

# 객체 지향 텍스트/시각 프로그래밍 교육의 평가 지표 개발

정덕길\* · 정민포\*\* · 조혁규\*\* · 노영욱\*\*\*

\*동의대학교 · \*\*영산대학교 · \*\*\*신라대학교

## A Development of the Evaluation Metrics of the Object-Oriented Text/Visual Programming Education

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

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

E-mail : dgjung@deu.ac.kr

### 요 약

객체지향 텍스트 프로그래밍 및 시각 프로그래밍 교육에 대한 연구는 학생들이 프로그램 개발을 진행하는 과정 혹은 프로그래밍을 교수하는 방법론과 프로그램 개발을 위한 적합한 분석 및 설계 방법론을 찾는 것이다. 이 논문에서는 컴퓨터 프로그래밍 교육에서 객체지향 텍스트/시각 프로그래밍 교육을 적용하고 평가하기 위한 평가 지표를 개발한다. 이 지표는 시각적 혹은 텍스트 기반 프로그래밍 언어/도구를 사용할 때 학생들의 프로그래밍 교육에 대한 적합성을 측정하고 분석하여 학생들의 이해력, 추상화 능력, 프로그램 구현 능력과 학생들의 수준별 적합성을 측정하고 분석하며 또한, 측정/분석 방법론을 개발하고 적용하는데 활용할 수 있다.

### 키워드

객체지향, 텍스트 프로그래밍, 시각 프로그래밍, 프로그래밍 교육 평가지표

### I. 서 론

컴퓨터 프로그래밍 교육은 컴퓨터과학 분야의 관련 전공과목을 공부하기 위하여 매우 중요할 뿐만 아니라, 학생들의 취업을 위한 기술 교육에서도 그 중요성은 아무리 강조해도 지나치지 않는다. 특히, 최근의 기술 추세와 산업계에 필요한 임베디드/모바일/웹/3D 프로그래밍 분야에 널리 사용되고 있는 프로그래밍 분야에서 객체지향 텍스트 기반 언어는 물론이고 시각 프로그래밍 언어/환경에 대한 유용성이 부각되고 있다.

학생들의 컴퓨터 프로그래밍 교육에 대한 수준별 적합성을 측정하고 분석하기 위한 과정으로는 대학 교양 프로그래밍 과정, 전산 관련학과의 프로그래밍 입문 과정, 전산 관련학과의 기초 과정을 이수했지만 프로그래밍 능력이 떨어지는 전산학과의 3,4학년 학생으로 구분하여 적용할 수 있다. 또한, 학생들의 창의력 및 창조성(예: 공학교육 인정 교육과정)을 향상시키기 위한 목적으로도 프로그래밍 과정을 도입할 수 있다.

이 논문에서는 컴퓨터 프로그래밍 교육에서 객

체지향 텍스트/시각 프로그래밍 교육을 적용하고 평가하기 위한 평가 지표를 개발한다. 이 지표는 시각적 혹은 텍스트 기반 프로그래밍 언어/도구를 사용할 때 학생들의 프로그래밍 교육에 대한 적합성을 측정하고 분석하여 학생들의 이해력 추상화 능력, 프로그램 구현 능력과 학생들의 수준별 적합성을 측정하고 분석하며 또한, 측정/분석 방법론을 개발하고 적용하는데 활용할 수 있다.

### II. Alice 및 LabVIEW 프로그래밍 교육

이 논문에서 개발하는 객체지향 텍스트 및 시각 프로그래밍 교육에서 채택하려고 하는 프로그래밍 언어에서 텍스트 기반 언어는 Java 언어[1]를 선택하여 프로그래밍 교육에 활용하는 경우를 상정한다. 시각 프로그래밍 언어로서는 교육용 및 상업용 프로그래밍 언어를 각각 선택하는 것을 상정하는데, 교육용 시각 프로그래밍 언어는 Alice 언어[2]를 선택하고 상업용 시각 프로그래밍 언어는 LabVIEW 프로그래밍 언어를 상정한다.

III. 객체지향 프로그래밍의 평가 지표 관련 연구

객체지향 프로그래밍의 관련 평가 지표에 관한 연구에는 객체지향 프로그래밍 및 개발 방법론과 관련한 평가지표와 객체지향 시각 프로그래밍과 관련된 평가 지표 등으로 구분된다

객체지향 프로그래밍 개발 방법론과 관련한 분석 항목에는 단순화/복잡성, 상호작용성, 시각화, 모델링, 프로그램의 구조에 대한 시각화 코드 템플릿, 그래픽 인터페이스와 같은 항목들이 포함된다.[3] OOP 분야에서의 프로그램 개발/분석/설계/코딩과 관련한 분석에는 클래스 및 객체를 시각적으로 표현하기, 객체의 시각적 표현 및 실체화의 용이성, 클래스(객체)의 사용에 대한 용이성 등이 포함된다.[4] 또한, 객체지향 소프트웨어 평가요인에서 클래스들 사이의 상관관계 멤버 변수 등이 프로그램 설계의 프로파일링 정보를 제공하는지에 대한 평가 지표 설정 및 분석에 관한 내용 등이 있다.[5]

객체 흐름(object-flow) 및 객체지향 시각 프로그래밍에는 데이터 흐름(data-flow)과 객체지향 프로그래밍의 병합 기법에 관한 내용이 있다[6] 또한, 시각 프로그래밍 언어/도구의 객체지향 개발 방법론에 대한 평가 항목 분석에 관한 연구 내용도 있다.[7]

IV. 텍스트/시각 프로그래밍 교육 평가 지표

객체지향 텍스트/시각 프로그래밍 교육에서 평가 지표로 고려되어야 할 요인으로는 객체지향 프로그래밍의 특성에 대한 이해 정도와 프로그램 개발 과정에 대한 이해 정도를 측정할 수 있는 요인들이 포함되어야 한다. 즉, 객체지향 시각 프로그래밍을 사용하여 프로그램을 개발하고 컴퓨터 프로그래밍 교육에서 객체지향 프로그래밍을 도입하는 경우에 대한 장점을 분석하고 평가하는 항목에는 유지보수성(maintainability), 확장성(scalability), 재사용성(reusability)이 포함되어야 한다.

프로그램의 개발 단계별로 언어 표기법에 따른 평가 항목을 고려할 수 있다. 즉, 어떤 표현법이 프로그램 분석, 설계, 구현 단계별로 더욱 효과적인지를 분석, 평가 항목이 포함되어야 한다. 특히, RFID 응용 프로그램과 같은 내장형 프로그램 분야에 적합한 시각적 분석/설계/구현 언어(도구) 혹은 다이어그램 표현법이 프로그램의 개발 및 학생들의 학습에 효과적인지를 분석 평가할 수 있는 항목들도 고려되어야 한다.

표 1. 시각/ 텍스트 프로그래밍 교육의 유용성 분석 항목

구분	분석 항목
프로그래밍의 현실성	- 프로그래밍 교육을 위한 응용 프로그램의 작성, 개발에서 도움이 되겠는가? - 취업을 위한 응용 프로그램의 작성, 개발에서 도움이 되겠는가?
강의 수강 전후 평가	- 성적 향상 정도 - OOP 수준, 이해도 등의 향상 정도 - 언어 작성 수준, 이해도 등의 향상 정도 - 프로그래밍에 대한 학습 의욕의 고취도
목표 프로그램 완성을 위한 이해도 및 성취도	- 프로그램 표기법에 대한 이해 정도 - 프로그래밍 표기법의 이해에 따른 구문 선택 능력 - 프로그램 조합에 의한 프로그램 작성 능력 - 최종 프로그래밍 학습 성취도 결과
유용성 테스트	- 프로그램 작성이 용이한가? - 프로그램 이해가 용이한가? - 문제 분석이 용이한가? - 프로그램 설계가 용이한가? - 프로그램 변경이 용이한가?
프로그램 작성 과정에서의 난이도	- 문법 배우기의 난이도 - 구문 오류 정도 - 프로그램 수정 관점에서의 난이도
OOP 분야에서의 프로그램 개발과 관련한 분석	- 시각적 표현에 대한 이해도 1) 클래스/객체 구성 및 관계 2) 속성(데이터 멤버)과 멤버 함수(메소드) 사이의 관계 - 클래스(객체)의 사용에 대한 이해도 1) 클래스와 객체에 대한 상관성 및 상호작용에 대한 이해도 2) 객체의 속성값에 대한 이해 정도 3) 메소드의 호출 및 기능에 대한 이해 정도 4) 클래스의 구현과 인터페이스에 대한 이해도

표 1에는 객체지향 텍스트/시각 프로그래밍 교육의 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.

학생들에 대한 프로그래밍 교육의 평가에는 다음과 같은 요소들이 포함된다.

- 1) 강의/시험 전/후 자료 수집 및 평가
- 2) 학점(성적) 향상 정도

- 3) OOP 언어(텍스트 및 시각 프로그래밍) 수준
- 4) 강의 수강 후에 OOP 언어(텍스트 및 시각 프로그래밍) 수준, 이해도 등의 향상 정도
- 5) 컴퓨터 프로그래밍에 대한 학습 의욕의 고취도(향상) 정도
- 6) 소프트웨어(프로그램) 분석/설계 방법론 능력(수준) 및 수강 전후의 향상도

또한, 객체지향 텍스트/시각 프로그래밍 언어를 사용한 프로그래밍 교육에서의 가치 있는 중요 특성 및 이유는 다음과 같다.

- 1) OOP 분야에서의 프로그램 분석/설계/코딩
- 2) 클래스 및 객체를 시각적으로 표현하기
- 3) 객체의 시각적 표현 및 실체화의 용이성
- 4) 시각 프로그래밍 언어/도구/환경에 대한 사용의 용이성
- 5) 클래스(객체)의 사용에 대한 용이성

## V. 결 론

이 논문에서는 학생들의 교육 및 취업에 적합한 교육용 및 상용 프로그래밍 언어 도구에 대한 타당성을 분석할 수 있는 유용성에 관한 평가 지표를 개발하여 프로그래밍 교육에 적합한 프로그래밍 언어/도구를 선택하고, 이에 대한 교육 과정을 개발하여 프로그래밍 현장 교육에 적용하고 활용하는데 목적이 있다. 구체적으로, 프로그래밍 언어 선택 및 학습 효과를 위한 평가 분석 항목에 대한 비교 및 평가 항목 설정을 위한 연구이며, 객체지향 텍스트 프로그래밍(JAVA)을 사용한 프로그래밍 교육과 시각 프로그래밍(LabVIEW)을 사용한 프로그래밍 교육을 비교 분석하는데 활용하려고 한다.

객체지향 시각 프로그래밍의 응용 분야에는 내장형 프로그래밍, 모바일 프로그래밍, 로봇틱스 프로그래밍(예: 레고 로봇) 등에 응용될 수 있는 프로그래밍 언어를 대상으로 하여 학생들에 대한 프로그래밍 교육의 적합성 및 수월성을 평가하는데 활용될 수 있다.

시각 프로그래밍 언어(예: LabVIEW)의 유용성 분석에서는 시각 프로그래밍 언어를 사용한 프로그래밍 교육과 텍스트 기반 언어(JAVA)를 사용한 프로그래밍 교육을 비교, 분석하여, 학생들이 프로그램 개발을 진행하는 과정 혹은 프로그래밍을 교수하는 방법론과 프로그램 개발을 위한 적합한 분석 및 설계 방법론을 찾는 것이 향후 연구의 방향이다.

## 참고문헌

- [1] 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.
- [2] K. Powers, S. Ecott, and L. Hirshfield, "Through the Looking Glass: Teaching CS0 with Alice," Conference'04, ACM, 2004.
- [3] L. McIver, "Evaluating Languages and Environments for Novice Programmers," 14th Annual Workshop of the Psychology of Programming Interest Group(PPIG 2002), pp.100-110, 2002.
- [4] M. Kolling and J. Rosenberg, "Guidelines for Teaching Object Orientation with Java," ACM SIGCSE Bulletin, 33(3), pp.33-36, 2001.
- [5] S. Puro and V. Vaishnavi, "Product Metrics for Object-Oriented Systems," ACM Computing Surveys, Vol.35, No.2, 2003, pp.191-221
- [6] T. Maila, "Object-flow Programming - Merging Dataflow and Object-Oriented Programming," <http://expressionflow.com/2007/04/27/object-flow-programming-merging-dataflow-and-object-oriented-programming/>
- [7] S. Georgantaki and S. Retalis, "Using Educational Tools for Teaching Object-Oriented Design and Programming," Journal of Information Technology Impact, Vol.7, No.2, pp.111-130, 2007.