

# Alice 프로그래밍을 통한 객체지향 시각 프로그래밍 교육의 유용성 분석

정민포\* · 조혁규\* · 노영욱\*\* · 정덕길\*\*\*

\*영산대학교 · \*\*신라대학교 · \*\*\*동의대학교

## An Analysis of the Object-Oriented Visual Programming Education Using Alice Programming

Min-Po Jung\* · Hyuk-Gyu Cho\* · Young-Uhg Lho\*\* · Deok-Gil Jung\*\*\*

\*Yongsan University · \*\*Silla University · \*\*\*Dong-eui University

E-mail : minpo@ysu.ac.kr

### 요 약

학생들이 프로그래밍 교육에서 학습에 어려움을 겪고 있는 객체지향 프로그램 개발 능력을 향상시킬 수 있는 교육방법으로서 이 논문에서는 시각 프로그래밍 교육 방법론과 교과과정을 도입한다. 3D 프로그래밍 환경을 제공하여 게임 형태의 프로그램을 객체지향 프로그래밍 방식으로 쉽게 작성할 수 있는 Alice 언어를 정규 교과과정에 도입하여 학생들의 프로그래밍 언어에 대한 이해도의 향상과 프로그래밍에 대한 학습 의욕을 고취하여 능력 있는 프로그래머의 양성에 기여할 수 있을 것으로 기대된다. 시각 프로그래밍 교육 방법의 도입에 대한 성과를 분석하기 위하여 프로그래밍 교육의 유용성을 분석할 수 있는 평가 지표를 제시하고 학생들의 프로그래밍 학습 효과를 분석한다.

### 키워드

Alice, JAVA, 프로그래밍 교육, 텍스트 프로그래밍, 시각 프로그래밍, 프로그래밍 교육 평가

## 1. 서 론

최근 우리나라 대학에서 컴퓨터과학은 심각한 문제에 직면해 있다. 학부 및 대학원에서 컴퓨터 과학을 전공하려고 하는 학생들의 수가 급감하고 있으며, 특히, 여학생들뿐만 아니라 남학생들도 컴퓨터 프로그래밍 교과목을 기피하는 분위가 확산되고 있어서, 프로그래밍에 대한 교육이 정상적으로 이루어지지 않으면 컴퓨터과학 분야의 인력 양성에 많은 문제를 야기하게 될 것이다.

이 논문에서는 이러한 상황에서 학생들에게 좀 더 쉽게 컴퓨터 프로그래밍을 학습할 수 있는 교과과정과 방법론을 추구하고자 한다. 한 가지 접근 방법으로는 학생들이 프로그래밍 개념을 쉽게 학습할 수 있는 교육용 프로그래밍 언어를 통하여 프로그래밍 개념과 프로그램 작성 능력을 쉽게 키우고, 이를 통하여 상업용 프로그래밍 학습에서도 그 효과가 미치도록 하고자 하는데 목적이 있다. 또한, 교육 내용에서는 최근 핵심 프로

그래밍 개념이면서도 학생들이 이해와 습득에 어려움을 겪고 있는 객체지향 프로그래밍에 대한 학습을 용이하게 할 수 있는 교육 방법론과 교과과정을 개발해야만 한다.

이 논문에서는 이러한 연구 배경과 필요성에서 현대의 대표적인 객체지향 언어인 Java 언어를 사용한 프로그래밍 능력 향상에 초점을 맞추어서 시각 프로그래밍에 대한 교육 방법론과 교육 과정을 도입하려고 한다.[1,2,3] 3D 프로그래밍 환경을 제공하여 그래픽 프로그래밍을 통한 애니메이션 프로그램을 쉽게 작성할 수 있는 Alice 언어 [4]를 프로그래밍 교육과정에 도입하여 학습 효과를 평가하고 분석한다.

## II. 객체지향 텍스트/시각 프로그래밍 교육 평가 지표

표 1에는 객체지향 텍스트/시각 프로그래밍 교

육의 유용성을 평가할 수 있는 분석 항목들을 제시하고 있다. 유용성 분석 항목들을 프로그래밍의 현실성, 강의 수강 전후 평가, 목표 프로그램 완성을 위한 이해도 및 성취도, 유용성 테스트, 프로그램 작성 과정에서의 난이도로 구분하여 제시하였으며, OOP 분야에서의 프로그램 개발과 관련한 분석 항목을 별도로 시각적 표현에 대한 이해도와 클래스(객체)의 사용에 대한 이해도를 별도로 구분하여 제안하였다.

표 1. 시각/ 텍스트 프로그래밍 교육의 유용성 분석 항목

구분	분석 항목
프로그래밍의 현실성	<ul style="list-style-type: none"> <li>- 프로그래밍 교육을 위한 응용 프로그램의 작성, 개발에서 도움이 되겠는가?</li> <li>- 취업을 위한 응용 프로그램의 작성, 개발에서 도움이 되겠는가?</li> </ul>
강의 수강 전후 평가	<ul style="list-style-type: none"> <li>- 성적 향상 정도</li> <li>- OOP 수준, 이해도 등의 향상 정도</li> <li>- 언어 작성 수준, 이해도 등의 향상 정도</li> <li>- 프로그래밍에 대한 학습 의욕의 고취도</li> </ul>
목표 프로그램 완성을 위한 이해도 및 성취도	<ul style="list-style-type: none"> <li>- 프로그램 표기법에 대한 이해 정도</li> <li>- 프로그래밍 표기법의 이해에 따른 구문 선택 능력</li> <li>- 프로그램 조합에 의한 프로그램 작성 능력</li> <li>- 최종 프로그래밍 학습 성취도 결과</li> </ul>
유용성 테스트	<ul style="list-style-type: none"> <li>- 프로그램 작성이 용이한가?</li> <li>- 프로그램 이해가 용이한가?</li> <li>- 문제 분석이 용이한가?</li> <li>- 프로그램 설계가 용이한가?</li> <li>- 프로그램 변경이 용이한가?</li> </ul>
프로그램 작성 과정에서의 난이도	<ul style="list-style-type: none"> <li>- 문법 배우기의 난이도</li> <li>- 구문 오류 정도</li> <li>- 프로그램 수정 관점에서의 난이도</li> </ul>
OOP 분야에서의 프로그램 개발과 관련한 분석	<ul style="list-style-type: none"> <li>- 시각적 표현에 대한 이해도                             <ol style="list-style-type: none"> <li>1) 클래스/객체 구성 및 관계</li> <li>2) 속성(데이터 멤버)과 멤버 함수(메소드) 사이의 관계</li> </ol> </li> <li>- 클래스(객체)의 사용에 대한 이해도                             <ol style="list-style-type: none"> <li>1) 클래스와 객체에 대한 상관성 및 상호작용에 대한 이해도</li> <li>2) 객체의 속성값에 대한 이해 정도</li> <li>3) 메소드의 호출 및 기능에 대한 이해 정도</li> <li>4) 클래스의 구현과 인터페이스에 대한 이해도</li> </ol> </li> </ul>

### III. Alice/Java 프로그래밍 교육과정

Alice는 스토리 텔링, 인터랙티브 게임 등의 에니메이션 프로그램을 쉽게 작성할 수 있는 프로

그래밍 교육을 위한 혁신적인 3D 프로그래밍 환경이다.[4] 이 논문에서 적용한 객체지향 시각 프로그래밍 언어로는 Alice를 선택했으며 학생들의 학습에 교재로 사용되었다.[5] 이 교재의 전반부는 Alice 프로그래밍에 관한 내용을 포함하고 있으며, 후반부는 JAVA 프로그래밍에 관한 내용으로 구성되어 있다. 전반부의 Alice 프로그래밍 과정에서 사용했던 개념을 Java 프로그램으로 구현하는 방법으로 상호 프로그램을 비교할 수 있게 해주며, Alice 프로그래밍으로 쉽게 객체지향 프로그래밍 개념을 학습하게 하고, 언이어 Java 프로그램으로의 전환을 용이하게 학습하도록 도와준다. 이 교재의 구성에 합치하여 한 학기 강의를 중간고사 이전에는 Alice 프로그래밍으로, 중간고사 이후에는 Java (그래픽) 프로그래밍으로 구성하였다.

Alice를 프로그래밍 언어로 사용한 전반 학기 강의의 주요 교육과정으로는 다음과 같은 항목을 포함한다.

- 1) (Alice) 객체
- 2) (Alice) 메소드, 데이터
- 3) (Alice) 제어문, 반복문
- 4) (Alice) 이벤트
- 5) (Alice) 리스트, 배열

Java 그래픽 프로그래밍을 다루는 후반 학기의 교과과정에는 다음과 같은 항목들을 포함한다

- 1) Alice에서 Java로 전환
- 2) (Java) 이벤트
- 3) (Java) 리스트, 배열
- 4) (Java) 클래스 상속, 계층도
- 5) (Java) 예외처리

### IV. Alice 기반 시각 프로그래밍의 유용성 분석

표 2에는 시각 프로그래밍 언어로는 Alice를 선택하여 반 학기 강의하고, 반 학기는 Alice 프로그래밍에서 다루었던 그래픽 개념을 Java로 프로그래밍하여 학기 종료 후에 Alice 기반의 시각 프로그래밍에 의한 객체지향 프로그래밍의 개념을 쉽게 학습할 수 있는지에 대한 조사 결과를 표시하고 있다. 비교 대상인 텍스트 프로그래밍 언어로는 JAVA 언어를 비교 언어로 설정하도록 지도하였다. 조사에 응한 학생들은 모두 25명이며, 컴퓨터과학과 3학년 학생들을 대상으로 하였다. 평가 척도는 5점 척도로 하였으며, 1(매우 낮음), 2(낮음), 3(보통), 4(높음), 5(매우 높음)으로 설정하였다.

- (1) 프로그램의 현실성

프로그래밍의 현실성에서는 학생들이 실제 업무에서 프로그램 개발시 도움이 되는지의 여부와 졸업 후에 취업을 위한 프로그램 개발에서 도움이 되는지의 분석 항목을 설정하였다. Alice는 교육용 프로그래밍 언어로 주로 사용되며 업무용이나 상용 프로그램 제작의 응용분야에 사용되지 않는다는 인식을 학생들이 가지고 있어서 Java 프로그래밍에 비해 Alice 프로그래밍이 현실성 측면에서는 뒤쳐진다고 평가하는 것으로 분석된다.

표 2. 시각/텍스트 프로그래밍에 대한 학업 성취도

구분	분석 항목	학업 성취도	
		Alice	Java
프로그래밍의 현실성	프로그래밍 교육을 위한 응용 프로그램의 작성, 개발에서 도움이 되겠는가?	3.40	3.76
	취업을 위한 응용 프로그램의 작성, 개발에서 도움이 되겠는가?	3.12	4.00
	<b>평균</b>	<b>3.26</b>	<b>3.88</b>
강의 수강 전후 평가	성적 향상 정도	3.68	3.48
	OOP 수준, 이해도 등의 향상 정도	3.68	3.08
	언어 작성 수준, 이해도 등의 향상 정도	3.56	3.36
	프로그래밍에 대한 학습 의욕의 고취도	4.04	3.28
<b>평균</b>	<b>3.74</b>	<b>3.30</b>	
목표 프로그램 완성을 위한 이해도 및 성취도	프로그램 표기법에 대한 이해 정도	3.64	3.28
	프로그래밍 표기법의 이해에 따른 구문 선택 능력	3.60	3.08
	프로그램 조합에 의한 프로그램 작성 능력	3.68	3.12
	최종 프로그래밍 학습 성취도 결과	3.92	3.32
<b>평균</b>	<b>3.71</b>	<b>3.20</b>	
유용성 테스트	프로그램 작성이 용이한가?	3.76	3.12
	프로그램 이해가 용이한가?	3.84	3.16
	문제 분석이 용이한가?	3.68	3.40
	프로그램 설계가 용이한가?	3.76	3.24
	프로그램 변경이 용이한가?	3.76	3.32
<b>평균</b>	<b>3.76</b>	<b>3.25</b>	
프로그램 작성 과정에서의 난이도	문법 배우기의 난이도	3.16	3.72
	구문 오류 정도	2.88	3.20
	프로그램 수정 관점에서의 난이도	3.00	3.60
	<b>평균</b>	<b>3.01</b>	<b>3.51</b>
OOP 분야에서의 프로그램 개발과 관련한 분석	시각적 표현에 대한 이해도	3.88	3.33
	클래스(객체)의 사용에 대한 이해도	3.92	3.46
	<b>평균</b>	<b>3.90</b>	<b>3.40</b>

(2) 강의수강 전후 평가

강의수강 전후의 학습 성취도와 관련된 평가 분야에서는 Alice 프로그래밍이 Java 프로그래밍보다 성적 향상 및 학습 의욕의 고취에 도움이 된 것으로 나타났다. 특히, OOP 수준, 이해도 등의 향상 정도에 관해서도 Alice 프로그래밍이 Java 프로그래밍보다 약 20% 높게 평가한 것으로 미루어 객체지향 프로그래밍의 이해도 향상에는 Alice 프로그래밍이 많은 도움이 된다는 것을 확인할 수 있다.

(3) 목표 프로그램 완성을 위한 이해도 및 성취도

학생들이 프로그램 작성 시 프로그램 완성을 위해서 습득해야 할 프로그래밍 언어에 대한 이해 및 프로그램 작성 능력에 대한 평가 항목으로서 Alice 프로그래밍을 통하여 프로그램의 표기법, 구문 선택, 조합 등 모든 항목에서 좋은 학습 효과가 나타남은 물론 최종 프로그래밍 학습 성취도에서도 학생들의 학습 성취도 달성에 많은 도움이 된다는 것을 확인할 수 있다.

(4) 유용성 테스트

이 평가 항목은 프로그램 작성/개발 단계별로 학생들의 이해도와 학업 성취도를 평가한 것으로서, Alice 프로그래밍 학습이 각 단계별로 모두 좋은 학업 결과를 보이고 있다. 시각 프로그래밍 학습을 통하여 학생들은 프로그램 개발의 전 단계를 쉽게 이해하고 학습하였다는 것을 확인한 결과로 보인다.

(5) 프로그램 작성 과정에서의 난이도

이 평가 항목은 프로그램 작성 과정에서 학생들이 어느 정도의 어려움을 느끼는지를 평가하는 항목군이다. Alice 프로그래밍에서는 학생들이 언어의 구문을 쉽게 배우며, 그에 따라 구문 오류도 적어지며, 프로그램 수정에 대한 난이도 역시 쉽게 생각함을 확인할 수 있다.

(6) OOP 분야에서의 프로그램 개발 관련 분석

이 평가 항목은 시각적 표현에 대한 이해도와 클래스 사용에 대한 이해도를 측정하는 항목군으로서, 객체지향 프로그램 개발에서 매우 중요한 평가 항목이다. 학생들이 Alice 프로그래밍을 통하여 시각적 프로그래밍에 대한 이해 및 클래스 사용에 대한 이해도의 상승은 객체지향 분석 설계 방법론에서 필수적으로 사용하는 도구인 UML[7]의 학습 및 활용에도 많은 도움이 될 것으로 분석된다.

## V. 결 론

## 참고문헌

전산 관련학과에서 컴퓨터 프로그래밍은 핵심 교과과정을 구성하며 IT 관련 산업의 소프트웨어 개발 인력 양성의 핵심 요소이다. 그러나 컴퓨터 프로그래밍에 대한 학생들의 무관심과 학습의 어려움으로 인하여 학생들의 프로그래밍 능력을 향상시킬 수 있는 프로그래밍에 대한 교육방법론이 필요하다. 이에 따라 이 논문에서는 그 방안으로 교육용 시각 프로그래밍 언어인 Alice를 통한 프로그래밍 교육방법론을 제시하였다.

학생들의 프로그램 작성 능력을 향상시키기 위한 구체적인 방안으로 Alice 프로그래밍을 통하여 프로그래밍의 기본 개념을 습득하고 프로그램 작성 능력을 향상시킨다.[6] Alice 프로그래밍을 통하여 향상된 프로그래밍 작성 능력을 바탕으로 상용 프로그래밍 언어로 프로그램을 개발하는 다음 단계로 자연스럽게 학습 과정을 확대하는 교과과정으로 적용할 수 있을 것이다. Alice를 사용한 시각 프로그래밍을 정규 교과과정에 도입함으로써 학생들의 프로그래밍에 대한 이해도의 고취와 아울러서 기본적인 프로그래밍 개념에 대한 이해도의 향상 뿐만이 아니라 객체지향 프로그래밍에 대한 이해도의 향상 및 프로그래밍 작성 능력의 향상을 확인할 수 있었다.

향후 연구로는 객체 지향 프로그램 제작과정에 필수적인 코딩 이전 단계인 분석/설계 단계에서부터 시각 프로그래밍 과정과 연계한 UML 모델링을 시각 프로그래밍 과정과 Java와 같은 텍스트 프로그래밍 교육과정과도 연계시킬 수 있는 프로그래밍 교육 방법론 및 교육과정의 개발이 필요하다.[7,8]

- [1] DrJava, <http://drjava.sourceforge.net/>
- [2] M. Kolling and J. Rosenberg, "Guidelines for Teaching Object Orientation with Java", ACM SIGCSE Bulletin, 33(3), pp.33-36, 2001.
- [3] L. Nettini, P. Crescenzi, G. Innocenti, and M. Loreti, "An Environment for Self-Assessing Java Programming Skills in Undergraduate First Programming Courses," Proc. of IEEE Int. Conf. on Advanced Learning Technologies(ICALT'04), 2004.
- [4] Alice, <http://www.a.alice.org>
- [5] J. Lewis and P. Depasquale, Programming with ALICE & JAVA, Pearson Education, Inc., 2009.
- [6] K. Powers, S. Ecott, and L. Hirshfield, "Through the Looking Glass: Teaching CS0 with Alice," Conference'04, ACM, 2004.
- [7] UML(Unified Modeling Language), <http://www.uml.org/>
- [8] H. Brand, et al., "The PHELIX Control System Based on UML Design Level Programming in LabVIEW," Proc. of ICALEPCS, pp.472-474, 2003.