



구성주의 관점의 코딩 교육 전략

2021년 1학기 작성
춘천교육대학교 교수 김홍래



학습 목표

1. 새로운 교양으로서 코딩의 의미를 설명할 수 있다.
2. 구성주의 관점의 코딩 교육을 위한 접근법을 설명할 수 있다.



The Comeback of coding

Project Headlight(Boston, 1990)

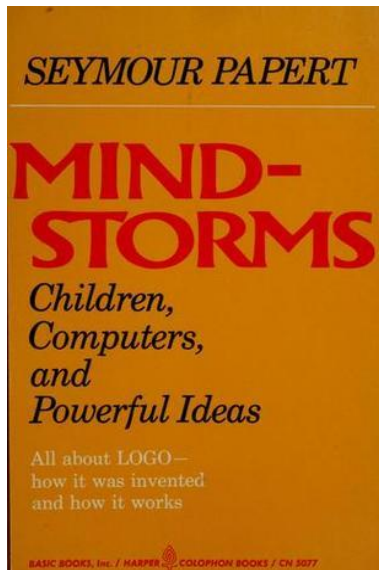


Figure 1.2

Project Headlight at a Boston elementary school, 1990. Photo by L. Barry Heatherington.



코딩 교육을 위한 철학적 배경



Papert, Seymour(1980).
Mindstorms - children,
computers and powerfull ideas.
NY:Basic Books.



- Constructionism
- Computational Thinking

21세기 새로운 교양으로서 코딩

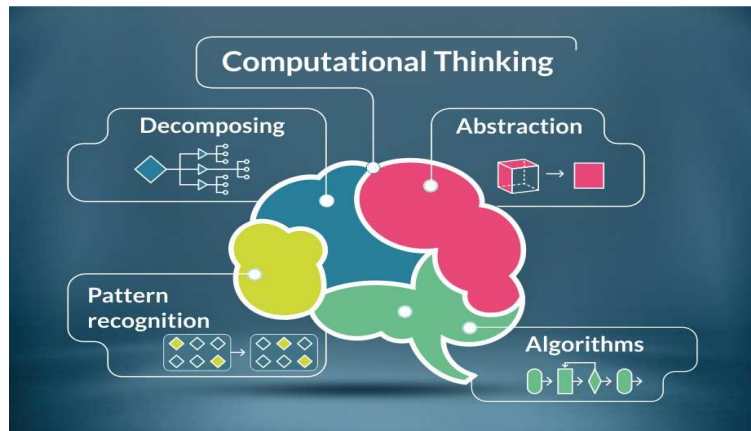


- 전문가의 전유물이었던 컴퓨터가 모든 사람들이 이용하는 도구가 되었음
- 인간은 다양한 형태의 컴퓨터를 통하여 자신을 표현
- 학자와 전문가의 영역에 있던 코딩이 교사에게는 중요한 기술
- 모든 학생을 위한 새로운 교양이 됨

Papert의 학습에 대한 인식

- Papert의 철학이 30년만에 학교로 되돌아온 까닭은 무엇인가?
 - 학생들에게 **Computational Thinking**을 가르하고자 하는 요구를 반영
 - 컴퓨터 과학자처럼 사고하는 방법을 학습함으로써 일상생활의 문제를 해결
 - 일상생활에서 사용하는 시스템을 설계하고 다른 영역으로 확장
 - 디지털 세대 젊은이들의 문화를 특징짓는 **DIY(Do It Yourself)**정신의 확산

What is Computational Thinking?



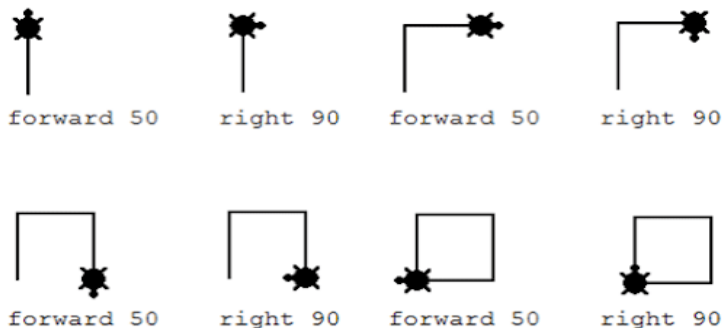
Papert의 학습에 대한 인식



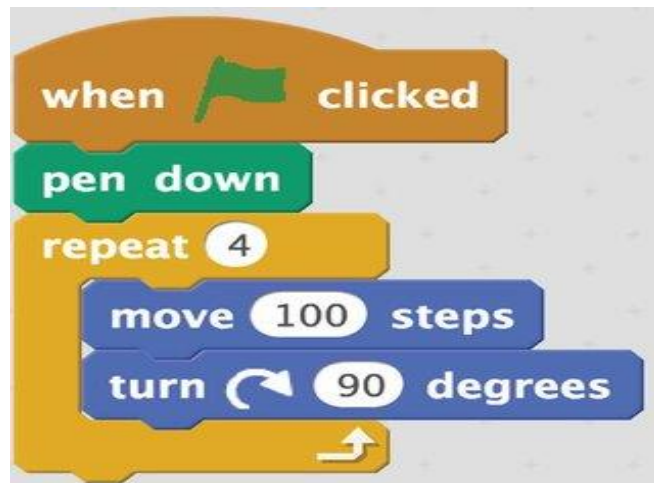
- **Papert**에게 학습이란?

- 개인적으로나 사회적으로 관련있는 인위적인 결과물을 탄생시키고, 오래된 지식과 새로운 지식을 결합하고, 타인과 교류하는 과정
- 참여와 공동작업, 컴퓨터 사용이 어떤 식으로 어린이들이 디지털 세상을 배우고 그 세상에서 살아갈 수 있도록 도울 수 있는지 탐구

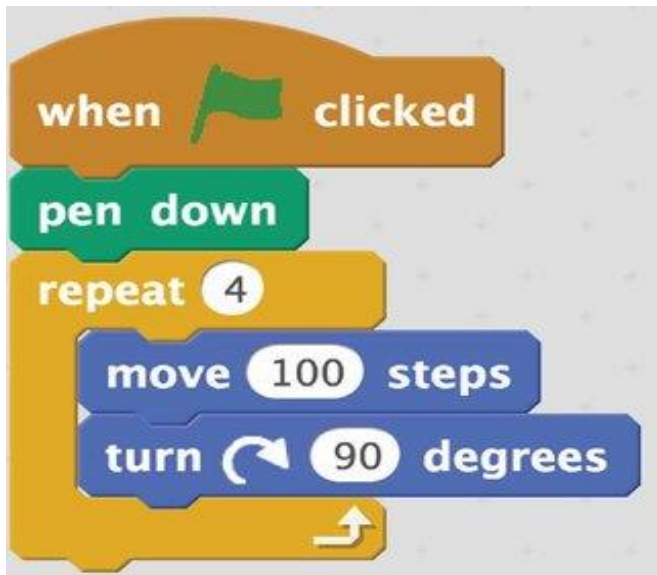
도구를 프로그래밍하다. 세상을 프로그래밍하다.



© 2000 Logo Foundation



도구를 프로그래밍하다. 세상을 프로그래밍하다.



- 코드 작성 방법을 배우는 과정
 - 절차를 구체적으로 설명
 - 패턴 인식(pattern recognition)
 - 디버깅(Debugging)
 - 프로그램이 예상대로 동작하지 않을 때
- 프로그램 작성 방법의 이해
- 새로운 아이디어를 이해하는 방법을 습득
- 학습하는 방법을 배우

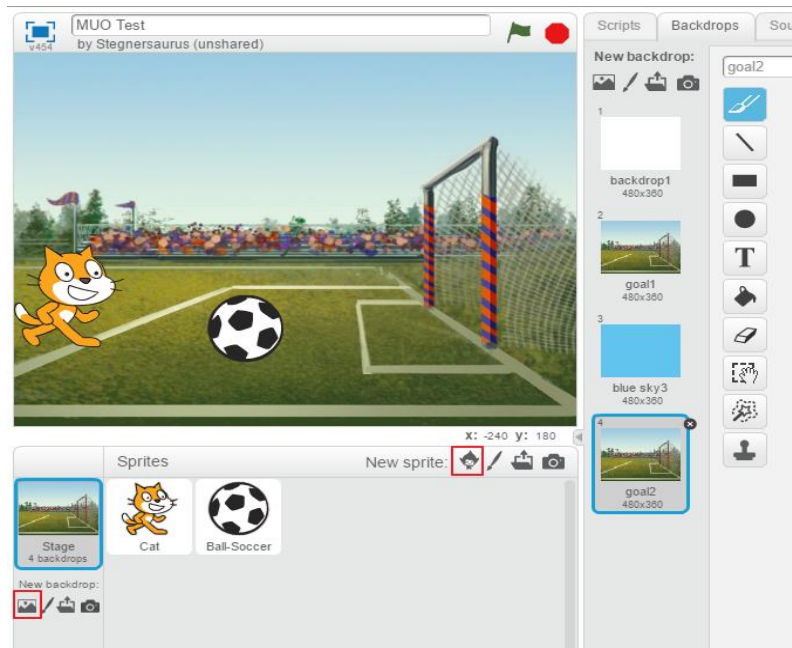
코딩 교육과 구성주의 이론

- 코딩교육에서 구성주의(constructionism) 이론의 영역
 - Personal Domain
 - From code to Application
 - Social Domain
 - From tools to Communities
 - Cultural Domain
 - From Scratch to Remix
 - Tangible Domain
 - From Screen to Tangibles

개인적 영역

From code to application

- 학습자가 프로그래밍 과정을 통하여 스스로 지식을 생산
- 자신이 만든 지식에 대해 개인적으로 동질감을 느끼는 과정
- 지적인 경험과 더불어 감성적 가치가 덧붙이는 과정
- Object-to-think-with
- 요약 정리



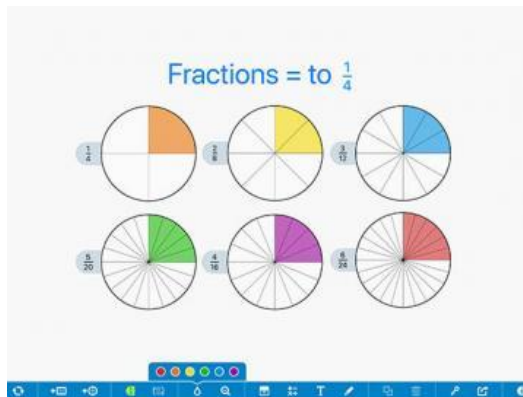
개인적 영역

From code to application



Figure 1.2
Project Headlight at a Boston elementary school, 1990. Photo by L. Barry Heatherington.

- 4학년 학생들은 비율과 나눗셈을 직접 응용해 혼자서 관련 자료를 제작하는 경험을 함
- 3학년 학생들에게 분수의 개념을 가르치기 위하여 게임과 시뮬레이션을 만들
- 학생들은 컴퓨터 화면 앞에 앉아 분수 문제를 푸는 대신 다른 학생을 위한 교육용 소프트웨어 디자이너로 변신함



CAI : fractions
Computer
Aided
Instruction

개인적 영역

From code to application

- 왜 초기의 코딩 교육은 성공하지 못하였을까?
 - 코딩은 다른 수업과 분리하여 별도의 내용만을 학습하였다.
 - 프로그래밍의 개념과 기능에 대한 학습이 주를 이루었다.
 - 초중학교에서 로고는 다른 교과와 통합되지 못하였다.
 - 코딩의 목적이 일상의 삶이나 교과의 주제와 연결되지 못함으로써 그 기반을 잃었다.
 - 교사들은 코딩의 가치를 이해할 수 없었으며 교과와의 연결점을 알지 못하였다.
- David Palumbo
 - 프로그래밍 교육 및 학습은 정규 과목 및 다른 교과 과정에서 가르치는 내용과 통합될 필요가 있으며 설명을 돕는 배경을 제시하고 학생 사이에서 개인적 관련성을 키워야 한다.

개인적 영역

From code to application

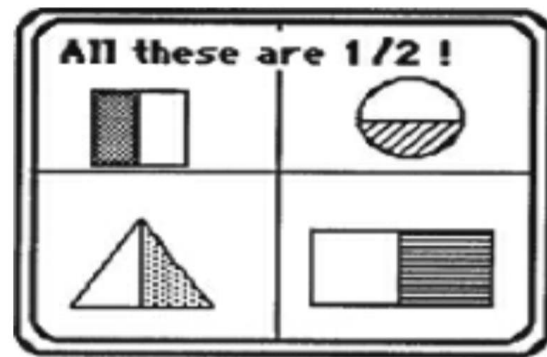
- 학생들이 학교에서 무엇을 어떤 식으로 배우고, 학교에서 배운 내용이 학교 밖에서 배우는 내용과 어떻게 연결되는가?
 - 학습의 실제성(authenticity): 학생들이 교실에서 접하는 내용이 실제 상황에서 전문가가 취하는 행동과 일치하는가이다.
 - 개인적 실제성(personal authenticity): 학생은 개인적으로 관심을 가지고 있는 것이 있으며 풍부한 지식을 쌓았음을 인정하자. 창작활동에서의 흥미
 - 실제적 청중(authentic audiences): 선생님뿐만 아니라 자신이 좋은 관계를 맺고 있는 관중을 위한 학습 결과물을 만든다.
 - 실세계 청중을 위한 설계(designing for real-world audiences): 실세계에 있는 청중을 위한 설계를 지향한다. 진정한 피드백과 어린이들에게 의미 있는 말로 적절한 평가를 지향한다.

개인적 영역

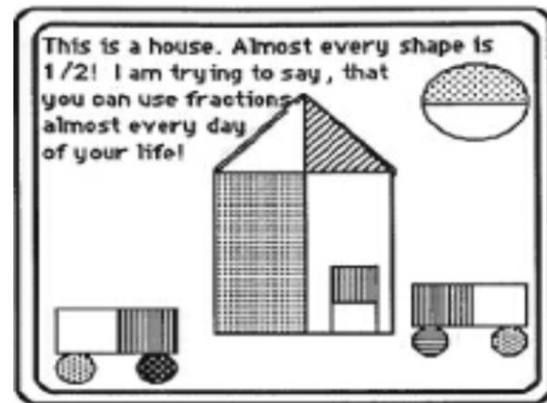
From code to application

코딩학습을 위한 소프트웨어 애플리케이션을 설계하다.

- 후배들을 위하여 분수 교육용 애플리케이션을 개발하다.
- 오른쪽의 그림처럼 분수로 만든 집으로 분수의 개념을 가르치고자 함
- (데비)는 한 종류(선생님이 가르치는 형식적인 지식)에서 다른 종류(개인적이고 구체적인 자신만의 지식)로 바꾸는 과정을 겪음
- 교실 공간에 한정된 수학의 개념(분수)을 프로그래밍을 이용하여 현실 세계의 객체와 연결(binding)하였다.
- 교수학적 변환의 의미를 확인



MARCH 23, 24



Debbie's final "House Scene"

개인적 영역

From code to application

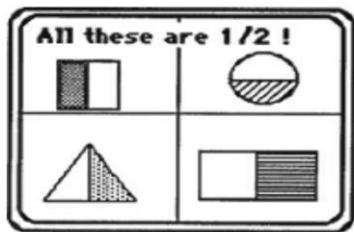
코딩학습을 위한 소프트웨어 애플리케이션을 설계하다.

- 후배들을 위하여 분수 교육용 애플리케이션을 개발하다.
- 오른쪽의 그림처럼 분수로 만든집으로 분수의 개념을 가르치고자 함
- 한 종류(formal, teacher knowledge)에서 다른 종류(personal, concrete, her own)로 바꾸는 과정을 겪음
- 교실 공간에 한정된 수학의 개념(분수)을 프로그래밍을 이용하여 현실 세계의 객체와 연결(binding)하였다.
- 교수학적 변환의 의미를 확인

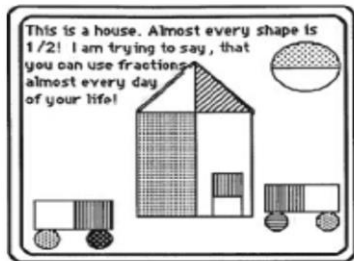


과업의 실제성(authenticity) - Debby and Naomi

Authentic audiences



MARCH 23, 24



Debbie's final "House Scene"

Personal
authenticity

House == real objects



Coding

1/2의 개념을 이미지로 표현하는 코딩(수학적 변환)

코딩에서 사회적 상호작용



Figure 1.2

Project Headlight at a Boston elementary school, 1990. Photo by L. Barry Heatherington.

- 사회적 상호 작용 - 또래간 상호작용, 온라인 커뮤니티에서의 상호작용(창의적, 공동작업), 연결된 커뮤니티를 통하여 학습
- 도제학습(apprenticeship)의 방법과 효과가 달라짐
 - 경험이 풍부한 구성원이 있는 팀에서는 계획 수립, 다른 사람을 돕거나 가르치고, 어린 학생에 대한 불안과 걱정을 이해
 - 컴퓨터를 이용한 참여와 관계 맺기
- 학생들을 교육용 소프트웨어 디자이너로 대우할 때, 효과가 더욱 높아짐
- 교사의 교수학적 지식(PCK + Coding) 필요

게임을 설계하는 권한을 학생에게 부여



- 최근에는 학생들이 마인크래프트, 코두 등과 같은 소프트웨어로 직접 게임을 개발할 수 있는 환경이 제공
- 학생들에게 가장 인기있는 게임 - 마인크래프트
- 켄 버크 - 스크래치 작가 워크숍에서 “프로그래밍을 가르치기 위해 스토리텔링을 도입해 성공을 거둔 사례를 발표함” - 국어교과와 연계([Happy mother's day](#))
- 자신만의 디지털 스토리텔링을 하도록 권장
- Digital storytelling + scratch = artifacts
- Storytelling + programming + subjects = authentic problems
- 지식과 기능을 가르치고 프로젝트로 진행하는 것이 아니라 반대 방향으로 수행

computer clubhouse



- 우리나라의 경우, 지역사회에 학생들이 쉽게 코딩을 접할 수 있는 센터가 많지 않다. 현재, SW미래채움센터가 운영되고 있으며, 사설 학원에서 운영하는 코딩 교실이 있다. 학교밖 코딩 교실이 매우 부족한 실정이다. 지역 아동센터나 작은 도서관 등에 코딩을 경험할 수 있는 설비가 제공되어야 한다.
- 컴퓨터 클럽하우스에는 무료로 운영하며, 학생들이 자유롭게 코딩할 수 있어야 하며, 다양한 연령의 사람들과 관계를 맺을 수 있는 공간이어야 한다.
- 소프트웨어를 도구로 사용하기 보다, 학생들이 설계자, 개발자로서의 역할을 할 수 있도록 메이커 경험을 제공하여야 한다.
- 지역사회의 중학생, 고등학생, 대학생 등이 함께 어우러져 코딩을 경험할 수 있어야 한다.
- 학생들이 스크래치를 기반으로 게임을 개발할 수 있는 경험을 제공할 수 있을 것이다.
- 학생들은 자신이 원하는 게임을 만드는 과정을 통하여 필요한 지식과 기술을 익히게 된다.

computer clubhouse(DIY 활동)

- 게임의 개발 과정 - personal binding
- 사회적 관계 형성 - Social connection
- 집단적 연결/온라인 연결 - Networked binding



디지털 문화에의 참여

computer clubhouse의 활동



- Digital Storytelling
- Make Games
- Draw Idea
- Share with everything
- Remix to others artifacts
- Construction through Social Interaction
- Social context, culture, Community
- Application-driven programming with scratch

개인적 영역

From code to application

- **Papert**는 지식 구성이란
 - 이미 존재하는 정신적 객체 사이에 연결 고리를 만드는 과정으로 구성된 의도적인 학습 행위
 - 로고로 프로그램을 작성하는 동안 학생들은 지식을 구성한다.
- 학습자가 프로그래밍 과정을 통하여 스스로 지식을 생산
- 자신이 만든 지식에 대해 개인적으로 동질감을 느끼는 과정
- 지적인 경험과 더불어 감성적 가치가 덧붙이는 과정
- **Object-to-think-with**(생각 유발하는 객체)
 - 프로그램, 로봇, 게임 등은 물리적인 세계와 디지털 세계에 존재하면서 머릿속에서 오래된 지식과 새로운 지식을 잇는 연결고리를 구성하고, 검토하고 수정하는데 필요한 것
 - 학습자는 프로그래밍 하는 객체와 자신을 동일시한다.
 - 프로그램이나 게임을 설계하고 개인적인 지식을 다른 사람에게 공개하고 공유한다.
 - 인터넷 문화는 개인의 생각이 함축된 객체(**objects**)를 공개하고 공유한다.



From tools to Communities

사회적 요인

From tools to communities

- 학습은 공동체 안에서 일어나는 사회적인 과정
- 개인의 성과도 중요하지만 학습의 사회적 과정에 관심이 필요
- 공동작업 행위는 지식 창출의 과정에서 학습자의 역할을 요구
- 다른 사람들과의 공유할 수 있는 결과물을 제시
- [Object-to-share-with](#)(intel Clubhouse)
- [내용 요약](#)

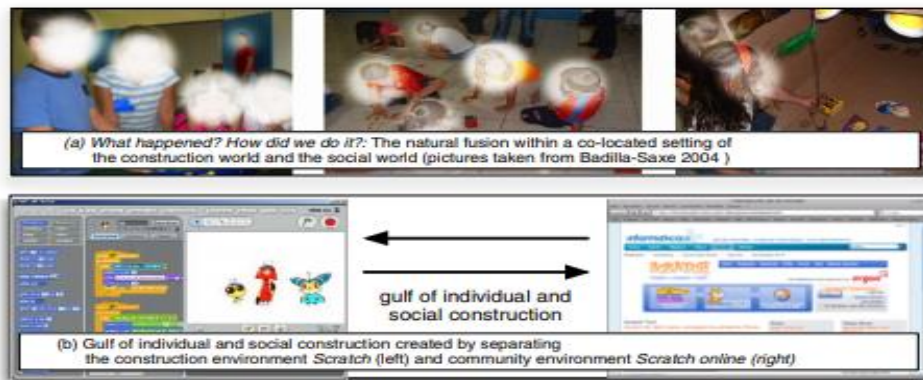
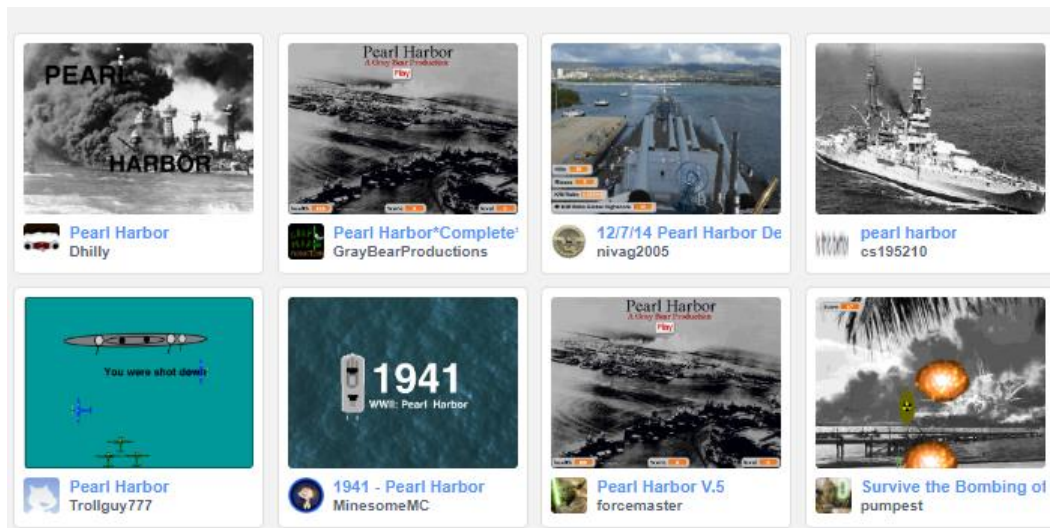


Fig. 1. Social learning in (a) co-located learning environments and (b) computer mediated environments

스크래치의 등장과 커뮤니티의 성장



Green Bear
Production
“Pearl Harbor” 개발
8살, 13살, 15살

브라질 삼바 학교와 프로그래밍

From Samba Schools to Computer Clubhouses: Cultural Institutions as Learning Environments

José Pablo Zagal and Amy S. Bruckman



Software Development as Samba School

Teaching Case

Benny Bornfeld

브라질 삼바 스쿨과 컴퓨터 문화



브라질 삼바 스쿨과 컴퓨터 문화



Computers and Computer Cultures

Seymour Papert



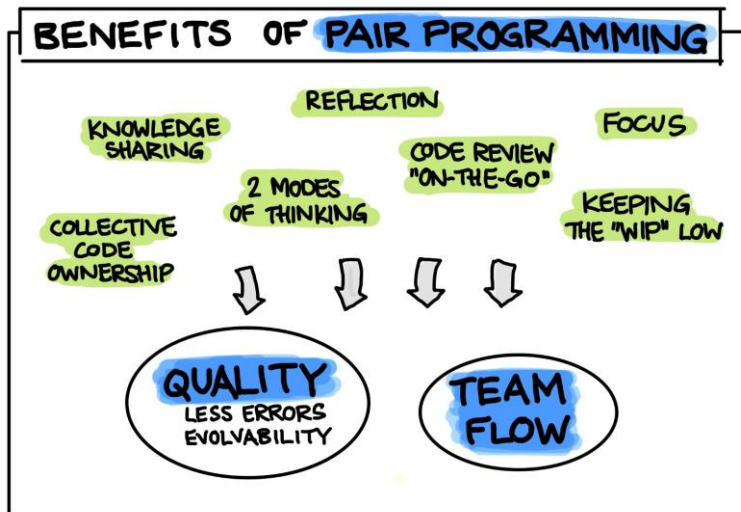
This material is a condensation of Chapter 1 of Mindstorms by Seymour Papert, published in 1980 by Basic Books, Inc., of New York City at \$12.95. Those who find themselves intimidated by this presentation will find it preferable to read the entire work.

builders, children appropriate to their own use materials they find about them, most saliently the models and metaphors suggested by the surrounding culture.

Papert writes about the order in which

construction of certain components of numerical and logical thinking. Children learn to count; they learn that the result of counting is independent of order and special arrangement; they extend the "conservation" to thinking about the properties of liquids as they are poured and of solids which change their shape. Children develop these components in

The diagram illustrates the transition from a traditional team structure to a self-driving car analogy. On the left, two black silhouettes of people are shown interacting with a large screen. Above the screen, a box labeled 'ONE MACHINE' is connected by a dashed line to the screen. To the left of the screen, a box labeled 'NAVIGATOR' is connected by a dashed line to the screen. To the right of the screen, a box labeled 'EFFICIENT TEAM' is connected by a dashed line to the screen. On the right side of the diagram, a blue car is shown inside a circular frame. Above the car, two boxes labeled 'DRIVER' and 'NAVIGATOR' are connected by dashed lines to the car. Below the car, a box labeled 'ONE MACHINE' is connected by a dashed line to the car. A blue arrow points from the 'EFFICIENT TEAM' box to the car, indicating the transition from a traditional team structure to a self-driving car analogy.



온라인 커뮤니티의 구성원리



- 개인이 창의적으로 표현하고 적극적으로 참여할 수 있는 기회를 극대화하라.
- 평범한 사람이 일반적으로 가정하는 수준보다 더 영리하고 창의적이라고 가정하라.
- 콘텐츠 제작자가 되도록 사용자에게 힘을 불어넣는 동시에 커뮤니티 내부에서 최소한의 기준을 도입하고 콘텐츠의 질을 유지하라.
- 커뮤니티가 학습을 지원하는데 필요한 기반을 구축하자.

스크래치의 커뮤니티 접근 원리



- 페퍼트의 구성주의적 접근
“초보자와 숙련자가 함께 모여 스크래치로 만든 문학작품을 읽고, 쓰고, 리믹스하는 일종의 ‘삼바 학교’가 되기를 바란다.”
- Yasmine와의 접근
“좋은 도구보다 노는 도구”
“단지 좋은 도구가 되는 쪽에서 또래들이 모여 창작하고, 공유하고, 리믹스하면서 그저 ‘노는’ 공간이 되는 쪽으로 구성”
- 또래들이 공통된 관심사를 바탕으로 결과물을 만드는 커뮤니티가 되는 쪽으로 스크래치를 만들고자 하였음
- 사람들에게 관중과 잠재적인 동료에게 다가갈 기회를 주는 동시에 제작자를 자극하고 제작자가 배우거나 리믹스할 수 있는 작품의 저장 공간을 제공한다.



From scratch to remix

문화적 요인

From Scratch to Remix



Remix

○ the process of creating something new from something old.

- 과거에는

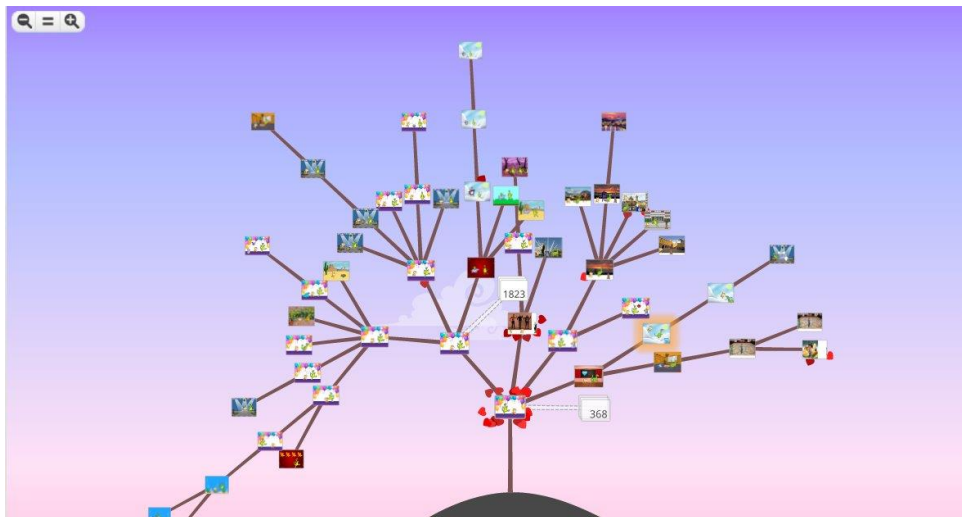
- 자신의 프로그래밍 실력을 드러내기 위하여 아무것도 없는 화면에서 코드를 작성
- 다른 사람과 공유하지 않고 저작권을 가짐
- 출력보다는 입력이라는 관점에서 프로그래밍의 가능성을 도입하는 접근 방식을 취함
- 무엇을 만드는 것이 아니라 무엇인가를 코딩하는 것에 초점
- 백지에서 유를 창조하는 것이 프로그래머의 역할이라고 생각

- 미래에는

- 스크래치(Scratch)는 힙합 DJ + 레코드판 1, 2, 3, ... → 새로운 음악을 탄생
- 기존 코드를 수정하여 새로운 것을 만듦
- 온라인에서의 '창작'은 리믹스이다.
- 스크래치 사이트는 학생들의 리믹스 저장소
- 프로그래밍의 입력보다 출력을 우선하여 생각하는 관점

왜 리믹스인가?

- 스크래치 사이트에 등록된 프로그램의 **1/4**이 리믹스한 결과물
- 스크래치에서 리믹스 하는 방법을 찾아 보기
- 리믹스는 오픈 소스 운동에 동의하고 있음
 - 저작자 표시 및 동일 조건 변경 허락
- 스크래치 리믹스 트리



문화적 요인

From Scratch to Remix

- 코딩 학습과정은 학습에 대한 문화적 가치평가의 과정을 거침(Politic)
- 브리콜뢰르(Bricoleur)
 - 한정된 자원과 도구로 새로운 것을 만드는 임시변통에 능한 사람
- 학교에서의 공식적인 학습 방법보다 브리콜뢰르가 더욱 문제를 해결하는데 훌륭한 접근법이 될 수도 있음
- 학습은 비판적인 사고활동과 물리적인 제작 행위에서 이루어짐
 - 비판적인 사고활동: 언어에 바탕을 둔 개념적인 활동
 - 물리적인 제작 행위: 구체적인 목표를 기반으로 물리적인 재료를 이용하는 작업
- 게임을 만들어라.



Remix

Remix 활동 유형



- 시간 보내기(**Hanging out**) : 사회화로써의 리믹스
- 장난치기(**Messing Around**) : 경로로서의 리믹스
- 괴짜되기(**Geeking Out**) : 창의적인 과제로 활용되는 리믹스

Hanging Out : 사회화로써 리믹스



- 목표는 다른 사람을 프로젝트에 초대하여 사회적 경험과 연계할 수 있는 조력하는 활동을 공유함
- 기존 코드를 바꾸는 대신 기존 코드에 다른 코드를 덧붙여 작성하는 경우가 많음
- 창의적인 새로운 것을 만들기 위하여 기존의 것을 모두 변경했을 때, 원 소유자와의 교류가 가능한가? 공동작업을 하는 상황에서 타인의 입장을 배려하는 것이 필요
- 스크래치 사이트에서 참여를 요청하는 프로젝트를 찾아봅시다.
 - 색칠하기 프로젝트

Messing Around : 경로로써의 리믹스

- 단순한 화면을 추가하는 것이 아니라 학습 목표에 합당한 활동으로 구성
 - “자신의 모습을 추가하세요” X
- 리믹스가 가장 크게 기여하는 점은
 - 교환 수단으로서 새로운 사용자에게 프로그래밍을 학습하는 발판을 제공
 - 복잡한 게임과 스토리, 예술 프로젝트를 제작할 능력 획득



Figure 5.3

Three “messing around” remixes: “How to Scroll,” “How to Use Variables (without Variables),” and “How to Make a Platform Game.”

Geeking Out : 창의적 경험으로써의 리믹스

- 어렵지만 재미(Hard-fun)를 추구
- 룸바 리믹스 - 타이머가 종료되기 전에 방을 효율적으로 청소할 것
- 프로그래머의 문제 해결 경험을 느낄 수 있는 기회를 제공
- 다양한 수정 조건을 제시하고 해결하는 경험을 제공

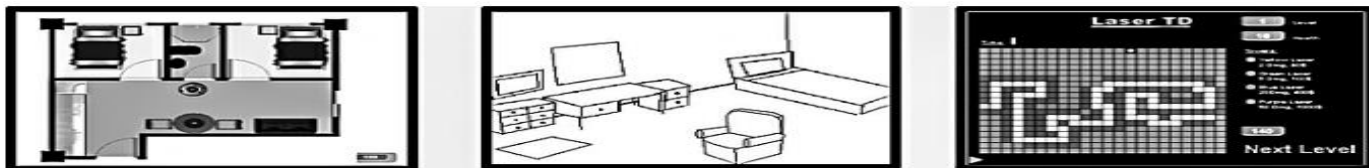


Figure 5.4

Three “geeking out” remixes: “The Roomba Remix Challenge,” “Bedroom Maker,” and “Laser Tower Defense.”

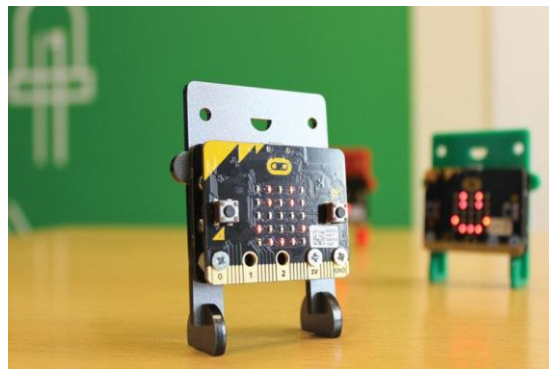


From screen to tangibles

실체적 요인

From Screen to Tangibles

- 스크래치의 고양이
 - 화면속 결과물은 추상적인 생각을 구체적으로 변환하는 능력
 - 추상적인 생각을 눈으로 보고 손으로 만질 수 있는 느낌을 제공
- 내부 탐험용 컴퓨터의 사례를 통해 알 수 있는 경험
- 로봇이나 키트 등을 통하여 학습하는 경험을 제공
- 요약내용



CT와 개념과 역량의 관계

Computational Thinking	Programming Concepts	21세기 역량
문제 정의(Problem Identification) 문제 분해(Problem Decomposition) 추상화(Abstraction) 논리적 사고(Logical Thinking) 패턴 인식(Patterns Recognition) 알고리즘(Algorithms) 디버깅(Debugging)	알고리즘 순차 입출력 처리 산술, 관계, 논리 처리 변수와 상수 조건문과 반복문 절차적 병행(Parallelism, events) 동기화(Synchronism) 검증 및 디버깅	비판적 사고력 창의성 협동 의사소통 정보 문해 미디어 문해 기술 문해 유연성 리더십 동기 및 의지 생산성 사회적 기술

프로젝트 과제

- 소프트웨어와 함께하는 창의력 여행([학생](#), [교사](#))
- PBL을 적용한 교수-학습
- 인지적 도제 이론을 적용한 학습 전략
- pair programming
- (과제)
 - 장애물 피하기 게임 만들기를 이용하여 리믹스하기
 - 코딩을 위한 구성주의 접근법의 적용 전략을 설명하시오.
 - 인지적 도제 학습 전략을 적용 방법을 설명하시오.