

경험 중심 교육을 기반으로 한 소프트웨어 교육 방안

진광훈^o, 이명숙^{*}

^o계명대학교 컴퓨터공학과 전산교육

^{*}계명대학교 타블라라사 칼리지

e-mail: kwanghun7843@naver.com^o, mslee@kmu.ac.kr^{*}

Software Education based on Experiential Education

Kwang-Hun Jin^o, Myung-Suk Lee^{*}

^oDept. of Computer Engineering, Keimyung University

^{*}Tabula rasa College, Keimyung University

● 요약 ●

본 연구는 기존의 소프트웨어 교육 방법이 가지는 문제점을 분석하고, 이를 해결하기 위한 방안으로 존듀이와 프레네의 경험 중심 교육 방법에 기반을 둔 새로운 소프트웨어 교육 모델을 제안한다. 경험 중심 기반의 새로운 소프트웨어 교육 방법은 학습자들의 자율성을 보장하고, 학습자들 간의 상호작용을 통해 일상생활의 문제점을 해결할 수 있는 문제 해결력을 기르는데 중점을 두고 있다. 따라서 본 연구는 자유로운 사고방식으로 일상생활의 다양한 문제를 발견할 수 있는 능력과 다양한 경험을 통해 이를 해결할 수 있는 능력을 기르도록 체계화된 소프트웨어 교육 모델을 구성할 것이며 이를 통해 소프트웨어 교육의 발전에 기여하고자 한다.

키워드: 소프트웨어교육(SW Education), 경험중심교육(experience-centered education), 존듀이(John Dewey), 프레네(Freinet)

I. Introduction

인공지능의 발전으로 인한 4차 산업이 활발해지면서 소프트웨어의 중요성이 나날이 커지고 있다. 우리나라에서 또한 소프트웨어 인재를 양성하는 일이 국가의 중요한 시책으로 주목받고 있다. 그에 따라 중학교는 2018년, 초등학교는 2019년 의무적으로 실시하게 되었다. 그리고 고등학교는 학교 재량으로 소프트웨어를 이해하고 제작하는 교육을 실시하게 되었다.

2000년 정보 통신 기술 교육 운영 지침에서는 소프트웨어 활용 교육이 중심이었던 반면, 2005년 운영지침은 이를 보완하기 위해 정보의 처리, 이해 영역에 알고리즘과 프로그래밍 교육을 추가하였으며, 2015년 소프트웨어 교육 운영 지침에서는 문제를 해결하는 역량을 기를 수 있는 컴퓨팅 사고력을 통해 일상생활의 문제를 해결하도록 구성하고 있다[1].

컴퓨팅 사고는 추상화와 분해를 통하여 복잡한 시스템이나 어려운 문제를 해결하는 것으로 이와 관련된 사고과정을 말한다. 또한 컴퓨팅 사고력은 이미 들어서 있는 4차 산업 시대에 누구나 갖추어야 할 기본적인 역량으로, 우리는 일상의 경험에서 컴퓨팅 사고를 접할 수 있다[2]. 한 가지 예로 라면을 끓일 때를 생각하자. “먼저 냄비에 물과 스프를 넣고 물이 끓을 때까지 기다린다. 물이 끓기 시작하면 면발을 넣고 익을 때 까지 기다린다. 면이 다 익으면 맛있게 먹는다.”

이처럼 일상생활에서 라면을 끓이는 것을 간단히 3가지 동작으로 표현할 수 있다.

그러나 현재 시행되고 있는 소프트웨어 교육은 학습자의 경험에서 출발한 교육이라기보다는 소프트웨어의 추상적인 개념에 좀 더 초점이 맞추어져 있었던 것이 사실이다[3]. 또한, 교사의 지도에 따라 똑같은 예제와 실습을 통해 똑같은 결과를 확인하는 수준에 머물러 있으며, 초등과정에서 학습한 내용(예, E-센서보드)을 중·고등과정에서 그대로 반복하는 경향이 있다. 따라서 효과적인 교육을 위해서 학습자 경험에 초점을 둔 교육내용을 편성할 필요가 있다.

II. 이론적 배경

1. 경험 중심 교육 이론

존 듀이는 학습자와 교육의 대립보다는 이들 간의 연속성을 강조하고, 교육중심 교육자들과 학습자 중심 교육자들 간의 이원론을 극복하고자 하였다. 가치 있는 경험은 경험 속에서 사고가 작동하는 것으로 말하고 있으며 경험의 성장을 위한 교과 발달의 세 단계를 다음과 같이 정리 할 수 있다[4].

1단계(구체적 활동)

듀이의 첫 번째 단계는 무엇인가 할 줄 알게 되는 단계로 학습자는 그 일을 하는 동안에 사물에 익숙하게 되고 일하는 과정을 터득하게 된다. 이 단계에서는 학습자가 유목적적인 활동에 참여함으로써 지식의 획득이 가능한 ‘놀이나 일’과 같은 구체적 활동을 제공한다.

2단계(언어적 상호단계)

이 단계에서는 구체적 활동 단계에의 직접적인 활동의 수행능력과 관련을 맺으면서 이를 확대하고 발전시킬 수 있는 수단으로서 언어화된 지식 또는 정보의 형태를 만든다. 즉, 학습자에게 의미 있는 교과과정의 중간 단계로 언어화된 지식 또는 정보를 통해 의미가 풍부해지는 단계이다.

3단계(논리적 사고)

최종 단계는 과학적 또는 논리적 지식의 형태로 과학적 지식이란 논리적인체계를 따라 조직된 지식을 의미하고, 어떠한 교과이든시간에 그 교과 자체의 지식에 들어있는 ‘논리적 함의’를 드러내는 과정을 의미하는 것으로 학습자의논리적인 사고능력을 향상시킬 수 있다[5].

위의 3단계에 따르면 학습자는 구체적인 활동에서 출발하여 언어적인 상호작용관계를 거쳐 합리적이고 논리적인 사고로 발전해 간다는 사실이고 교과는 성장의 고정된 표준이 아니라 잠정된 표준으로서의 의미를 지니게 된다는 것이다.

듀이가 경험 중심 교육의 일차적인 이론이었다면, 프레네는 일차적인 교육실천이었다. 자신의 경험을 토대로 다음과 같이 수업 기법의 기본 명제와 프레네 교육을 적용한 학교의 수업 사례 중 하나이다.

1. 모든 차원에서 이루어지는 자유로운 표현
2. 본문과 미디어 및 정보와 창조적으로 교류하기
3. 의사소통과 협동
4. 학교 학습을 실제 환경으로 지향시키기
5. 학습을 스스로 조직하기
6. 더듬어 찾아보는 식의 학습
7. 실천을 통한 학습
8. 간문화적 학습
9. 선발 없는 학습

수학카드 목록 이틀리에

학습자들이 원하는 공부를 원하는 만큼 원하는 속도로 할 수 있도록 고안한 학습카드를 활용한 방법으로 학습자들은 비치되어 있는 카드를 선택해 활용한다. 그리고 교사는 특별한 어려움을 지닌 학습자들의 학습을 도와주는 방식이다. 이 카드학습은 일차적으로 학습자가 스스로 공부하고 이차적으로 친구들의 도움을 주고받아가면서 공부하는 협동학습이다[6].

2. 기존 SW교육 방법

현재 대부분의 학교에서는 학습자와 지도하는 교사의 수준에 따라 다양한 소프트웨어 교육 방법이 존재하며, 대표적인 3가지만 알아보도록 한다.

첫째, 보드게임, 블루머블게임 또는 Fig 1과 같은 이미지 표현이다. 이 교육 방법은 언플러그드 학습으로 컴퓨터를 사용하지 않고 컴퓨터의 동작원리를 이해할 수 있는 구체적인 조작활동의 형태의 놀이 학습이다[7].

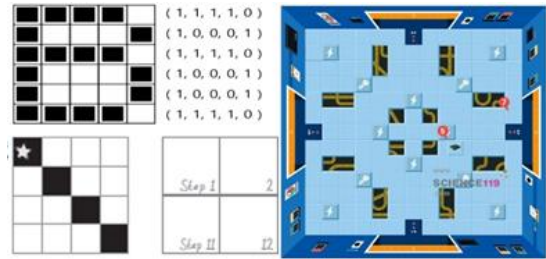


Fig. 1. 언플러그드 활동 [출처 : code.org, 팝콘에듀]

둘째, 교육용 프로그래밍을 통한 소프트웨어 교육이 있다. 교육용 프로그래밍 언어는 Fig 2와 같이 상대적으로 지도가 쉬운 스크래치 및 엔트리와 같은 블록 기반의 프로그래밍 언어, 문법적 이해와 전문적 지식을 필요로 하는 텍스트 기반의 프로그래밍 언어가 존재한다. 교육용 프로그래밍을 통한 교육은 주어진 문제를 해결하여 컴퓨터로 그 결과를 확인하는 것으로 예를 들어 블록 기반의 프로그래밍 언어를 사용하여 ‘시간에 따라 색상이 변하는 꽃잎 만들기’라는 주어진 문제를 해결하면, 해당 프로그램에 시간이 자남에 따라 색상이 변환을 꽃잎을 확인할 수 있는 것이다.

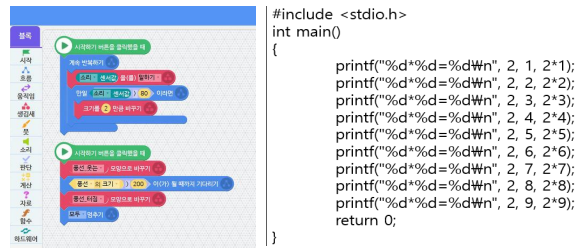


Fig. 2. 교육용 프로그래밍 언어

셋째, 위에 언급한 프로그래밍 언어를 물리적 즉, 눈으로 확인할 수 있는 피지컬 컴퓨팅이 존재한다. 피지컬 컴퓨팅이란 디지털 기술과 장치를 이용하여 현실세계의 모습을 전구(LED)와 그 외 물리적인 장치로 출력하는 것으로, 대부분 햄스터봇, E-센서보드 등을 사용한다.

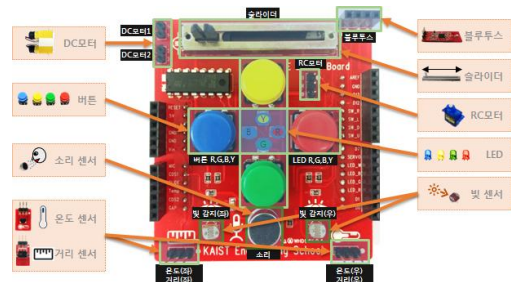


Fig. 3. 피지컬 컴퓨팅 [출처 : 초등컴퓨팅교사회]

위에 언급한 3가지의 대표적인 소프트웨어 교육 방법은 단계별로 교육이 이뤄지면 컴퓨팅 사고력을 키우는 것이 효과적으로 나타날 수 있다. 그러나 단계별 학습이 이뤄지지 않고 각각의 단계로 교육이 끝나는 점과 동일한 과정이 반복되는 문제가 있으며, 소프트웨어 교육을 지도하는 교사의 지도하에 똑같은 문제점을 해결하기 위해 따라하고 학습자들 모두 같은 방법으로 같은 결과를 확인하는 점에서 “문제를 해결하는 역량을 기를 수 있는 컴퓨팅 사고력을 통해 일상생활의 문제를 해결”이라는 운영지침에 맞지 않는다고 볼 수 있다. 또한, 물리적인 장치를 이용하는 피지컬 컴퓨팅 교육에서도 햄스터 붓을 이용한 미로 찾기 또는 라인트레이서 등의 한정된 실습을 통해 학습자들의 문제 해결력을 키우는 것에 대해 한계점을 지닌다. 이에 따라 학습자의 경험 중심 기반의 새로운 교육 방안을 다음 절에서 살펴본다.

III. 경험 중심의 SW교육 모델

1. 경험 중심의 SW교육 모델

본 연구에서 제안하는 교육 모델은 Fig.4와 같으며 다음과 같은 전제 조건을 필요로 한다.

1. 학습자를 지도하는 교사는 다양한 프로젝트를 수행할 수 있는 하드웨어 및 소프트웨어 지식이 충분해야 한다.
2. 교육이 이뤄지는 장소는 프래네 교육을 적용한다. 이 장소에서는 학습자들의 자유를 보장하고 개인 또는 그룹별로 학습을 진행할 수 있다. 또한 학습에 있어 학습자 간 의사소통이 가능하다. 지도교사는 편성된 그룹에는 높은 수준의 있는 학습자가 존재하도록 조절한다.
3. 학습자들의 단계별 수준 평가는 그룹 내 발표 또는 토의내용으로 파악한다.

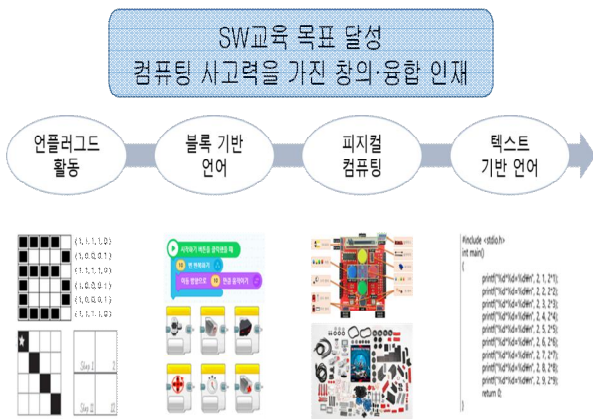


Fig. 4. 경험 중심의 SW교육 모델 [출처 : 레고코리아]

학습자의 컴퓨팅 사고력을 효과적으로 이끌어내고 더 나아가 창의 융합 인재를 육성하기 위해 듀이의 교과 발달 단계에 근거하여 다음과 같이 교과 과정을 새롭게 구성한다.

1단계(언플러그드 활동)

이 단계에서는 학습 활동을 통하여 언플러그드의 목적인 다양하고

복잡한 정보의 표현 방법과 알고리즘(순서도) 표현에 중점을 둔 단계이다.

필수 활동으로 Binary image와 Graph Paper programming을 들 수 있다. Binary image는 학습자들은 다양한 이진법으로 정보를 나타낼 수 있고 이를 통해 컴퓨터가 복잡한 정보를 어떻게 표현하는지 이해하고 설명할 수 있다. 그리고 Graph Paper programming 활동에서는 눈금으로 표시된 이미지 용지에 내가 원하는 그림을 표현하기 위해서 ←, →, ↑, ↓의 이동과 관련된 키워드와 √의 색칠에 관련된 키워드로 순서도를 작성할 수 있다. 더 나아가 일상생활에서 접할 수 있는 활동을 순서화시키는 활동을 할 수 있다.

2단계(블록 기반 언어)

두 번째 단계는 피지컬 컴퓨팅을 학습하기 위한 기초 학습 단계로 엔트리 또는 Lego MindStorm을 활용하여 1단계에서 학습한 알고리즘을 블록 기반의 교육용 언어를 사용하여 표현하고, 다양한 문제에 대해서 해결하는 과정이다. 또한 학습자들은 같이 문제를 해결하기 위해 노력하고 다른 해결방법에 대해 공유할 수 있다.

3단계(피지컬 컴퓨팅)

가장 중요한 단계로 1~2단계에서 습득한 컴퓨팅 사고력을 바탕으로 일상생활 뿐 아니라 여러 과목(수학, 미술, 과학 등)의 내용을 주제로 학습활동으로 수행하여 학습자들끼리 공유하는 단계이다. 이 단계에서는 교육용 아두이노 또는 Lego MindStorm을 이용하여 학습활동을 진행한다.

지도 교사는 이 단계에서 학습자가 블록 기반 언어에서 텍스트 기반 언어로 자연스럽게 넘어갈 수 있도록 도와주어야 한다. 엔트리의 경우 파이썬 코드를 볼 수 있는 환경이 마련되어 이 기능을 적극 활용하고, Lego MindStorm의 경우 RobotC 또는 BricxCC로 2단계에서 습득한 내용을 표현할 수 있도록 적극적인 태도로 지도해야 한다.

4단계(텍스트 기반 언어)

3단계에서 피지컬 컴퓨팅을 통해 텍스트 기반 언어를 자주 접한 학습자는 다른 학습자에 비해 부담을 덜 느끼게 된다. 3단계에서 엔트리로 활동을 진행하였다면 파이썬, Lego MindStorm을 사용하였다면 C언어로 활동을 진행하는 것을 권장한다.

이 단계에서는 학습자들은 블록 기반의 언어에서 학습한 내용을 텍스트 기반의 언어로 학습을 하고, 제공된 블록으로만 문제를 해결하는 것으로부터 자유로운 사고를 할 수 있도록 지도한다.

최종적으로 학습자들이 일상생활에서 가지는 의문점이나 학습자들이 하고 싶었던 주제를 피지컬 컴퓨팅 도구를 사용하여 프로젝트를 구성하고 이를 해결하기 위한 컴퓨팅 사고력을 기르는 데에 중점을 두어야 한다.

학습자들은 기존의 SW교육 방법은 프로그래밍 언어와 피지컬 컴퓨팅을 다른 것으로 판단하는 경우도 있다. 또한 교육용 프로그래밍 언어를 교육할 경우, 학습자들은 지루함을 느끼고 특히, 텍스트기반의 언어는 높은 진입장벽으로 어려움을 많이 느끼게 된다. 따라서 제안하

는 SW교육 방법으로 텍스트 기반 언어의 진입 장벽을 낮추고, 피지컬 컴퓨팅을 통해 학습자의 잠재적 능력이 나타날 수 있도록 지도해야 한다.

이 교육 모델에서의 지도교사는 프레네 교육에서의 마찬가지로 관찰자의 역할을 수행하며, 학습자에게 해답을 제공하는 것이 아니라 문제 해결을 위한 방향을 지시하고, 학습자간 토의를 정리해 주어야 한다.

III. Conclusions

현재 이뤄지고 있는 SW 교육의 대부분은 교사가 계획한 지도안에 따라 학습자가 똑같은 방법과 기지재(예, 햄스터봇)로 따라하는 교육 방법이 대부분이다. 이러한 기존의 교육방법은 학습자들의 컴퓨팅 사고력을 기르는 데에 있어 비효율적이다.

이에 본 논문에서는 경험 중심의 교육과정을 알아보고, SW 교육에 대한 새로운 교육 방법을 제시하였다. 본 논문에서 제시된 방법은 기존의 SW 교육 과정을 새롭게 구성하고 듀이와 프레네의 교육법을 부분적으로 적용한 것으로, 학습자들이 자유롭게 활동하며 컴퓨팅 사고력을 키울 수 있는 환경을 제공 할 수 있을 것으로 예상된다.

REFERENCES

- [1] NCIC 국가교육과정 정보센터, <http://ncic.kice.re.kr>.
- [2] Jeannette M. Wing “Computational Thinking”, *Communications of the ACM*, 49(3), 33-35, 2006
- [3] 조윤록, 이영준, “존 듀이의 교육철학이 소프트웨어에 주는 시사점”, *한국컴퓨터정보학회 학술발표 논문집*, 121-122, 2017. 1
- [4] 김재춘, “교육과정 제 2판”, *교육과학사* 91-116, 2017.9
- [5] 이은영, 최지연, “듀이의 교과외 진보적 조직에 터한 실과 단원 설계 모형과 예시 단원의 개발”, *한국실과교육학회지*, 24(3), 143-163, 2011.9
- [6] 송순재, “셀레스탱 프레네와 프레네 교육학”, *우리교육* 2011 여름, 210-220, 2011.6
- [7] 박진화, “놀이 활동 교육 자료를 활용한 중학교 정보 표현 학습이 학업 성취도에 미치는 영향”, *국교원대학교 대학원*, 2011