

엔트리 프로그래밍 교육이 초등예비교원의 몰입에 미치는 영향

한규정

공주교육대학교 컴퓨터교육과

요 약

본 논문은 몰입기반 학습 요소를 갖는 엔트리 프로그래밍 교육과정이 학생들의 학습 몰입도에 영향을 주는지를 검증하는 데 그 목적이 있다. 연구 대상은 교육대학교 3학년 학생으로 그전에 코딩을 경험하지 않았던 학생들이며 총 3개월동안 주당 2시간의 프로그래밍 학습을 시행하였다. 학습의 교재로는 엔트리 홈페이지에서 제공하는 학습 콘텐츠와 초급과정의 엔트리 학습 교재를 활용하였다. 교수 학습과정은 명확한 목표제시, 동기유발, 따라 하기 및 빠른 피드백, 응용문제 제공, 학생들의 반추 시간 제공 등으로 구성하였다. 연구의 결과로서 총9가지의 몰입요소를 실험전과 실험 후를 비교한 결과 7가지 요소인 도전-기술 균형, 구체적 피드백, 행동-지각 일치, 과제 집중, 통제감, 자의식 상실, 자기 목적적 경험 등이 유의하다는 결과를 얻었다.

키워드 : 소프트웨어교육, 코딩교육, 몰입, 엔트리, 엔트리 프로그래밍

The effect of the entry programming course on the flow of elementary pre-service teacher

Kyujung Han

Dept. of Computer Education, Gongju University of Education

ABSTRACT

The purpose of this paper is to verify whether the entry programming course applied with the flow based learning type is to affect the flow of students' learning. The subjects of the study were the students in the 3rd grade of the National University of education who had not experienced the coding before, and conducted two hours of programming lessons per week for three months. Learning contents and learning materials of beginner course in Entry Homepage were used as learning materials. The teaching and learning process consisted of clear goals setting, motivation, follow-up and immediate feedback, providing application problems, and providing reflection time for students. As a result of the study, the nine factors of flow were compared before and after the experiment. The seven factors - A balance between challenges and skills, Immediate feedback, Action and awareness are merged, Distractions are excluded from consciousness, No worry of failure. Self-consciousness disappears, The activity becomes autotelic- were obtained.

Keywords : Software education, Coding Education, Flow, Entry, Entry programming

1. 서론

최근 매우 빠른 과학 기술의 발달과 혁신으로 새로운 시대가 열리고 있는데 이를 “제4차 산업 혁명”이라고 한다. 클라우스 슈밥[11]에 의하면 제4차 산업혁명의 특징 다음과 같다. 첫째, 기존의 1,2,3차 혁명이 선형적인 속도로 전개된다면 제4차 혁명은 기하급수적으로 전개되며 우리가 살고 있는 세계가 다면적으로 서로 깊게 연계되어 새롭고 더 발전된 기술을 만들어내고 있다. 둘째, 제4차 혁명은 다양한 범위와 깊이로 영향을 주고 있다. 디지털 혁명을 바탕으로 여러 과학 및 정보기술을 융합하여, 개인뿐만 아니라 기업, 사회, 경제전반에 새로운 패러다임을 보여주고 있다. 셋째, 기업 간, 산업간 그리고 사회전체 시스템에 충격과 변화를 주고 있다.

이러한 급격한 패러다임 전환이 일어나고 있는 제4차 산업혁명의 국제화 사회에서 개인, 기업 혹은 국가가 생존하기 위한 기술로는 컴퓨터 기술, 정보기술, 미디어 및 정보리터러시 기술 등을 요구하고 있다[17]. 오닐라스[18]등은 컴퓨터과학교육 분야에서 많은 학생들이 문제 해결 능력과 컴퓨터적 사고 기술이 부족하다고 역설하였다. 이런 관점에서 학생들 대상의 ‘프로그래밍 학습’은 컴퓨터적 사고와 문제 해결 능력을 길러주는 시발점이 된다.

최근 우리나라의 초중등 교육에서도 소프트웨어교육을 강화하려는 시도가 나타나고 있다. 교육부가 2015년 소프트웨어교육과정 고시를 통해 2018년부터는 초등학교에서도 소프트웨어 교육을 필수적으로 실시할 예정이다. 그 내용으로는 실과교과에서 17시간이상의 소프트웨어 교육이 포함되어 있고 관련된 내용체계와 성취기준 및 해설 등이 발표되었다[8].

이러한 시점에 초등학생들에게 소프트웨어교육을 할 예비교원들의 프로그래밍 능력에 대한 역량강화의 노력이 매우 필요하다. 현재 교육대학교의 교육과정에서는 컴퓨터적 내용학의 영역으로 컴퓨터적 사고(Computational Thinking), 언플러그드 교육(Unplugged Education), 프로그래밍언어, 교육용프로그래밍 언어(Educational Programming Language, EPL)과 교과교육학 영역으로 소프트웨어교육, 컴퓨터교육 등을 실시하고 있다.

컴퓨터공학을 전공하지 않은 예비교원에게 능숙한 프

로그래밍 능력을 기대하기는 상당히 어렵다. 그 이유는 다음과 같다. 첫째, C 언어, 파이선 등과 같은 텍스트 기반 언어의 교육은 예비교원들에게는 언어 자체가 갖는 난이도 등으로 인해 인지적 부하를 많이 주고 있다. 둘째, 교육대학교의 교육과정상 부족한 시수로 인해 충분한 프로그래밍 교육이 잘 이루어지 못하고 있다. 컴퓨터 교육과가 아닌 대다수의 학생들의 컴퓨터교육에 관련된 시수는 매우 부족한 관계로 프로그래밍 관련교과에서 컴퓨터교육의 이론 부분과 같이 프로그래밍 언어 교육을 하는 경우가 많다. 예컨대, C 교육대학의 ‘초등컴퓨터교육’의 과목에서는 15차시에서 불과 6차시내지 8차시 정도의 프로그래밍 교육을 실시할 정도이다. 셋째, 학생들에게 실시하는 따라 하기식의 프로그래밍 교육 방법은 흥미를 반감시키며 학습효과를 기대하기가 어렵다. 이는 일반적인 프로그래밍 언어의 교육방법에 문제점과도 관련이 있다.

본 연구에서는 주어진 짧은 프로그래밍 교육시간에 최대의 교육 효과를 내기 위해 블록기반프로그래밍 언어 선택과 학생들의 프로그래밍 학습의 몰입을 유도하는 방법을 적용하였다. 텍스트기반이 아닌 블록 기반의 엔트리(Entry) 라는 교육용 프로그래밍 언어를 도구로 학생들의 코딩교육에 대한 접근성을 용이하도록 하였다. 또한 코딩을 즐거움을 맛보고, 학생들의 사고력을 증진시키기 위한 코딩 학습에 몰입할 수 있는 교육 내용을 구성하였다.

본 연구에서는 프로그래밍 교육에 몰입을 적용하였다. 칙센트 미하이[3]는 “몰입을 쉽게 하기 위해서는 명확한 목표를 가지고, 결과의 피드백이 빨라야하고 과제의 난이도가 적절해야 한다”고 하였다. 이를 기반으로 본 연구에서의 프로그래밍 교육 내용은 첫째, 각 차시의 목표를 명확하게 하고, 둘째, 학생들이 수행하는 과제의 결과에 대한 피드백을 빠르게 주고, 셋째, 수업할 내용의 난이도를 수준별로 조절하였다.

본 연구에서는 기존의 연구에서처럼 단순히 블록언어 프로그래밍 언어를 교육하는 것이 학습자의 몰입을 극대화 하지는 못할 것이라는 개념에서 출발하였다. 따라서 본 연구에서는 프로그래밍 교육에 엔트리 언어의 선택, 흥미유발 학습콘텐츠의 적용, 그리고 몰입기반 학습요소의 교수학습 상황에 적용 등이 학습자의 몰입도에 영향을 주는지를 검증하였다.

다만 연구의 제한요소로는 적용된 몰입교수학습과정은 칩센트 미하이의 몰입을 변형하였고, 또한 짧은 수업시간의 몰입주도 학습요소 등이 학생들의 몰입도에 어떻게 영향을 주는가에 대해서는 심도 있는 연구가 필요하다.

2. 이론적 배경 및 선행연구

2.1 프로그래밍의 난점

프로그래밍을 배우는 것은 일반적으로 어려운 일이다. 솔로웨이 [21] 등은 초보자가 능숙하게 프로그래머가 되려면 10년의 기간이 걸린다고 하였다. 콜링[12]등은 일반 초보자들은 프로그래밍시 프로그램의 큰 구조 측면에서 접근하기보다는 “line by line” 방식으로 접근함으로써 코드의 설계, 테스트 등에는 시간을 적은 시간만 투자하고 오로지 간단한 예러 수정만으로 프로그램을 작성하려는 경향을 보인다고 하였다.

학생들의 프로그래밍에 관련된 개성도 영향을 준다. 퍼킨스[19]는 학생들이 문제 상황에 접했을 때 대처하는 2가지 유형을 소개하였는데 “멈추는 학생(stopper)”과 “움직이는 학생(mover)”이다. 문제 상황에서 “멈추는 학생”은 문제를 해결하기 않고 단순히 포기하는 반면에 “움직이는 학생”은 끊임없이 예러를 효과적으로 해결하고, 코드를 수정하려한다고 하였다.

고메스[6]등은 프로그램을 배우는 것이 쉽지 않은 이유로써 첫째, 강의가 개인의 특성에 맞게 실행되지 않고, 둘째, 강의가 학생들의 다양한 학습양식을 고려하지 않고 있으며 셋째, 프로그램 자체는 동적으로 변화무쌍한 특성을 가지고 있는 반면에 강의는 정적인 학습 자료인 프로젝터의 활용, 말로만 설명, 텍스트 위주 등으로 이루어지고 있으며, 넷째, 교사는 프로그래밍언어의 지도 즉 문법 등에 치중하고 프로그래밍 언어를 활용한 문제 해결 등에는 소홀하고 있다고 하였다.

초보자가 프로그래밍에서 느끼는 어려움은 첫째, 변수의 초기화이고 둘째, 반복문과 조건문에서의 예러가 빈번하며 “for” 반복문에서 수정되는 반복 변수에 대한 예러, 셋째, 재귀(recursion)에 대한 이해 부족, 넷째, C 언어에서의 포인터의 개념의 어려움, 다섯째, 구문적으로는 비슷한데 내용이 다른 것에서 오는 어려움으로 문

자 “123”, 숫자 123 의 차이 등이며 여섯째, 굉장히 복잡한 언어 특성으로 “템플릿(template)”, “다형화(polymorphism)”, “함수오버로딩(function overloading)” 등이다[9][16].

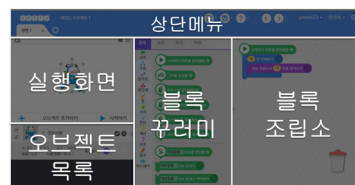
이와 같이 프로그래밍 언어의 복잡성, 텍스트 기반 등은 초보자에게 문제 해결의 작업 중점보다는 언어 그 자체에 관심을 요구함으로써 쉽게 흥미를 잃고 프로그래밍 과정을 완수하지 못하는 요인이 되고 있다. 오늘날 스[18]는 학생들이 스크래치 언어를 활용하면 컴퓨터적 사고 기술이나 문제해결 기술을 개발할 수 있고 즉각적인 문제 해결 방법을 찾을 수 있다고 하였다.

스크래치의 효과에 관한 연구는 다양하다. 램[15]은 스크래치의 코드의 재사용에 주목하였는데 이 특성은 코드를 쉽게 수정하고, 합치고, 공유할 수 있다는 것이다. 따라서 이런 재사용성은 학생들이 스토리, 게임, 미술, 음악, 애니메이션 등 그 이상을 가능하게 해준다. 레이[14]는 스크래치와 같은 시각화 프로그래밍 언어의 문제 해결 작업의 효과를 알아보기 위해 초등학교 6학년 학생들에게 적용한 결과, 학생들의 문제해결능력에 유의한 결과를 준 것을 알 수 있었고 특히 예측논리 측면에 영향을 주었다. 월슨[24]은 초등학교 초보자를 위한 8주간 IT 교육과정으로 스크래치의 사용을 평가하였는데 인지적 측면(스크래치가 개념을 잘 지도하는지), 정적적 측면(사용하는 데 즐거우는지) 등에서 장점이 있음을 발견하였다.

2.2 엔트리 프로그램과 몰입

2.2.1 엔트리 언어의 특성

엔트리 프로그램도 스크래치와 유사한 블록기반 비주얼 프로그래밍 언어로 엔트리교육연구소에서 개발한 소프트웨어교육 플랫폼이다. (Fig. 1) 과 같이 쉬운 인터페이스로 구성되어 있다.



(Fig. 1) The Menu of Entry[4]

엔트리도 학생들이 스크래치와 유사하게 스토리, 게임, 애니메이션 등을 다양하게 제작할 수 있으며 제작된 프로그램의 공유 및 재사용 등을 통해 수정, 재창조, 확장 등을 쉽게 할 수 있다. 따라서 학생들은 기존의 프로그래밍 언어 학습에서의 문제점인 지나친 문법 학습이나 에러 수정에 시간을 소요하는 것보다 주어진 과제의 문제해결과 컴퓨터적 사고를 직관적으로 할 수 있다. 스크래치에 대비한 강점은 다양한 한글로 된 교육자료 및 엔트리 프로그램의 공유가 가능하며 또한 스크래치보다 훨씬 많은 오브젝트를 자체 제공함으로써 사용자가 쉽게 필요한 그림이나 캐릭터를 활용할 수 있다.

본 연구에서는 학생들이 주로 문제해결에만 관심을 두고 프로그래밍 기술을 쉽게 배울 수 있는 언어로써 엔트리 프로그램을 선택하였고, 몰입기반 교육학습 프로그래밍 학습을 적용하여 학생들의 몰입정도를 연구하였다.

2.2.2 몰입

칙센트미하이[3]는 “몰입이란 사람들이 다른 어떤 일에는 관심이 없을 정도로 지금 하고 있는 일에 푹 빠져 있는 상태”라고 하였다. 그는 사람들이 자신이 하는 일에 전적으로 집중할 때 더 행복감을 느낀다는 것을 발견했다. 몰입을 플로우(Flow)로 표현하였고 어떤 일에 몰입했을 때 그 활동이 마치 ‘물 흐르듯이 자연스럽게 되는 것’에서 오는 행복을 의미한다.

칙센트미하이[2]는 몰입 경험에서 일어나는 공통적인 특성으로 다음과 같은 9가지 요소를 들었다.

가) 도전과 능력의 조화

개인이 상황에 대한 도전을 이루기 위해서는 개인의 능력 혹은 기술이 도전의 수준에 균형이 될 때 몰입 경험이 이루어진다.

나) 행위와 의식의 통합

몰입 상태에서는 자신들의 행위와 지각이 일치하여 자발적으로 거의 자동적으로 진행된다. 몰입의 수행은 거의 물 흐르듯 저절로 수행이 된다.

다) 명확한 목표

사전에 분명한 목표를 설정함으로써 정확히 무엇을

해야 할지 알 때 몰입 경험이 가능하다. 명확한 목표는 자신이 무엇을 하고 있는지를 정확한 파악 가능하며, 자신이 의도하지 않은 상황이 발생 했을 때는 행동을 수정하여 자신이 세운 처음 목표에 달성할 수 있도록 도와준다.

라) 구체적인 피드백

구체적인 피드백은 자신의 수행 중의 활동에 대한 정확하고 신속한 피드백이다. 수행 활동에 대해 스스로가 제대로 수행하고 있는지 그리고 그 활동이 잘 진행되고 있는지에 대해 매번 깨닫고 점검하는 것이다.

마) 과제에 대한 집중

다른 사람 혹은 다른 과제에는 전혀 관심을 가지지 않고 완벽하게 자신의 현재 과제에만 집중한다.

바) 통제감

몰입에 빠지면 자신에 대한 고도의 통제감을 갖게 된다. 몰입은 외부적 상황에서의 압력이나 강제가 아닌 자발적인 상황이므로 개인은 자신이 처한 상황에 대해 통제감을 가진다.

사) 자의식의 상실

사람이 몰입 상태에 빠지면 자기 자신의 존재를 인식하지 못할 만큼 활동과 자신이 하나가 된다.

아) 시간감각의 왜곡

몰입 상태에 도달하면 사람들이 평소 지각하는 시간의 흐름과 달리 매우 빠르게 흐르거나 느리게 흐르는 것으로 시간 감각이 왜곡된다.

자) 자기목적적 경험

어떤 외부적인 보상이나 목표도 필요로 하지 않고 수행 자체 즐거움과 만족감을 느끼게 된다. 즉 내재적으로 보상받는 경험이다.

2.3 기존의 프로그래밍에서의 몰입 연구

유인환[7]은 로봇을 활용한 프로그래밍 교육이 초급 프로그래밍 학습자들의 몰입에 매우 긍정적으로 기여하

고 있는 것을 발견하였다. 특히 몰입의 하위 요소인 선행, 경험, 효과요소가 모두 유의하게 높게 나타났다. 안경미[10]등은 초등학생을 대상으로 스크래치 프로그래밍 교육을 실시하고 학습 몰입과 프로그래밍 능력에 미치는 효과에 대해서 분석하였다. 박영숙[23]등은 블렌디드 로봇교육 활동이 아동의 학습동기와 학습 몰입에 미치는 효과의 논문에서 온라인과 오프라인을 병행한 블렌디드 방식의 로봇교육활동이 아동의 학습동기와 학습 몰입에 긍정적인 효과를 준다고 하였다. 신승용[20]은 로봇을 활용한 STEAM 수업에 ‘도전과 기술의 조화’와 같은 몰입 요인이 학습자의 학습 지속 태도에 영향을 주는 과정을 분석하였다. 황성진[22]은 초등정보영재의 학습 몰입 향상을 위해 직관적으로 스마트 기기 용 앱을 개발할 수 있는 앱 인벤터를 이용하여 컴퓨터적 사고 기반의 초등정보영재 교육 프로그램을 개발하였다. 김경현[13]은 초등학교 로봇활용 수업은 특히 국어, 과학, 미술 등의 교과 수업에서 학생들의 학습 몰입도를 긍정적으로 변화시키는 영향이 크다는 것을 발견하였다. 이은경[5]은 중학생을 대상으로 스크래치 프로그래밍 교육이 학생들의 몰입수준과 프로그래밍 능력에 미치는 영향을 분석하였다. 스크래치를 활용한 실험집단이 비주열 베이직을 활용한 통제 집단보다 몰입수준이 유의하게 높게 나타났다.

본 연구는 기존연구와의 유사점은 스크래치와 유사한 엔트리 언어를 사용하였다는 점이다. 차이점은 첫째, 엔트리의 흥미유발형 학습콘텐츠를 활용하였고, 둘째, 교수 학습 과정에 칙센트미하이의 몰입요소를 투입하여 프로그래밍에 몰입을 하도록 유도하였다.

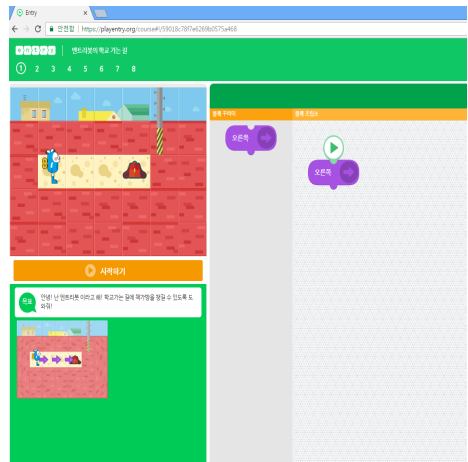
2.4 몰입 지향 엔트리 프로그래밍 수업

전혀 프로그래밍의 개념이 없는 수업대상 학생을 위해서 수업의 단계는 2가지로 구분하여 진행하였다. 그 하나는 준비단계로서 대략 1주동안 엔트리와 익숙해지고 흥미 유발을 위해 엔트리의 게임형 콘텐츠를 활용하였고, 다른 하나는 엔트리 교육용 교재를 활용하였다.

2.4.1 흥미유발 학습형 엔트리 콘텐츠 활용

엔트리의 홈페이지의 ‘학습하기’중 초등학교 학년별

추천과정에서 초등학교 3~4학년, 초등학교 5~6학년 단계의 총 콘텐츠 6개를 학생들이 직접 작동하여 엔트리의 명령어와 개념을 자연스럽게 익히도록 하였다. 또한 중등이상의 콘텐츠 3개를 과제로 제출하여 강좌 홈페이지에 각 콘텐츠를 수행한 총 완성 시간을 올리도록 하였다. (Fig.2)는 학습형 엔트리 콘텐츠이다.



(Fig. 2) Example of Entry Contents

2.4.2 엔트리 교재의 활용

몰입 지향 수업을 위한 엔트리 수업교재는 엔트리 회사에서 제공하는 ‘차근차근 따라하며 배우는 엔트리 프로그래밍(초급)’[4]을 사용하였고 학생들에게는 인쇄물을 나누어주지 않고 각 PC에 교재를 E-Book 형태로 설치하여 주었다. 그 주제는 다음과 같다. 수업은 한 주의 수업시간 2시간에서 대략 2단원 주제를 실습위주로 진행하였다. <Table 1>은 학습 주제이다.

<Table 1> The Contents of Learning Material

Unit	Activities
1	Subscribing to Entry
2	Exploring the interface of Entry
3	A rocker who can't play the guitar
4	Playing a good guitar
5	Walking Alex
6	A faithful Alex
7	Eagle flying back and forth
8	I am a Chameleon of love

9	Stoker EntryBot
10	Driving a Car
11	Let's fly, space far!
12	Tone and Jelly
13	Sharing Entry programs
14	Creating one's own music player
15	Referring and developing other's works
16	Short answer about the national capital of country
17	If you enjoy bouncing balls in the moon
18	Let's express the birth the universe
19	Wolverine went to the weapons shop
20	A penguin walking on the water
21	Creating a simple piano keyboard
22	Creating Robots for addition operation
23	Creating Robots for arithmetic operation
24	EntryBot memorize times table
25	Drawing program for regular polygon
26	Changing playback speed

2.4.3 프로그래밍 학습 지도안 예시

몰입 지향 학습을 위의 구조는 명확한 학습 목표, 동기 유발, 따라하기 및 빠른 피드백, 응용문제(에러 고치기 등 포함), 성찰학습 등으로 구성하였으며 그 주요 내용은 <Table 2>와 같다.

<Table 2>는 지도안의 예시로서 이 차시에서는 국가의 수도를 알아맞히는 프로그램을 작성하는 데 그 목표가 있다. 명확한 학습 목표를 위해 학생들이 작성하려고 하는 프로그램의 결과물을 보여주고, 핵심 프로그래밍 요소로 반복문과, 문고 기다리기 블록의 특성을 발견하게

하였다. 몇 가지 블록코딩을 따라 하기를 하며 학생들의 에러나 어려움이 있으면 그 원인에 대한 설명 등을 즉각적인 피드백을 주었고, 시간이 부족한 관계로 동료학습의 과정으로 피드백을 코딩을 잘하는 학생이 도와주도록 하였다. 학습목표에 도달한 학생들에게는 몇 가지 국가 수도 맞추기 응용문제를 풀도록 하였고, ‘반복문 벗어나기’라는 블록이 없는 에러를 심은 프로그램을 주어졌을 때 문제를 발견하고 에러를 수정할 수 있는지를 도전하도록 하였다. 학습의 마무리 단계에서는 학생들에게 오늘 배운 학습에 대해 3분동안 눈을 감고 반추하고 백지에 적어보도록 하여 스스로 학습을 정리하도록 하였다.

3. 연구방법

3.1 연구대상

본 연구는 2016년 8월부터 11월까지 대략 3개월동안 수행하였으며, 연구 대상은 엔트리기반 프로그래밍 수업을 수강한 국내 모 교육대학교 3학년 학생 29명이다. 이 학생들은 이전에는 프로그래밍학습의 경험이 없다.

3.2 실험설계

연구의 실험 설계는 엔트리 기반 프로그래밍 수업을

<Table 2> The Example of Lesson Plan

Step	Teacher	Student	The main factors of Learning
Introduction	1. Describe the content of the 'Short answer about the national capital of country' program to learn and show the resulting of program execution 2. Presenting Learning Objectives: Can do programming using 'Text boxes', '~Asking and Waiting block', 'Stop Repeating block'	Listen and run the program	Motivation (check the result)
Development	1. Using 'Text Boxes' 2. Using '~Asking and Waiting block', 2. Applying 'Stop Repeating block' 3. Extending and applying 'Short answer about the national capital of country' program to three countries	-Following -Find the wrong parts in program	Interaction Application problems
Wrap up	1. To organize what student has learned 2, To do advanced application program make scores appear in quizzes.	Reflect on what student has learned	Reflection Challenge problem

수강한 집단을 대상으로 단일집단 전 후 검사 설계를 채택하였다. 사전검사와 사후 검사에 학습 몰입도에 관련된 검사를 통해 엔트리기반 프로그래밍의 수업이 9가지 학습 몰입도에 영향을 주는지를 검증하였다. <Table 3>은 실험설계 과정이다.

<Table 3> Experimental design

O1	X	O2
O1 : pre test (Test Learning Flow)		
X : Class of Entry Programming		
O2 : post test (Test Learning Flow)		

3.3 학습 몰입도 검사지

본 연구에 적용된 학습 몰입도 검사지는 김아영[1]의 ‘성인용 학습 몰입 척도 개발 및 타당화’ 연구의 학습 몰입요인 9개요인을 측정하기 위한 29문항을 적용하였다. 9개 요인은 도전-기술 균형(3개 문항), 명확한 목표(3개 문항), 구체적 피드백(3개 문항), 행동-지각 일치(3개 문항), 과제집중(3개 문항), 통제감(3개 문항), 자의식 상실(3개 문항), 변형된 시간 감각(3개 문항), 자기 목적적 경험(5개 문항) 으로 구성되어 있다. 9개의 학습몰입도 요인과 그 하위 문항 그리고 신뢰도는 <Table 4>와 같다.

<Table 4 Item Internal Reliability of Test Flow

Domain	Item Number	Numbers of Item	Cronbach's α	
			pre	post
①	1,2,3	3	.883	.920
②	4,5,6	3	.949	.808
③	7,8,9	3	.919	.938
④	10, 11, 12	3	.956	.965
⑤	13, 14, 15	3	.882	.966
⑥	16, 17, 18	3	.907	.959
⑦	19, 20, 21	3	.942	.953
⑧	22, 23, 24	3	.902	.916
⑨	25, 26, 27, 28, 29	5	.961	.940
Total		29	.989	

- ① A balance between challenges and skills
- ② Clear goals
- ③ Immediate feedback

- ④ Action and awareness are merged.
- ⑤ Distractions are excluded from consciousness
- ⑥ No worry of failure.
- ⑦ Self-consciousness disappears.
- ⑧ The sense of time becomes distorted
- ⑨ The activity becomes autotelic.

4. 연구결과

4.1 학습 몰입도 검증 결과

컴퓨터 교육용 프로그램이 학생의 몰입도에 미치는 효과를 알아보기 위해 몰입에 관련된 9가지 요소에 대해서 대응표본 t-검증을 실시하였다.

4.1.1 도전-기술 균형

<Table 5>는 도전-기술 균형에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 4.02에서 4.16로 증가하였고 통계적으로 유의미한 차이를 가졌다($p < .05$)

<Table 5 A balance between challenges and skills(t-test)

	M	N	SD	t	df	p
Pre	4.02	29	.67	3.266	28	.003
Post	4.16	29	.62			

$p < .05$

4.1.2 명확한 목표

<Table 6>은 명확한 학습목표에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값은 변화가 없었고 통계적으로 유의미한 차이가 없었다($p > .05$)

<Table 6> Clear goals(t-test)

	M	N	SD	t	df	p
Pre	3.74	29	.81	.000	28	1.000
Post	3.74	29	.63			

$p < .05$

4.1.3 구체적 피드백

<Table 7>은 구체적 피드백에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 3.29에서 3.78으로 증가하였고 통계적으로 유의미한 차이를 가졌다($p < .05$).

<Table 7> Immediate feedback(t-test)

	M	N	SD	t	df	p
Pre	3.29	29	.93	8.764	28	.000
Post	3.78	29	.92			

$p < .05$

4.1.4 행동-지각 일치

<Table 8>은 행동-지각 일치에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 2.89에서 3.26으로 증가하였고 통계적으로 유의미한 차이를 가졌다($p < .05$).

<Table 8> Action and awareness are merged(t-test)

	M	N	SD	t	df	p
Pre	2.89	29	.99	5.299	28	.000
Post	3.26	29	1.05			

$p < .05$

4.1.5 과제 집중

<Table 9>는 과제 집중에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 2.64에서 3.32로 증가하였고 통계적으로 유의미한 차이를 가졌다($p < .05$).

<Table 9> Distractions are excluded from consciousness(t-test)

	M	N	SD	t	df	p
Pre	2.64	29	.98	8.646	28	.000
Post	3.32	29	1.09			

$p < .05$

4.1.6 통제감

<Table 10>은 통제감에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 3.10에서 3.49로 증가하였

고 통계적으로 유의미한 차이를 가졌다($p < .05$)

<Table 10> No worry of failure(t-test)

	M	N	SD	t	df	p
Pre	3.10	29	.90	6.535	28	.000
Post	3.49	29	.97			

$p < .05$

4.1.7 자의식 상실

<Table 11>은 자의식 상실에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 2.84에서 2.99로 증가하였으나 통계적으로 유의미한 차이를 가졌다($p < .05$).

<Table 11> Self-consciousness disappears(t-test)

	M	N	SD	t	df	p
Pre	2.84	29	1.02	2.095	28	.045
Post	2.99	29	.97			

$p < .05$

4.1.8 변형된 시간 감각

<Table 12>는 변형된 시간감각에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 3.34에서 3.57로 증가하였으나 통계적으로 유의미한 차이가 없었다 ($p > .05$).

<Table 12> The sense of time becomes distorted(t-test)

	M	N	SD	t	df	p
Pre	3.34	29	.99	1.889	28	.069
Post	3.57	29	.93			

$p < .05$

4.1.9 자기 목적적 경험

<Table 13>은 변형된 시간감각에 대한 실험집단의 사전, 사후를 검증한 결과로서 평균값이 2.84에서 2.99로 증가하였으나 통계적으로 유의미한 차이를 가졌다($p < .05$)

<Table 13> The activity becomes autotelic(t-test)

	M	N	SD	t	df	p
Pre	2.84	29	1.02	2.095	28	.045
Post	2.99	29	.97			

p < .05

4.2 결과 해석

기존 연구 분석을 통해서 스크래치와 같은 블록기반 언어가 학습자의 몰입에 영향을 준 것이 엔트리와 같은 블록 기반언어에도 적용되는지를 알아보고 또한 여기에 교수학습 과정에 몰입기반학습요소가 학습자의 몰입요소에 어떤 영향을 주는지를 검증하였다.

검증결과 도전-기술 균형, 구체적 피드백, 행동-지각 일치, 과제 집중, 통제감, 자의식 상실, 자기 목적적 경험 등이 실험전과 실험 후를 비교한 결과 유의하다는 결과를 얻었다. 즉 9가지 학습몰입 요인 중 7가지의 요인에 영향을 주고 있음을 알 수 있었다. 그러나 명확한 학습목표와 변형된 시간감각은 유의하지 않은 결과를 얻었다.

원인을 유추하면 수업을 통해 프로그램의 결과를 미리 보여주고 무엇을 작성할 것인가에 대한 목표 제시 유형이 학생들에게는 오히려 프로그램 작성이라는 인지적 부하를 야기하고 직접해보지 않은 코딩 작업에 대한 두려움 등이 목표에 대한 명확성 인식에 영향을 주지 못한 것으로 생각된다. 따라서 목표제시에 좀 더 단계적인 목표 제시와 쉬운 내용을 소개하는 방안이 요구된다.

변형된 시간감각이 유의하지 않은 결과는 2시간의 짧은 수업시간에 여러 단원의 프로그램의 학습은 학생들에게 프로그램에 몰입을 통한 변형된 시간감각에 영향을 주지 못한 것으로 해석된다. 따라서 3시간 내지 4시간의 블록강의를 하면 보다 심도 있는 교육을 통해 변형된 시간감각에도 영향을 주지 않을까 예상해본다.

이러한 7가지 요인은 엔트리 프로그래밍 언어 교육에 다음과 같은 시사점을 준다. 첫째, 도전-기술 균형, 구체적 피드백으로 학생들에게 단계적으로 쉬운 프로그래밍 과제를 주어서 계속적으로 몰입하도록 하여야 한다. 가령 처음에는 예제 문제를 주고 그 다음단계에서는 여러가 포함된 문제를 풀게 하는 등 단계적인 도전과제를

제시하는 것이다. 둘째, 행동-지각 일치, 과제 집중, 통제감, 자의식 상실 등의 요소로 학생들이 관심을 갖는 분야에 관한 과제제시이다. 예비교사에게 엔트리로 교육에 관한 문제와 정답을 맞히는 예제는 코딩에 좀 더 몰입하게 되는 것을 발견하였다. 셋째, 자기목적적 경험은 학생들이 코딩이 학습이 아니라 자체를 즐기는 수준에 이르게 하는 것이다.

5. 결론

제4차 산업 혁명은 기업이나 국가뿐만 아니라 교육전반에도 새로운 개혁과 노력을 요구하고 있다. 이런 환경에서 최근 우리나라의 초중등 교육에서도 소프트웨어교육을 강화하려는 시도가 나타나고 있다. 본 연구에서는 주어진 짧은 프로그래밍 교육시간에 최대의 교육 효과를 주기 위해 텍스트기반이 아닌 블록 기반의 엔트리(Entry) 라는 교육용 프로그래밍 언어를 도구로 흥미 유발 콘텐츠의 적용, 몰입교수학습 요소의 적용으로 학생들의 몰입도에 영향을 주는지 검증해보았다. 예비교원 29명을 대상으로 약 3개월간의 Entry 프로그래밍 교육과정에 몰입적 학습과정을 적용하였다. 몰입을 위한 엔트리 프로그래밍 수업으로 학습형 콘텐츠의 활용과 엔트리교재를 활용하였다.

학습 몰입도 검사는 김아영[1]의 ‘성인용 학습 몰입 척도 개발 및 타당화’ 연구의 학습 몰입요인 9개 요인을 측정하기 위한 29문항을 적용하였고, 학생의 몰입도에 미치는 효과를 알아보기 위해 대응표본 t-검증을 실시하였다.

검증결과 도전-기술 균형, 구체적 피드백, 행동-지각 일치, 과제 집중, 통제감, 자의식 상실, 자기 목적적 경험 등이 실험 전과 실험 후를 비교한 결과 유의하다는 결과를 얻었다. 즉 9가지 학습몰입 요인 중 7가지의 요인에 영향을 주고 있음을 알 수 있었다.

본 연구에서는 기존의 연구에서처럼 단순히 블록언어 프로그래밍 언어를 교육하는 것이 학습자의 몰입을 극대화 하지는 못할 것이라는 개념에서 출발하였다. 따라서 프로그래밍 교육에 엔트리 언어의 선택, 흥미유발학습콘텐츠의 적용, 몰입기반 학습 요소를 적용하였다.

본 연구 제한은 29명의 학생들만을 대상으로 실험한

결과이므로 대상집단 수가 많아지고 학습의 주제와 시간이 변화가 될 경우에는 그 결과가 달라질 수 있다. 만약 학습 주제가 보다 생활기반 동기 유발 적이고 학습의 시간도 2시간이 아닌 3시간 내지 4시간정도의 강의와 더불어 프로젝트 학습유형으로 진행된다면 본 실험에서 유의하지 못한 결과를 얻은 ‘명확한 학습목표’와 ‘변형된 시간감각’등의 몰입요소가 결과가 달라질 수 있음을 밝힌다.

또한 블록언어 프로그래밍의 적용과 몰입기반 학습요소 중 어느 부분이 더 학습자의 몰입에 영향을 주었는지에 대한 자세한 검증에 대해서는 후속 연구가 필요하다.

참고문헌

- [1] A-Young Kim, Ha-yan Tak, Che-Hee Lee(2010). Development and Validation of a Learning Flow Scale for Adults. *The Korea Journal of Education Psychology*, 24(1), 39-59.
- [2] Csikszentmihalyi. M(2003). *The Technique of Flow*, Dubula Press.
- [3] Csikszentmihalyi. M(2004). *Flow*. Hanwolrim Press.
- [4] Entry(2016). Following and Learning Entry Programming(Beginner Course), <https://play-entry.org/tt#!/basic/materials>.
- [5] Eun Kyoung Lee, Young Jun Lee(2008). *The Effects of Scratch Based Programming Education on Middle School Students' Flow Level and Programming Achievement*. Middle School Education 56(2), 359-382.
- [6] Gomes, Anabela, Mendes, A.J.(2007), Learning to Program - difficulties and solutions, *International Conference on Engineering Education*, 121-125.
- [7] In-Whan Yu(2013). The Effects on Flow at Using Robots of Introductory Programming Course. *Journal of the Korea Association of Information Education*, 17(3), 329-337.
- [8] Kap-Su Kim(2016). Recognition Analysis of Elementary Teachers for Software Education of 2015 Revised Korea Curriculum. *Journal of the Korea Association of Information Education*, 20(1), 47-56.
- [9] Kay, J., Barg, M., Fekete, A., Greening, T., Hollands, O., Kingston, J & Crawford, K.(2000). Problems-based learning for foundation computer science courses. *Computer Science Education*, 10(2), 109-128.
- [10] Keong-Mi An, Won-Sung An, Yun-Cheul Choi(2011). The Effect of Scratch Programming Education on Learning-Flow and Programming Ability for Elementary Students. *Journal of the Korea Association of Information Education*, 15(1), 1-10.
- [11] Klaus Schwab(2016). *The Fourth Industrial Revolution*, New Present Press
- [12] Kolling, M. & Rosenberg, J(1996). Blue- A Language for Teaching Object-Oriented Programming, *Proc. of the 27th SIGCSE Technical Symposium on Computer Science Education*, 190-194.
- [13] Kyung-Hyun Kim(2011). The Effects of the Robot Based Instruction on Improving Immersion Learning. *The Journal of Korea Association of Computer Education*, 14(2), 1-12.
- [14] Lai, A., Yang, S.(2011). The learning effect of visualized programming learning on 6th graders' problem solving and logical reasoning abilities. *In: International Conference on Electrical and Control Engineering (ICECE)*, 6940-6944.
- [15] Lamb, A., Johnson, L.(2011). Scratch: computer programming for 21st century learners. *Teacher Librarian*, 38(4),64-68.
- [16] Milne, I. Rowe, G(2002). Difficulties in learning and teaching Programming - Views of Students and Tutors, *Education and Information Technologies*, 7(1), 55-66.
- [17] Nelson, J.(2009). Celebrating Scratch in libraries: creation software helps young people develop 21st-century literacy skills. *School Library*

Journal, 20-21.

- [18] Ornelas Marques, F., Marques, M.T.(2012). No problem? No research, little learning ... big problem!. *Systemic, Cybernetics and Informatics*, 10(3), 60-62.
- [19] Perkins, D., Hanconck, C., Hobbs, R., martin, F. & Simmons, R(1989). Conditions of learning in novice programmers. In Soloway & Spohrer: *Studying the Novice Programmer*, 261-279.
- [20] Seung-Young Shin(2012). Factor Analysis of Elementary School Student's Learning Satisfaction after the Robot utilized STEAM Education. *The Journal of Korea Association of Computer Education*, 15(5), 11-22.
- [21] Soloway, E. & Spohrer, J.(1989). Studying the Novice Programmer, Lawrence Erlbaum Associates, Hillsdale, New jersey. 497.
- [22] Sung Jin Hwang, Jeong Won Cho, Young Jun Lee(2014). Development of Education Plan of Using App inventor for improving Informatics Gifted Elementary Students' Learning Flow. *The Proceeding of Korea Association of Computer Education*, 18(2). 185-189.
- [23] Young-Suk Park(2012). The Effects Blended Robot Education Program on Learning Motivation and Immersion of Children. *The Journal of Korea Elementary Education* 23(4). 121-140.
- [24] Wilsonm, A., Moffat, D.C.(2010), Evaluating Scratch to introduce younger schoolchildren to programming, <http://scratched.media.mit.edu/sites/default/files/wilson-moffat-ppig2010-final.pdf>, 1-12.

저자소개



한 규 정

1991. 중앙대학교 컴퓨터공학과 (박사)
 2004.~2006. 플로리다주립대 교육공학과(연구교수)
 2007.~2009. (사)한국정보교육학회 회장 역임
 2009.~2010.캘리포니아주립대 (샌버나디노) 수학 및 과학 및 교육공학과 방문교수
 1992.~ 현재 공주교육대학교 컴퓨터교육과 교수
 2013-현재 국제청소년로봇연맹 회장
 관심분야 : SW 교육방법, 피지컬 컴퓨팅,
 E-mail: kyujhan@daum.net