

스크래치2.0을 활용한 초등정보영재의 프로그래밍 교육방안

박재형[○], 최정원^{*}, 이영준^{*}

[○]한국교원대학교 컴퓨터 교육학과

e-mail: pjh3269@nate.com[○], cjw0829@daum.net^{*}, yjlee@knue.ac.kr^{*}

Programming education plan for elementary informatics gifted using Scratch 2.0

Jaehyeong Park[○], JeongWon Choi^{*}, YoungJun Lee^{*}

[○]Dept. of Computer Education, Korea-National University of Education

● 요약 ●

본 논문에서는 스크래치2.0을 활용한 초등정보영재의 프로그래밍 교육 방안을 제안하고자 한다. 스크래치는 블록을 쌓는 것만으로 프로그래밍이 가능하기 때문에 프로그래밍을 처음 접하는 초등정보영재 학생의 프로그래밍 교육과정으로 적합하다. 또한 2.0으로 버전업되면서 스크립트 보기(See inside)나 리믹스(Remix)기능이 추가되어 프로그램의 공유 및 협업이 더 용이하여졌다. 따라서 이러한 스크래치2.0의 기능을 사용한다면 학습자에게 필요한 협업 및 의사소통 능력을 기르게 하는데 적합하다고 할 수 있다. 또한 영재학생들은 학습자 스스로 목표를 정하고 계획을 능동적으로 수행해나갈 때 높은 성과를 보이는 특성이 있다. 따라서 자기 주도적 학습 모형을 영재학생들에게 사용한다면 더 효과적이다. 이에 본 연구에서는 스크래치 2.0을 활용하여 Treffinger의 자기 주도적 학습모형을 적용한 프로그래밍 교육 방안을 제안하였다. 추후에 초, 중, 고 영재기관 연계를 위한 프로그래밍 교육위계에 대한 연구가 필요하다.

키워드: 정보영재(informatics gifted), 프로그래밍 교육(programming education), 스크래치 2.0(scratch 2.0)

I. 서론

21세기 미래사회는 지식이 폭발적으로 증가하고 그 생성과 소멸의 주기가 짧아지고 있다. 이렇게 지식이 빠르게 변화하고 있는 정보화 사회에서 전문적인 정보화 능력을 갖춘 인재를 양성하는 것은 자아의 실현과 더불어 국가의 발전을 위해 매우 중요한 일이라 할 수 있다. 또한 21세기 학습을 위한 파트너십(Partnership for 21st Century Skills)에서는 창의력, 비판적 사고력 및 문제해결력, 협업능력, 의사소통능력을 핵심 학습 역량으로 들고 있다. 많은 연구에서 살펴볼 수 있듯이 프로그래밍 교육을 통해 창의력, 논리적사고력, 문제해결력을 길러 줄 수 있다[1][2]. 거기에 최근 버전업된 스크래치 2.0의 Remix 및 Backpack기능 등을 적절히 이용 한다면 학생들에게 협업능력 및 의사소통능력을 기르는데 많은 도움이 될 수 있을 것이다.

이에 본 연구에서는 스크래치 2.0 교육용 프로그래밍 언어를 활용한 초등정보영재의 프로그래밍 교육 방안을 제안하고자 한다.

II. 관련 연구

1. 스크래치

스크래치(scratch)는 2007년 MIT Media Lab의 연구팀에 의해 프로그래밍 교육을 위해 개발된 것으로 객체지향을 지원하는 언어이다.

문법위주의 교육보다는 프로그램의 구조를 익히는 것과 논리적인 문제에 초점을 맞추었기 때문에 초보자의 입문과정으로 적합하다[3].

1.1 스크래치의 특징

다른 교육용 프로그래밍 언어에 비해 다음과 같은 특징을 가지고 있다. 첫째, 스크래치는 레고나 퍼즐조각처럼 블록을 쌓으며 놀이하듯이 학습하는 프로그래밍 언어이다. 둘째, 다양한 멀티미디어 소스를 활용하여 재미있는 프로그래밍 활동을 할 수 있다. 셋

제, 병렬 실행 및 단계적 실행이 가능하다. 넷째, 공유 및 공동 작업이 가능하다. 다섯째, 외부에서 센싱한 값을 받아들여 프로그래밍 하는 것이 가능하다[4].

스크래치는 블록을 쌓는 것만으로 프로그래밍이 가능하기 때문에 명령어를 암기하고 직접 입력해야 하는 부담감이 없으며 블록에 쓰여 있는 단어만으로도 직관적인 파악이 가능하고 프로그래밍을 처음 접하는 초등학교생들도 쉽고 편하게 다룰 수 있다[5].

1.2 스크래치 2.0

2009년 1.4버전 발표이후 4년 만에 발표한 버전으로 가장 큰 특징은 클라우드 방식이다. 마이크로소프트사의 오피스 365나 구글 드라이브처럼 운영체제와 웹 브라우저 환경 PC의 성능과 관계 없이 언제 어디서나 인터넷만 연결되어 있으면 바로 사용이 가능한 클라우드 시스템이라는 점이다. 두 번째 특징은 사용자가 필요한 명령어 블록을 다른 블록들을 조합하여 만들 수 있어서 비교적 구조화된 프로그래밍이 가능하게 되었다. 세 번째 특징으로는 다른 친구들의 프로젝트를 보면서 스크립트 보기(See inside)를 통해 직접 코드를 확인할 수 있고 리믹스(Remix)기능을 통해 다른 친구들의 프로젝트를 내 프로젝트로 저장해서 변경해 볼 수 있다. 또한 변경된 프로젝트를 공유하게 되면 최초 프로젝트를 만든 사용자와 연결이 된다. 이 기능을 이용하면 교사가 최초의 프로젝트를 제공하고 학생들과 협업을 통해 창의적이고 다양한 프로젝트의 구성이 가능하다. 마지막으로 PC에 연결된 웹캠을 이용하여 움직임 인식한 프로그래밍도 가능하여졌으며 저장매체에 별도로 저장하지 않아도 로그인 상태에서 실시간 저장이 가능해졌다[6].

2. 프로그래밍 교육

2.1 프로그래밍 교육의 목적

프로그래밍 교육의 목적은 크게 두 가지 관점으로 나눌 수 있는데 그중 하나는 프로그래밍 언어습득에 초점을 두는 것으로 특정 프로그래밍 언어의 습득 및 여러 명령어의 혼합된 사용에 관계된 규칙의 이해를 강조하는 관점이며 다른 하나는 고등 인지기능의 습득에 초점을 두어 프로그래밍 과정에서 요구되는 문제해결과 지속적 오류검증 및 수정 작업이 요구하는 반성적 사고를 통해 고등 인지기술 향상에 초점을 두는 것이다[7]. 초등정보영재학생들에게는 두 번째 관점에 초점을 두어 다양한 인지적 기능을 사용하여 사고력을 향상시키는데 더 우선적인 목적을 두는 것이 바람직하다.

2.2 정보영재의 프로그래밍 교육의 효과

정보영재에게 프로그래밍을 지도함으로써 얻을 수 있는 효과는 다음과 같다. 첫째, 학습자의 논리적 사고력, 문제해결력 등의 고등 인지 능력을 신장시킨다. 또한 학습자는 프로그램을 완성하는 과정에서 성취감과 자신감을 획득하고 토론과정을 통해 협동심과 상호작용능력을 향상시킬 수 있다. 둘째, 오류 수정 과정을 통하여 스스로 문제를 해결함으로써 자신의 사고 과정을 점검할 수 있

으며 오류 수정 과정 자체가 교육 과정이 된다. 셋째, 현실 생활의 문제의 원인을 파악하고 해결책을 찾을 수 있는 능력을 기를 수 있다. 넷째, 컴퓨터를 더욱 깊이 이해하고 활용할 수 있는 기초를 닦을 수 있다[4].

3. 자기 주도적 학습

3.1 자기 주도적 학습의 원리

자기 주도적 학습의 원리는 첫째, 학습자 자신의 내면적 동기 유발로부터 출발한다. 둘째, 학습자가 자신의 모든 학습과정을 계획하고 조직한다. 셋째, 학습자 스스로 주도권을 쥐고 능동적으로 계획을 실행해 나간다. 넷째, 학습자는 자신에게 최적으로 가용한 학습기회, 학습방법, 학습자료 등을 스스로 선택한다. 다섯째, 타인과의 경쟁을 벌이기보다는 자아의 내적인 경쟁을 벌인다. 여섯째, 학습자는 자기스스로에게 교수자의 역할을 발휘한다. 일곱째, 학습자 스스로 세운 평가 준거들에 의해 스스로를 평가한다[8].

3.2 Treffinger의 자기 주도적 학습 모형

Treffinger의 자기 주도적 학습은 학생들이 자신의 학습과정에 보다 많이 참여할수록 보다 더 효과적인 학습이 일어난다는 점과 학생들이 스스로 자신이 학습한 내용을 결정하면 동기 유발이 잘 된다는 이 두 가지 가정에 기초하고 있다. 이모형은 영재들이 자신의 관심분야를 직접 계획하고 수행하고 평가할 때 그들의 잠재능력과 재능이 최대로 개발될 수 있다.

자기 주도적 학습 모형은 4단계로 이루어져 있으며 이는 다음과 같다[9].

1) 교사주도 수업단계(command style)

교사는 교육목표를 설정하고, 학습과 개별학생의 학습활동을 처방하고, 수업의 모든 활동을 주도하며, 평가의 준거도 마련한다.

2) 영재주도 학습1단계(task style)

교사가 학생들의 관심분야와 학습속도에 따라서 학습 프로그램의 내용과 수준에 대한 선택의 기회를 준다. 이 단계에서 교사가 학생들을 위하여 다양한 학습활동이나 프로젝트 등을 개발하고, 학생들에게 자신의 능력과 학습속도에 맞추어 학습활동을 선택할 기회를 제공한다.

3) 영재주도 학습2단계(peer-partner style)

학생들은 자신의 학습활동, 목표, 평가 등을 결정하는데 참여하는 등 좀 더 적극적인 역할을 한다. 교사는 학생들이 무엇을 할 것인가에 대한 창조적인 선택을 하도록 유도하며, 학생들이 배우게 될 내용을 결정하고 선택하는 과정에 관여한다.

4) 영재주도 학습3단계(self-directed style)

학생들이 학습할 내용을 스스로 선택하도록 하고, 이때 교사의 주된 역할은 필요한 경우에 자료나 정보를 제공해 주는 역할을 맡는다.

이모형을 적용할 때 가장 유의할 점은 처음 단계부터 마지막단 계까지 무조건 순차적으로 적용하거나 또는 모든 학생들에게 일률적으로 적용할 필요는 없다. 교사는 개인차를 고려하여 개인의 능력수준에 적합한 자기주도 학습의 단계에 참여할 수 있도록 도와 주어야 한다.

III. 본 론

1. 프로그래밍 교육방안

Treffinger의 자기 주도적 학습 모형을 적용하여 초등정보영재 학생을 위한 프로그래밍 교육 방안을 제안하면 [표1]과 같다.

표 1 초등정보영재를 위한 프로그래밍교육방안
Table 1. Programming education plan for informatics gifted

단계(시간)	내용
교사 주도 수업 단계 (2주)	<ul style="list-style-type: none"> · 프로그래밍의 이해 · 스크래치의 기본적인 개념과 특징 · 스크래치의 여러 블록들을 활용 · 프로그램의 흐름 제어(조건문, 반복문, 메시지전달), · 클론 만들고 활용하기 · 비디오 카메라 활용하기
영재주도 학습1단계 (1주)	<ul style="list-style-type: none"> · 간단한 게임 프로그램 소개 (스포츠헤, 미션수행형 등) · 팀별 개발할 게임 선택
영재주도 학습2단계 (2주)	<ul style="list-style-type: none"> · 게임 종류별 추가 기능 탐색 · 스포츠헤 게임 <ul style="list-style-type: none"> - 점수, 모션인식, Player추가, 종목변경, Level 상승 등 · 미션 수행형 게임 <ul style="list-style-type: none"> - 점수, 시간제한, 단계추가, 아이템획득, 장애물 추가 등
영재주도 학습3단계 (1주)	<ul style="list-style-type: none"> · 팀별 종합게임 완성 · 평가 및 발표

단계별로 활동내용을 살펴보면 첫 번째 단계인 교사주도 수업 단계에서는 프로그래밍의 이해와 스크래치의 블록들을 활용한 기본적인 프로그램 작성 방법을 간단한 예제와 함께 학습한다. 두 번째 단계인 영재주도 학습1단계에서는 스크래치로 작성할 수 있는 다양한 게임프로그램을 소개하고 프로그램 작성방법을 스크래치 2.0에 있는 스크립트 보기(See inside) 기능을 활용하여 살펴본 후 학생들의 관심과 수준에 따라 프로그램을 선택 한다. 또한 필요한 블록모음을 Backpack에 담아두어 프로그램을 수정하거나 필요할 때 적절히 활용할 수 있도록 한다. 세 번째 단계인 영재주도 학습 2단계에서는 게임별로 추가할 수 있는 기능 등을 살펴보고 학습1 단계에서 팀별 선택한 게임에 스크래치2.0의 리믹스(Remix) 기능을 이용하여 게임에 추가하고 싶은 기능을 팀원 개개인이 추가한 후 팀별 협의 과정을 갖는다. 교사는 [그림 1]과 같이 리믹스 트리를 이용하여 구성원들의 참여 정도를 살펴보고 적절한 지도를 한다.

마지막으로 영재주도 학습3단계에서는 팀별 협의 과정을 거쳐 논의된 기능들을 추가하고 오류를 수정하여 종합게임을 완성한다. 발표를 통해 게임 개발 과정에서 사용한 알고리즘 및 기능 등을

소개하고 다른 팀에서 배울 점 및 보완할 점을 피드백 받는다.

IV. 결 론

본 연구에서는 스크래치2.0을 활용한 초등정보영재 프로그래밍 교육 방안을 제안하였다. 스크래치는 블록 쌓기를 통해 프로그래밍이 가능한 직관적인 언어로 학습자들이 보다 쉽게 프로그래밍을 이해하고 작성할 수 있다. 최근 2.0으로 버전업되면서 See inside,

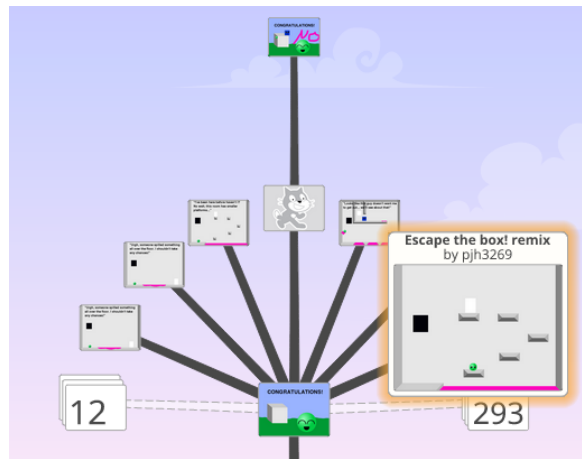


그림 1. 리믹스 트리 활용
Fig 1. Remix tree utilization

Remix, Backpack기능 등이 추가 되어 협업이 더 용이하여 졌다. 이러한 기능들을 잘 활용하고 거기에 영재를 위한 자기주도 학습 모형을 정보영재교육과정에 적용한다면 정보영재교육에 좋은 사례가 될 수 있을 것이다. 하지만 프로그래밍의 위계에 따른 초, 중, 고 정보영재의 교육과정 대한 연구는 부족한 형편이다. 교육기관별 정보 영재교육과정 연계에 따른 더 많은 연구가 요구된다.

참고문헌

[1] M. H. LEE, "Effect of a Programming Class Using Dolittle on Enhancing Creativity, Problem Solving Ability, and Interest in Programming", M.A. thesis, Jeonju National University of Education, 2009.

[2] S. K. Jeon, "The Effect of Divergent Thinking centered CPS Programming Education on Creative Problem Solving Ability", M.A. thesis, Korea National University of Education, 2011.

[3] H. H. Lee, "Effects of using scratch on the fundamental programming concepts learning of professional high school", M.A. thesis, Korea National University of Education, 2009.

[4] J. H. Chae, "Scratch Lesson Design for Programming Concept Learning of Gifted Students in Informatics",

- M.A. thesis, Chungnam National University, 2010
- [5] G. M. An, "The Effect of the Programming Education on the Elementary School Student's Learning-Flow and programming ability" M.A. thesis, Gyeongin National University of Education, 2010.
- [6] Scratch Site, <http://scratch.mit.edu>
- [7] D. H. Shim, "Effects of Learning of Visual Basic and Scratch Programming on the Logical Thinking and Problem Solving Ability", M.A. thesis, Korea National University of Education, 2012.
- [8] S. H. Lee, "Explore ways of teaching and learning", Yang Seo Won, Seoul.
- [9] Treffinger, D. J, "Teaching for self-directed learning : A priority of the gifted and talented. Gifted child Quarterly Vol. 19, pp. 46-59, 1975