

군산금빛초 개념기반 탐구수업 모델 단위 설계안

1 초학문적 주제 중심 개념기반 탐구 단위 설계

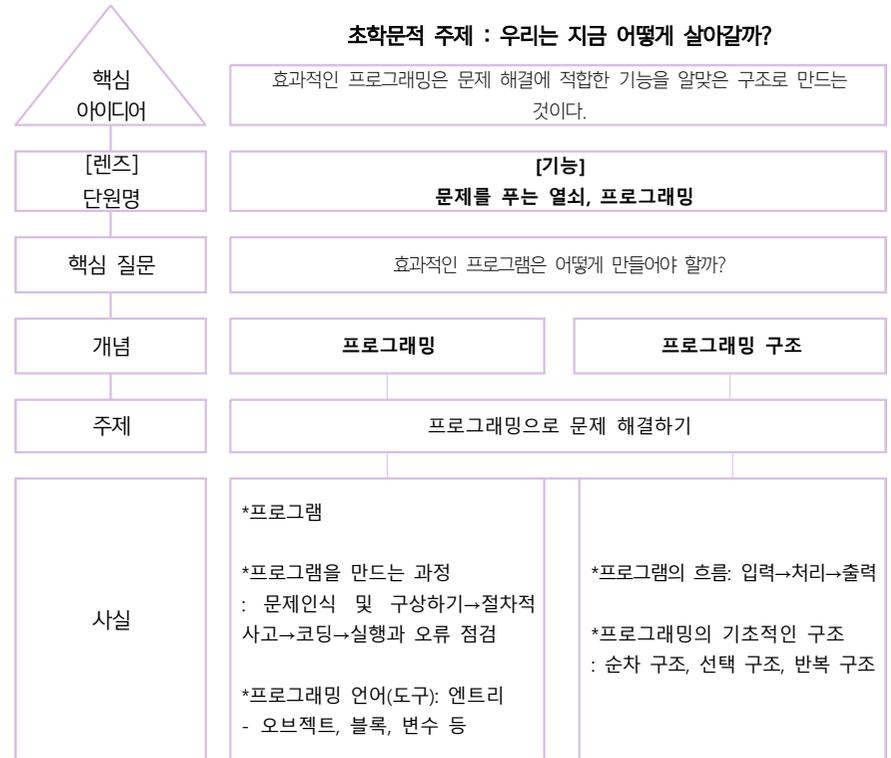
초학문적 주제	우리는 지금 어떻게 살아갈까?				
단원명	문제를 푸는 열쇠, 프로그래밍				
학년	6	관련 교과	실과	차시	12
핵심 아이디어	프로그래밍은 디지털 사회에서 발생하는 다양한 문제를 해결하는 데 도움을 준다.				
단원 핵심아이디어	효과적인 프로그래밍은 문제 해결에 적합한 기능을 알맞은 구조로 만드는 것이다.				
개념 렌즈	기능	관련 개념	프로그래밍, 프로그래밍 구조		
핵심 질문	효과적인 프로그램은 어떻게 만들어야 할까?				
성취 기준	<p>[6실04-09] 프로그래밍 도구를 사용하여 기초적인 프로그래밍 과정을 체험한다.</p> <p>[6실04-10] 자료를 입력하고 필요한 처리를 수행한 후 결과를 출력하는 단순한 프로그램을 설계한다.</p> <p>[6실04-11] 문제를 해결하는 프로그램을 만드는 과정에서 순차, 선택, 반복 등의 구조를 이해한다.</p>				
단원 설계 의도	<p>‘소프트웨어와 생활’ 단원은 ‘①소프트웨어의 이해 ②절차적 문제 해결 ③프로그래밍의 요소와 구조’ 세 개의 소단원으로 구성되었습니다. ‘소프트웨어와 절차적 사고’ 개념을 학습한 후 이어지는 세 번째 단원을 개념기반 탐구수업 모델을 적용해 설계했습니다. 이 단원은 ‘기능’ 개념 렌즈를 통해, 프로그래밍이 문제 해결을 위한 도구임을 이해하고, 목적에 맞는 구조와 기능을 갖춘 프로그램을 설계하는 역량을 기를 수 있도록 구성되었습니다.</p> <p>학습자는 프로그래밍 도구를 직접 체험하면서, 각 구성 요소가 어떤 기능을 수행하며 어떤 형태로 사용되는지를 탐색합니다. 이를 통해 순차, 선택, 반복과 같은 기본 구조를 이해하고, 자료의 입력-처리-출력 과정을 따라 간단한 프로그램을 설계하고 실행해 볼 수 있습니다. 안내된 프로그래밍 과정을 단순히 따라하는 것이 아니라, 가능한 최소한의 안내 안에서 스스로 기능에 적합한 구조를 찾아 알맞은 순서로 블록을 설계하고 오류를 수정하는 프로그래밍 과정을 학습할 수 있도록 단원을 설계했습니다.</p>				

2 초학문적 주제 중심 교육과정 내용 분석

	지식·이해	과정·기능	가치·태도
내용 체계표	<ul style="list-style-type: none"> 프로그래밍의 필요성과 의미 프로그래밍 언어 구성 요소 프로그래밍의 구조 	<ul style="list-style-type: none"> 프로그래밍 언어 구성 요소 탐색하기 기초적인 프로그램 설계하기 프로그래밍 오류 수정하기 	<ul style="list-style-type: none"> 프로그래밍 요소와 구조를 적극적으로 탐색해 보려는 자세 오류나 실패에 유연하게 대처하려는 자세 프로그래밍을 통해 만든 산출물을 타인과 공유하고 협력하려는 자세

일반화	탐구질문 / 핵심질문
효과적인 프로그래밍은 문제 해결에 적합한 기능을 알맞은 구조로 만드는 것이다.	<ul style="list-style-type: none"> 프로그래밍은 무엇일까? 프로그래밍 언어는 어떻게 구성되어 있을까? 프로그램은 어떤 흐름으로 작동할까? 프로그램의 기능을 효과적으로 실행하려면 어떤 구조로 조직해야 할까? 효과적인 프로그램은 어떻게 만들어야 할까?

3 단원의 조망도



4 단원 평가

수행과제	생활 속 문제를 해결하는 프로그램을 설계하고, 구현하여 소개하세요.			
과제 설명	목표	일상생활 속에서 반복되거나 귀찮은 문제를 하나 선택하고, 그 문제를 해결할 수 있는 프로그램을 설계 및 구현한 뒤, 자신이 선택한 구조와 기능 흐름을 설명한다.		
	역할	일상생활 문제를 해결하려는 학생 소프트웨어 개발자이다.		
	청중	친구들(동료 개발자 및 사용자), 교사		
	상황	학교 또는 가정에서 반복되는 불편함(예: 시간 놓치기 등)을 해결하는 간단한 엔트리 기반 프로그램이 필요하며, 내가 개발한 프로그램을 청중들에게 소개해야 한다. <예시 주제> - 수업 시작 중 치고 나서 수업 준비를 함: 쉬는 시간 종료 1분 전 안내 프로그램 - 학생 번호와 이름이 헷갈림: 우리 반 번호를 입력하면 이름을 알려주는 프로그램 - 맨손체조를 하는데 박자를 놓침: 4박자로 리듬을 알려주는 프로그램 - 바닥을 깨끗이 지우고 싶음: 로봇 청소기 프로그램		
	결과물	다음 3가지 결과물을 제출해야 한다. ① 문제와 구조를 설명하는 설계안(순서도 또는 흐름도) ② 엔트리 프로그램 구현 파일 또는 실행 시연 ③ 발표 자료 (내가 선택한 구조, 구현 과정, 받은 피드백, 성장한 점 등)		
준거	1. 문제와 구조의 적절성 2. 구현의 정확성 및 개선 반영 3. 공유와 협력의 태도			
기준	잘함	보통	노력요함	
평가요소	문제 해결을 위한 구조 선택 (지식-이해)	문제 상황을 정확히 분석하고, 가장 적절한 구조(순차/선택/반복)를 기능 중심으로 명확하게 설명함	문제와 구조의 연결이 타당하나 기능적 설명이 다소 부족함	구조를 선택하였으나 문제와의 관련성 또는 설명이 미흡함
	프로그래밍 구현과 수정 (과정-기능)	엔트리로 구조를 정확히 구현하고, 피드백을 바탕으로 기능적 개선까지 이루어짐	대부분의 구조를 구현하였으며 간단한 수정-보완이 이루어짐	구현은 일부 이루어졌으나 기능 오류 수정이나 개선이 부족함
	공유와 협력의 태도 (가치-태도)	자신의 산출물을 적극적으로 설명하고, 타인의 의견을 수용하여 협력적 개선을 시도함. 자발적으로 친구에게 조언하거나 응원의 표현을 함	자신의 산출물을 설명하며 친구의 피드백을 이해하고 반응함. 일부 협력적인 태도가 보이지만 적극성은 부족함	자신의 산출물을 설명하는 데에 소극적이고 피드백 수용 및 협력적인 태도가 부족함

5 군산금빛초 개념기반 탐구수업 모델 단계별 수업 과정

탐구 흐름	발견하기 (Discover)	• 프로그래밍의 필요성과 의미 이해하기
	↓	
	탐구하기 (Explore)	• 프로그래밍 언어(엔트리)의 구성 요소 탐색하기 • 프로그램의 기능 수행을 위한 작동 순서 탐구하기(입력→처리→출력) • 프로그래밍의 구조인 순차, 선택, 반복 구조를 직접 체험하고, 쓰임을 이해하기
	↓	
	연결하기 (Connect)	• 내가 해결하고 싶은 문제를 선택하고 적절한 프로그래밍 구조 결정 및 흐름도 작성 • 효과적인 프로그래밍의 특성 일반화 하기
↓		
실천하기 (Act)	• 엔트리로 직접 프로그래밍하기 • 엔트리로 프로그램 완성 후 오류 수정 및 기능 추가하기	
↓		
성찰하기 (Reflect)	• 나의 프로그래밍 구조 및 기능 발표, 친구들 작품 감상 및 피드백	

단계	주도성 요소 일반화	개념기반 탐구수업의 흐름	자료(★) 수업팁
발견하기	C	<1차시> [사실] 프로그래밍은 무엇일까? <도입> ○ 생활 속 사례 제시(자판기, 자동차, 교통신호등, 자율주행 자동차) - 컴퓨터나 기계가 스스로 일을 할 수 있는 이유는 무엇일까? <발견> [Think-Puzzle-Explore] ○ Think: 프로그래밍에 대해 알고 있는 것이 있나요? - 브레인스토밍 후 하나의 개념으로 정리하기 - 프로그래밍은 (컴퓨터로 문제를 해결하는 도구, 컴퓨터로 문제를 해결하기 위해 프로그램을 만드는 과정)이다. ○ Puzzle: 프로그래밍에서 아직 잘 모르겠는 점이나, 더 알고 싶은 점은 무엇인가? - 앞으로의 탐구 질문과 연결함 ○ Explore: 프로그래밍에 대해 더 알기 위해 우리는 무엇을 할 수 있을까? <정리> ○ 프로그래밍이 사용된 사례 더 찾아보기 - 생활 속에서 본 '프로그래밍이 사용된 기계'를 하나 찾아, 프로그램에 있는 기능과 명령어 생각하기	★이미지 (생활 속 사례) ★활동지 (땀커벌보드)
		<2-3차시> [사실] 프로그래밍 언어는 어떻게 구성되어 있을까? <도입> ○ 프로그래밍 언어의 필요성 및 의미 이해하기 - 컴퓨터에게 명령을 하려면 컴퓨터가 이해할 수 언어가 필요함을 인식하기 - 프로그래밍과 관련된 학습한 개념 은행 만들기	

 	<p>[Know-Want to know-Learned]</p> <p>○ Know: 엔트리에 대해 무엇을 알고 있나요? - 엔트리가 프로그래밍 언어의 한 종류임을 인지하기</p> <p><탐구></p> <p>○ Want to know: 프로그래밍 언어(엔트리)에 대해 어떤 것이 궁금한가요? ○ 엔트리 탐색 미션: 엔트리 구성 요소 찾기 - 엔트리 화면 구성 살펴보기 - 블록 종류 구분하기 - 블록 실행해 보기</p> <p>○ 기초적인 프로그램 만들기 - 자기를 소개하는 프로그램을 만들어 봅시다.</p> <p><정리></p> <p>○ Learned: 엔트리에 대해 더 알게 된 점은 무엇인가요? - K-W-L 활동지 내용 공유 및 발표하기</p> <p><4차시></p>	<p>★개인활동지 (패들릿)</p> <p>★ 활동지 (엔트리 구성 요소 탐험 노트)</p>
	<p>[사실] 프로그램은 어떤 흐름으로 작동할까?</p> <p><도입></p> <p>○ 생활 속 예시 제시: 디지털 도어락으로 잠긴 문을 열 때 흐름 찾기 - 비밀번호(버튼)를 누른다. - 입력한 번호가 설정한 번호랑 같은지 확인한다. - 잠금장치가 풀린다.</p> <p><탐구></p> <p>[Flow Mapping]</p> <p>○ 생활 속 문제(디지털 도어락)를 3단계로 나누어 보고, 각 단계를 나타내는 말을 개념어로 정리해 봅시다. - 비밀번호를 누른다.→'입력: 프로그램이 정보를 받는 단계' - 저장된 비밀번호와 맞는지 비교한다.→'처리: 받은 정보를 이해하고, 계산 및 판단하는 단계' - 비밀번호가 맞으면 문을 연다.→'출력: 처리한 결과를 화면, 소리, 동작 등으로 보여주는 단계'</p> <p>○ 간단한 프로그램(두 수를 더하는 프로그램) 예시 보며 프로그래밍의 흐름 이해하기 - '입력'→첫 번째 수와 두 번째 수를 입력한다. - '처리'→두 수를 더한다. - '출력'→더한 결과를 대답한다.</p> <p>○ '입력-처리-출력' 흐름을 시각적으로 표현하고 간단한 프로그램 스스로 만들기</p> <p><정리></p> <p>○ 개념 정리하기: 입력-처리-출력 흐름 구조 - 정보를 받고(입력)→생각하고(처리)→결과를 보여준다.(출력) - 프로그램은 입력을 받아 처리한 뒤, 그 결과를 출력하는 논리적인 흐름을 따른다.</p> <p><5차시></p>	<p>★ 자판기 이미지</p> <p>★ 패들릿 (흐름도)</p> <p>★활동지 (흐름도)</p> <p>★블록예시</p>

	<p>[개념] 프로그램의 기능을 효과적으로 실행하려면 어떤 구조로 조직해야 할까?</p> <p><도입></p> <p>○ '순차 구조' 일상 예시 질문하기 - 양치하기 순서를 바꾸면 어떻게 될까? - 일의 순서를 잘못 나열했을 경우 문제가 생기는 상황은 무엇일까요? : 옷 입기, 라면 끓이기, 청소하기 등 -기능을 정해진 순서대로 실행해야 할 때 필요한 구조는 무엇일까요?</p> <p><탐구></p> <p>○ 간단한 순차 프로그램(고양이를 만나는 프로그램) 예시 보며 순차구조 이해하기 - 고양이가 오른쪽으로 이동하기→고양이 울음소리 내기→인사말하기</p> <p>[What Makes You Say That?]</p> <p>○ 엔트리 미션: 캐릭터를 만날 때 반응 프로그램 만들기 - 소녀(년)가 말하기→캐릭터에게 이동하기→방귀소리 내기→2초 후 얼굴색 변하기</p> <p>○ 질문하기 - 왜 그 순서로 블록을 조립했나요? - 이런 결과가 나온 이유는 무엇일까요? - 처음엔 이렇게 했는데, 왜 고쳤나요? - 친구의 조립 방식과 다른 점이 뭐고, 왜 그렇게 했을까요? - 다시 조립한다면 똑같이 할 건가요? 왜 그렇게 생각하나요?</p> <p><정리></p> <p>○ 순차 구조가 필요한 기능은 무엇일까요? - 인사 및 소개하기 - 절차 안내하기(손 씻는 방법 안내) - 이야기 전달(이야기 줄거리는 순차적으로 전달) - 타이머 및 알람 기능(시간의 흐름에 따라 순서대로 동작)</p> <p><6차시></p>	<p>★ 참고자료 (흐름도,블록)</p> <p>★미션활동지</p> <p>★패들릿 (모둠별로 정리한 내용 공유)</p>
	<p><도입></p> <p>○ '선택 구조' 일상 예시 질문하기 - 횡단보도에서 어떤 조건일 때 걸어가고, 어떤 조건일 때 멈출까요? : 초록불이면 걸어가고, 빨간불이면 멈춰요. - 특정한 조건이나 상황에 따라서 행동을 다르게 하는 경우는 어떤 것들이 있을까? : 비가 오면 우산을 쓴다(안 오면 그냥 간다), 배가 고프면 밥을 먹는다(고프지 않으면 밥을 먹지 않는다). 영화관에서 영화를 예약할 때 영화가 15세 이상 이용 가라면 예약할 수 없다. - 상황에 따라 다르게 작동하려면 어떤 구조가 필요할까?</p> <p><탐구></p> <p>○ 간단한 선택 프로그램(나라 수도 맞는 프로그램) 예시 보며 선택 구조 이해하기</p> <p>○ 엔트리 미션: 정답을 맞거나 틀리면 소리가 나는 프로그램 만들기</p> <p>[What if]</p> <p>○ 추가 미션 활동 - 정답을 바꾸면? - "땡" 대신 칭찬 소리? - 정답이 두 개라면?</p>	<p>★ 횡단보도 이미지</p> <p>★ 참고자료 (흐름도,블록)</p> <p>★미션활동지</p>

	<ul style="list-style-type: none"> - 정답을 맞힌 개수를 알려주려면? <p><정리></p> <ul style="list-style-type: none"> ○ 선택 구조가 필요한 기능은 무엇일까요? - 정답 판별 - 날씨에 따른 옷 추천 - 장애물이 있으면 멈추고, 없으면 전진 - 버튼에 따라 다르게 반응 	<p>★패들릿 (모듬별로 정리한 내용 공유)</p>
	<p><7차시></p> <p><도입></p> <ul style="list-style-type: none"> ○ 일상 예시 질문 - 박수를 칠 때 '손바닥 끼리 부딪혀.'를 10번 반복해서 말하면 어떨까? 다르게 표현할 수 있는 방법은 없을까? - 반복이 필요한 상황은 언제일까요? : 박수치기, 계단 오르기, 알람 등 - 같은 기능을 반복하려면 어떤 구조가 필요할까? <p><탐구></p> <ul style="list-style-type: none"> ○ 반복문이 없는 것과 간단한 반복 프로그램(삼각형 그리기) 예시 보여 반복구조 이해하기 ○ 엔트리 미션: 꽃잎 도장 프로그램 만들기 <p><정리></p> <p>[Connect-Extend-Challenge]</p> <ul style="list-style-type: none"> ○ 반복 구조 정리하기 - Connect: 반복 구조는 다른 구조와 어떻게 조합해서 사용할 수 있을까요? - Extend: 프로그래밍에서 반복 구조가 왜 중요할까요? - Challenge: 반복 구조를 배우면서 어렵거나, 궁금한 점은 무엇인가요? ○ 반복 구조가 필요한 기능은 무엇일까요? - 반복해서 선 긋기(정삼각형, 정사각형 등 그리기) - 클릭할 때마다 도장이 찍힘 - 소리 및 리듬 패턴 만들기 	<p>★참고자료 (흐름도,블록)</p> <p>★미션활동지</p> <p>★패들릿 (모듬별로 정리한 내용 공유)</p>
연결하기	<p><8차시></p> <div style="border: 1px solid black; padding: 5px; background-color: #fff9c4; display: inline-block;">[개념] 효과적인 프로그램은 어떻게 만들어야 할까?</div> <p><도입></p> <ul style="list-style-type: none"> ○ 지금까지 만든 프로그램의 기능과 구조 연결하기 - 순서대로 자기 소개하기 프로그램: 순차 구조 - 문제를 맞히는 프로그램: 선택 구조 - 일정한 간격으로 같은 모양의 도장 찍기: 반복 구조 <p><연결> 모듬활동</p> <p>[Generate-Sort-Connect-Elaborate]</p> <ul style="list-style-type: none"> ○ 문제 해결에 적합한 기능과 구조 정리하기 - Generate: 프로그래밍으로 해결하고 싶은 일상 속 문제는?(브레인스토밍) - Sort: 문제를 분류하고 어떤 구조가 어울릴까? 이 구조를 선택한 이유는? : 정해진 순서 문제(순차 구조), 상황별 반응 문제(선택 구조), 반복되는 문제(반복 구조) 	<p>★ 활동지</p>

	<p>적합한 기능을 알맞은 구조로 만드는 것이다.</p> <ul style="list-style-type: none"> - Connect: 문제를 해결하기 위해 어떤 기능이 필요할까? - Elaborate: 입력→처리→출력 흐름 정리하기 <p><정리></p> <ul style="list-style-type: none"> ○ 일반화 문장 만들기 - 프로그램을 만들기 위해 우리는 무엇을 먼저 생각했나요? : 어떤 기능이 필요한가?, 기능에 맞는 구조는 무엇인가? - 지금까지 배운 걸 바탕으로, 효과적인 프로그램이란 어떤 조건을 갖춰야 한다고 생각하나요? 한 문장으로 써 봅시다. : 효과적인 프로그램은 ()이다. <p><9-10차시> 생활 속 문제를 해결하는 프로그램을 설계하고, 구현하기</p> <p>[Plan-Do-Reflect]</p> <p><도입></p> <ul style="list-style-type: none"> ○ 수행 과제 확인하기 - 생활 속 문제를 해결하는 프로그램을 설계하고, 구현하여 소개하기 <p><예시></p> <ul style="list-style-type: none"> - 쉬는 시간 종료 1분 전 안내 프로그램 - 우리 반 번호를 입력하면 이름을 알려주는 프로그램 - 4박자로 리듬을 알려주는 프로그램 - 로봇 청소기 프로그램 <p><실천></p> <ul style="list-style-type: none"> ○ Plan: 이 프로그램은 어떤 흐름(입력-처리-출력)으로 작동해야 하죠? 어떤 블록들이 필요할지 먼저 정리해 볼까요? ○ Do: 설계도를 보며 엔트리로 프로그램 구현하기 ○ Reflect: 생각한 대로 작동했나요? 잘 안된 부분에서 어떤 점을 바꾸었고, 결과는 어떻게 되었나요? - 내 프로그램이 문제를 잘 해결하게 되었나요? - 다음에 프로그래밍을 할 때 기억하고 싶은 점은 무엇인가요? <p><정리></p> <ul style="list-style-type: none"> ○ 발표 자료 만들기 	<p>★ 패들릿</p> <p>루브릭평가 내용참고</p> <p>★ 활동지</p> <p>★캔바</p>
실천하기	<p><11-12차시> 내가 만든 프로그램 발표 및 성찰하기</p> <p><도입></p> <ul style="list-style-type: none"> ○ 발표 자료 정리하기 <p><실천></p> <ul style="list-style-type: none"> ○ 자기 프로그램 발표 후 친구에게 피드백 받기 - 내가 해결하고자 했던 문제는? - 문제를 해결하기 위한 프로그램에는 어떤 기능이 필요했나요? - 내가 선택한 프로그래밍 구조와 그 이유는? - 내 프로그램의 입력-처리-출력 흐름, 블록 구조 정리 <p><정리></p> <p>[I used to think...Now I think...]</p> <ul style="list-style-type: none"> ○ 프로그래밍, 구조, 기능, 과정, 감정 등을 중심으로 이 단원을 공부하면서 변화한 점을 표현하기 - I Used to Think: 예전에는 이렇게 생각했어요. - Now I Think: 지금은 이렇게 생각해요.. 	<p>★발표 자료</p>

6 수업 차시 과정안

초학문적 주제	우리는 지금 어떻게 살아갈까?		
단원명	문제를 푸는 열쇠, 프로그래밍	관련 교과	실과
학년	6학년	차시	4/12
개념렌즈	기능	주도성 구성요소	자기조절 
탐구질문	프로그램은 어떤 흐름으로 작동할까?		
단계	교수·학습 활동		자료(★) 주도성 요소
동기유발 (5')	<ul style="list-style-type: none"> ◎ 생활 속 프로그램 흐름이 적용되는 예시 제시 <ul style="list-style-type: none"> • 디지털 도어락으로 잠긴 문을 열 때 어떻게 작동할까요? <ul style="list-style-type: none"> - 비밀번호(버튼)를 누른다. - 입력한 번호가 설정한 번호랑 같은지 확인한다. - 잠금장치가 풀린다. ◎ 탐구 질문 확인하기 <ul style="list-style-type: none"> • 프로그램은 어떤 흐름으로 작동할까요? 		★자판기 이미지
과제탐색 (15')	<p><Flow Mapping></p> <ul style="list-style-type: none"> ◎ 생활 속 프로그램 작동 흐름(3단계)을 개념어로 정리하기 <ul style="list-style-type: none"> • 디지털 도어락 동작 흐름을 3단계로 나누어 보고, 각 단계에 해당하는 말을 찾아 개념어로 정리하기 <ul style="list-style-type: none"> - 비밀번호를 누른다.→'입력: 프로그램이 정보를 받는 단계' - 저장된 비밀번호와 맞는지 비교한다.→'처리: 받은 정보를 이해하고, 계산 및 판단하는 단계' - 비밀번호가 맞으면 문을 연다.→'출력: 처리한 결과를 화면, 소리, 동작 등으로 보여주는 단계' ◎ 간단한 프로그램(두 수를 더하는 프로그램) 예시 보여 프로그램 작동 흐름 이해하기 <ul style="list-style-type: none"> • 입력-처리-출력 흐름으로 구분하여 시각화하기 <ul style="list-style-type: none"> - 입력: 첫 번째 수와 두 번째 수를 입력한다. - 처리: 두 수를 더한다. - 출력: 더한 결과를 대담한다. • 어떤 블록을 사용해야 할까요? • 흐름을 생각하며 블록 쌓기 		 ★흐름도(모둠)
과제해결 (15')	<ul style="list-style-type: none"> ◎ 간단한 프로그램 만들기 <ul style="list-style-type: none"> • 선택 미션 제시: ① 두 친구의 이름을 입력하면 하나로 합쳐주는 프로그램 ② 두 수를 빼는(곱하는, 나누는) 프로그램 ③ 세 수를 더하는 프로그램 • 입력-처리-출력 흐름으로 구분하여 시각화하기 • 흐름을 생각하며 블록 쌓기 • 결과 공유하기 		 ★흐름도(개인)
과제성찰 (5')	<ul style="list-style-type: none"> ◎ 정리 및 성찰 <ul style="list-style-type: none"> • 개념 정리하기 - 정보를 받고(입력)→생각하고(처리)→결과를 보여준다.(출력) - 프로그램은 입력을 받아 처리한 뒤, 그 결과를 출력하는 논리적인 흐름을 따른다. 		엔트리(학급)전시

<참 고 문 헌>

- 교육부(2022). 2022 개정 교육과정 총론 해설.
- 교육부(2015). 2015 개정 교육과정 총론 해설.
- 교육부(2022). 2022 개정 교육과정(실과).
- 교육부(2015). 2015 개정 교육과정(실과).
- (주)금성출판사(2024). 초등학교 5~6학년군 실과 6 교사용 지도서.
- 론 리치하트, 마크 처치, 캐린 모리스 공저(2023). 생각이 보이는 교실.