

# 강의계획서

1. 교과목명 : 컴퓨팅사고와 문제해결

2. 학점 : 3학점

3. 대상 : 3학년 1학기

4. 강의 목적

초등학교에서부터 소프트웨어교육(코딩교육)이 정규 교육과정에 포함됨에 따라 이를 지도할 수 있는 역량을 갖춘 교사를 양성하는 것이 이 강좌의 목적이다. 소프트웨어 교육의 배경이 되는 컴퓨터과학에 대한 이해를 바탕으로 교육과정의 이해를 공고히 하고 컴퓨터 과학을 통하여 컴퓨팅 사고(Computational Thinking)를 활용한 문제해결능력을 지도할 수 있는 역량을 강화한다.

5. 강의 개요

본 강의는 최근 많은 관심을 받고 있는 컴퓨팅 사고(computational thinking) 역량을 향상시키기 위한 내용으로 구성하였다. 이를 위하여 컴퓨터 과학의 기본 개념을 이해하고 일상의 공간에서 컴퓨팅 관련 개념을 탐구하고자 한다. 우리의 일상 생활에서 컴퓨터 과학이 접목된 사례를 찾아 관련 개념을 탐색하고 문제를 해결하는 경험을 공유한다.

6. 강의 학습 목표

- 학습 목표

- 컴퓨팅사고(computational thinking)력을 설명할 수 있다.
- 컴퓨터과학의 기본 개념을 상황에 맞게 적용할 수 있다.
- 컴퓨터 과학의 개념이 적용된 사례를 찾아 발표할 수 있다.
- 문제를 해결하기 위하여 다양한 도구가 활용됨을 이해하고 적극 활용할 수 있다.
- 초등학교에서의 소프트웨어 교육 정책의 변천과정을 설명할 수 있다.
- 초등학교 소프트웨어 교육 과정을 이해하고 적용할 수 있다.
- 구성주의적 소프트웨어 교육 방법을 탐구하고 실천할 수 있다.
- 학습자 중심의 소프트웨어 교육의 평가 방법을 알고 실행할 수 있다.

7. 컴퓨터과학교육 내용 요소(PCK)

- 교과 내용학

- 컴퓨터 과학의 기본 개념을 이해
- 컴퓨터과학에서 문제해결을 위한 주요 개념의 이해
- 컴퓨터과학의 개념의 적용 사례를 이해
- 컴퓨터 과학의 발달 과정을 이해
- 컴퓨팅 사고의 개념을 이해하고 이를 실천할 수 있다.

교과교육학

- 컴퓨터과학의 주요 개념과 소프트웨어 교육의 관련성을 설명할 수 있다.
- 컴퓨터과학 지식을 지도하기 위한 방법론을 이해할 수 있다.

테크놀로지의 적용

- physical computing의 사례를 경험하고 문제를 해결할 수 있다.

- 문제 해결을 위하여 다양한 학습 도구를 적용할 수 있다.
- 

## 8. 강의 운영 방법

매주 강의 및 실습으로 강의를 진행한다. 실습을 위해서는 구글 클라우드와 스크래치와 같은 프로그래밍 도구를 활용한다. 강의 자료는 구글 클라우드를 통하여 제공하고 상호 작용한다. 구체적인 내용은 다음과 같다.

- 강의
  - 매주 해당 주제의 핵심 내용에 대한 이해를 돕기 위해 강의를 이루어진다.
  - 컴퓨터과학의 주요 개념을 이해하고 이를 일상의 공간에서 탐색한다.
  - 학생들은 다양한 정보 기기를 활용하여 수업에 참여할 수 있다.

### 실습

- 강의는 테크놀로지를 기반으로 운영되므로 구글 클라우드를 융통성 있게 활용한다. 모든 자료는 구글 클라우드를 통하여 학생들에게 제공되고 수업 관련된 학생 활동은 대부분 클라우드에서 상호 작용하도록 구성한다. 컴퓨터 과학의 개념을 적용할 수 있는 사례를 경험하도록 구성한다.

- 과제
  - 각 차시 별로 강의와 실습이 진행되며 그 과정은 구글 클라우드를 통하여 포트폴리오를 생성한다.. 그 외 구체적인 과제 계획은 다음과 같다.

과제 번호	해당 주차	과제 명	형태	과제 내용
과정평가	매주	포트폴리오 및 성찰일기	개별	각 주 차별 강의 내용에 대한 포트폴리오 및 성찰일지 작성
과제2	과제	프로젝트	개별	컴퓨터과학의 개념을 적용할 수 있는 사례를 프로그래밍도구를 활용하여 개발한다.
기말평가	평가	총괄평가	개별	본 과목에서 학습한 내용에 대한 총괄평가를 실시하여 학습 목표 달성 정도를 파악한다.

## 9. 강의 교재 및 자료

- computational thinking 연구 보고서 및 자체 개발한 ppt 등 교재, 각종 멀티미디어 및 인터넷 자료를 활용한다. 구체적인 강의 교재 및 자료는 다음과 같다.
- 서적
  - David D.Riley, Kenny a. Hunt.(2014). computational thinking. CRC Press Taylor & Francis Group.
  - James bird, Helen Caldwell, Peter Mayne, Adam Scribbans, Sway Grantham, alison Witts and Mark dorling(2014). Lessons in Teaching computing in primary schools. SAGE.
  - David D. Riley and Kenny A. Hunt. computatational Thinking for the modern problem slover.
  - 교육부(2015), 초등학교 교육과정 총론. 교육부

- 정영식, 유정수, 임진숙, 홍지연(2019). 소프트웨어 교육론. 씨마스.

- 자체 개발한 교재 및 PPT
  - 소프트웨어 교육과 관련된 저널이나 논문 토대로 개발한 ppt 자료
- 각종 멀티미디어 및 인터넷 자료
  - 소프트웨어 교육과 관련한 신문기사. 정책 자료 등의 활용
  - 국내에서의 소프트웨어 교육 사례 등의 활용

#### 10. 평가 방법

- 포트폴리오(20%), 시험(20%), 프로젝트(40%), 출석(10%), 태도(10%) 등에 따라 총점을 산출하며, 절대평가한다. 각 항목에 대한 구체적인 내용은 다음과 같다.
- 시험
  - 시험은 기말고사 한 번만 실시한다. 문제는 모두 주관식으로 출제한다.
  - 포트폴리오는 매주 또는 필요시 작성하며 수업 및 수업에서의 활동 등을 기록한다.
  - 학습자 중심의 활동이 있는 경우, 개인별 성찰일지를 작성한다.

#### 과제

- 강의 운영 방법에 제시된 과제 수행 계획에 근거하여 평가한다.
- 프로젝트마다 가중치를 부여하고 성취도를 5단계로 평가한다.
- 태도
  - 수업에 참여하는 성실성에 대한 평가와 강좌 운영에 도움을 주거나 학습 분위기 형성에 좋은 태도를 보인 경우 추가적으로 부과되는 평가이다.

## 11. 강좌의 개요

주	강의주제	강의 내용	강의 방법 및 교재	과제
1	오리엔테이션	강좌 소개 강의의 목표와 학습과정 안내 1주차 강의 안내	강의 강의 계획서 프로젝트 안내	
2	컴퓨터 과학의 개요	컴퓨터과학의 이해 컴퓨터의 개요 Computer Educaton in K-12	강의, 토론 자체 개발 교재	포트폴리오
3	2015개정 교육과정의 SW교육	소프트웨어 교육의 현황 및 개선과제(국회) AI 교육정책	강의	
4	Computational Thinking의 이해	Coding and Computational Thinking 스크래치에서 적용되는 컴퓨팅 개념	소프트웨어 도구 체험하기	308 호
5	컴퓨팅 사고력 탐구하기	CT and Coding for Every Student Code.org를 활용한 코딩 교육	Hour of code 프로젝트 수행하기	포트폴리오
6	AI, Edtech	학교에서의 AI, SW교육을 위한 Edutech	강의, 토론	포트폴리오
7	중간고사			
8	로봇과 컴퓨터광학	비봇으로 코딩하기 비봇으로 할 수 있는 컴퓨터과학 지식	강의, 문제해결	포트폴리오
9	구성주의 관점의 교수학습 방법	SW교육의 구성주의적 접근 방법	강의, 토론	성찰일기
10	교수-학습전략1	코딩을 위한 교수학습 전략, 접근법(1), 접근법(2)	강의, 토론	성찰일기
11	교수-학습전략2	코딩을 위한 교수학습 전략 (Remix) 스크래치와 교과와의 연계 활동	PBL 리믹스하기	성찰일기
12	소프트웨어 교육 교재 개발하기	프로젝트 주제 설정	프로젝트 사례 제시	포트폴리오
13		프로젝트 진행	청소로봇 사례	포트폴리오
14		프로젝트 발표	프로젝트	포트폴리오
15	기말고사			