 텍스트 프로그래밍 요소를 비주얼 프로그래밍 요소로 대체하기 위해 객체지향을 지원하는 LabVIEW OOP를 사용하여 학습자들에게 프로그래밍 교육을 하는 방법을 제시하였다. 또한, 제시된 프로그래밍 교육 방법에 대해 설문조사를 실시하여 교육적인 효과를 분석하였다.

키워드: UML(Unifiled Modeling Language), 랩뷰 OOP(LabVIEW OOP), 프로그래밍교육(Programming Education), 비주얼프로그래밍(Visual Programming)

I. 서 론

컴퓨터 언어를 배우는 학습자는 다양한 분야에 적합한 프로그래밍 언어를 배우고 텍스트 기반으로 된 프로그래밍을 하는 것이 매우 어렵다고 생각한다. 이러한 문제를 쉽게 풀기 위한 한 방법이 프로그램을 비주얼로 표현하는 것이다. 기존의 시각 프로그래밍인 Visual C++, Visual Basic, Delphi와 같은 비주얼 언어는 외부 인터페이스는 비주얼 컴포넌트로 표현되고 컴포넌트의 작동에 대해서는 텍스트 기반으로 표현한다. 이러한 프로그램을 배우는 학습자들은 컴포넌트 작동에 대한 텍스트 프로그래밍에 대해 어려워하고 있으며 프로그래밍을 싫어하는 한 요소가 되었다[1]. 이러한 문제를 해결하기 위해 논리적 사고를 표현하면서 객체지향을 지원하기 위해 UML을 도입하고 텍스트 프로그래밍 요소를 비주얼 프로그래밍 요소로 대체하기 위해 객체지향을 지원하는 LabVIEW OOP를 사용하여 학습자들에게 프로그래밍 교육을 하는 방법을 제시하였다. 또한, 제시된 프로그래밍 교육 방법에 대해 설문조사를 실시하여 교육적인 효과를 분석하였다.

II. 본 론

컴퓨터 프로그래밍 언어 분야에서 프로그래밍 언어란 논리적 사고의 표현 도구라고 표현하였다[2]. 프로그래밍 언어는 단순히 프로그래머의 프로그램 작성 도구로서만 교육되어서는 안 되며, 논리적인 사고를 계발하고 표현하는데 중점을 두고 교육이 되어야 한다. 이 논문에서는 기존의 프로그래밍 교육 방식과 다른 개선된 교육 방식이 필요하다. 특히 논리적 사고를 표현하기 위해 체계화되고 검증된 소프트웨어 설계도구인 UML을 통해 프로그램에 대한 논리적인 사고를 효율적으로 표현하고, LabVIEW OOP와 같은 비주얼 개발 도구를 이용하여 구현하는 교육 방식을 제안한다[3][4].

1. UML과 LabVIEW OOP/JAVA를 활용한 교육 적용 방안

프로그래밍 언어 교육현장에서 그래픽 언어인 LabVIEW OOP와 텍스트 언어인 JAVA를 이용한 프로그래밍 언어 교육에 대한 적용 단계는 그림 1과 같다.

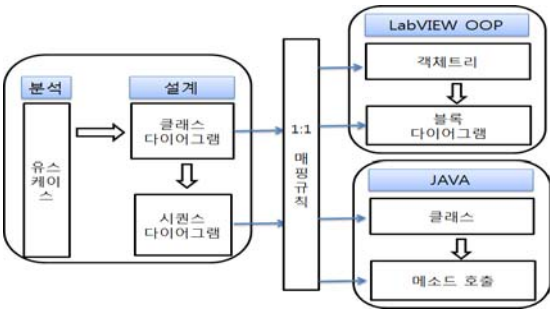

그림 1. UML 다이어그램의 LabVIEW OOP 클래스 및 JAVA 클래스의 1:1 매핑

2. 교육효과 적용에 대한 설문 방법

이 논문에서는 RFID의 FA(Factory Automation) 프로그램 개발에 대하여 교육하고, LabVIEW OOP 방법과 JAVA로 개발하는 것을 실험하였다. 5일의 교육을 통해 학생들에게 예제를 사용하여 교육을 실시하였다. 교육 실시 후, 표 1에 따르는 설문조사를 실시하였다. 그림 2에서 과정을 도식화 하였다.

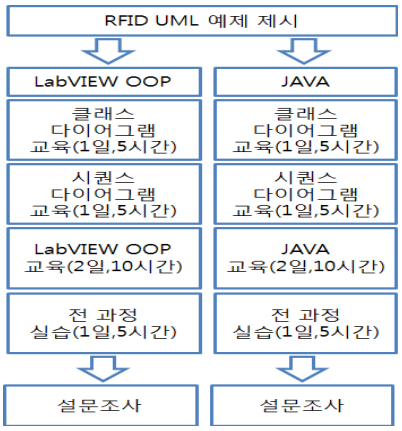

그림 2. 교육 적용 과정

표 1. 설문조사 양식(축약)

도구(언어) 표기법 / 평가 항목	UML	LabVIW OOP	JAVA
요구사항 분석			
프로그램 설계			
프로그램 구현(코딩)			
유지보수 가능성			
확장성			
재사용성			

설문 조사 대상은 일반 IT 전공학과 3학년, 4학년(시스템/보안 전공, C언어, JAVA 가능함) 40명을 대상으로 조사를 하였다.

UML과 JAVA를 잘못하는 학생들을 대상으로 미리 사전 교육을 5일 정도 실시하여 클래스 다이어그램, 시퀀스 다이어그램, 유스케이스 다이어그램을 이해하고 기본적인 것을 작성하는 수준이다. 이 논문에서 설문한 3가지 교육방법(UML, LabVIEW OOP, JAVA)에서 설문 문항별로 만족도의 집단 평균치에 차이를 자료 분석하였다.

17개의 설문문항은 6개의 평가 문항으로 분류하였다. 각 평가 문항의 분석에서 설문 문항별로 UML, LabVIEW OOP, JAVA의 평균값을 차트로 표시하였다. 본 연구의 자료 분석은 IBM SPSS Statistics 19를 사용하였으며 5점 척도의 채점표 분석에 평균분석을 이용하였다. 채점표의 5점 척도로 '매우 쉽다' 5점, '쉽다' 4점, '보통' 3점, '조금 어렵다' 2점, '매우 어렵다' 1점으로 하였다. 실증적 분석 방법은 일원분산분석법을 활용하여 분석하고 신뢰도 95%의 범주에 포함되면 유의성 있다고 해석하였다. 어떤 교육방법들이 평균차이를 보이는지에 대한 사후검증을 위해 Scheffe를 사용하였다.

III. 결 론

프로그래밍 언어 교육 시, 논리적인 사고를 텍스트가 아닌 그래픽 방식으로 표현하여 교육적인 효과를 높이는 것은 매우 중요하다. 이 논문에서는 논리적 사고를 표현하기 위해 체계화되고 검증된 소프트웨어 설계도구인 UML을 통해 프로그램에 대한 논리적인 사고를 효율적으로 표현하고, LabVIEW OOP와 같은 비주얼 개발 도구를 이용하여 구현하는 교육을 시행하였다. 프로그래밍을 개발할 때, 요구 분석 단계 및 프로그램 설계 단계에서는 UML과 같은 객체지향 그래픽 설계 도구에서 이해도가 높았으며, 프로그램 구현 단계에서는 LabVIEW OOP로 프로그래밍하는 것이 효율적이라는 결과가 나왔다.

결론적으로, 객체지향 개념을 가지는 도구로 먼저 개념을 도식화하고 LabVIEW OOP와 같은 그래픽 도구로 도식화된 개념을 프로그래밍하는 것이 가장 효과가 높았다.

참고문헌

[1] 주식회사 리얼게인, "그래픽 프로그래밍 언어 개발", 정보통신 산업기술개발사업, p.3, 2003.

[2] 표창우, "프로그래밍 언어 교육 현황 및 개선 방향", 정보과학회지 제16권 제9호, pp.55-57, 1998.

[3] 김상욱, "시각 프로그래밍 기술", 한국통신학회지 제11권 제5호, pp. 33- 45, 1994.

[4] 이성환, 박희선, "시각 프로그래밍과 프로그램 시각화 기술", 정보과학회지 제9권 제5호, pp.41-55, 1991.