



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

석사학위논문

문제 해결 과정 중심의
컴퓨팅 사고력 평가

제주대학교 교육대학원

컴퓨터교육전공

이 정 훈

2020년 8월



문제 해결 과정 중심의 컴퓨팅 사고력 평가

지도교수 조 정 원

이 정 훈

이 논문을 교육학 석사학위 논문으로 제출함

2020년 6월

이정훈의 교육학 석사학위 논문을 인준함

심사위원장 _____ 박 찬 정



위 원 _____ 김 철 민



위 원 _____ 조 정 원



제주대학교 교육대학원

2020 년 8 월



<국문초록>

문제 해결 과정 중심의 컴퓨팅 사고력 평가

이 정 훈

제주대학교 교육대학원 컴퓨터교육전공

지도교수 조 정 원

미래창조과학부는 소프트웨어가 혁신, 성장, 가치창출에 중심이 되고 개인, 기업, 국가에 영향을 미치는 소프트웨어 중심사회가 도래하였다고 선언하였다. 이에 따라 미래사회에서 소프트웨어가 가지는 비중은 매우 높아졌으며, 소프트웨어 교육의 중요성도 확대 되었다. 우리나라는 2015개정 교육과정에서 소프트웨어교육의 중요성을 강조하고 정보 교과를 필수 과목으로 지정하였다. 소프트웨어 교육의 목적은 컴퓨터 과학의 기본 원리를 바탕으로 문제를 해결해 나가는 컴퓨팅 사고력을 기르는 데에 있다. 소프트웨어 교육의 중요성이 커지면서 컴퓨팅 사고력 관련 논의는 활발히 이루어지고 있으나, 컴퓨팅 사고력을 어떻게 평가할 것인지에 대한 논의는 비교적 부족한 실정이다.

컴퓨팅 사고력을 어떻게 평가할 것인지는 매우 중요한 문제이다. 평가는 자신의 학습 수준을 파악하고 자기 성찰을 통해 부족한 부분을 보완하고 개선하는 영역으로, 소프트웨어에서의 평가의 방향도 평가를 통해 컴퓨팅 사고력을 향상시킬 수 있어야 한다. 최근 평가의 방향은 이전 평가와 다르게 학생의 수행을 통해 이루어지고, 학습결과에 대한 평가에서 학습을 위한 평가로 변화하고 있다. 이러한 평가의 방향으로 과정중심평가가 떠오르고 있다. 과정중심평가는 학습의 과정을 중시하는 평가를 강화하여 학생이 수행을 통해 자신의 학습을 성찰하도록 하고, 평가를 활용

하여 학생에게 피드백해주는 학습으로써의 평가 방향이다. 이를 가장 잘 반영한 평가는 개인이 습득한 지식을 실제 생활이나 평가 상황에서 얼마나 잘 수행하는지 어떻게 수행하는지 평가하는 과정중심 수행평가 방식이다. 컴퓨팅 사고력은 문제 해결 과정에서 발현되는 역량으로 컴퓨팅 사고력에 대한 평가 또한 학생의 문제 해결 과정 중심으로 평가가 이루어져야 한다.

본 논문은 학생의 문제 해결 과정에서 발현되는 컴퓨팅 사고력을 평가하기 위해 과정중심평가 방식을 활용하는 것을 목적으로 한다.

이를 위해 본 연구는 다음과 같은 절차로 진행되었다.

첫째, 컴퓨팅 사고력 기반 문제 해결 과정과 컴퓨팅 사고력의 핵심 요소를 정의하기 위해 관련 연구 동향과 문헌, 선행 연구를 분석하여 비교 및 정리하여 정의하였다.

둘째, 과정중심평가 방식을 적용하기 위하여 컴퓨팅 사고력 기반의 문제 해결 과정 성취기준을 정의하였다. 성취기준은 2015개정 중등 정보교과를 중심으로 분석 및 개발하였다.

셋째, 개발한 성취기준을 학습 내용과 평가 내용으로 분해 및 분석하여 평가 요소를 추출하였다. 평가 요소는 학습자를 관찰 및 평가·피드백 할 수 있는 근거가 된다.

넷째, 평가 활동에서 학생이 어느 정도 수준에 도달하였는지 확인할 수 있는 실질적인 기준을 제시하기 위하여 평가 기준을 제시하였다. 평가 기준을 상/중/하 3단계 위계로 구분되며 행동동사를 사용하였다.

문제 해결 과정 중심 컴퓨팅 사고력 평가는 개발된 성취기준을 바탕으로 교수·학습 내용과 평가 방법을 설계하며, 학습자의 문제 해결 과정 중심으로 평가 요소 및 평가 기준을 토대로 학습자를 관찰, 결과물, 포트폴리오 등 다방면으로 평가하고 평가 과정에서 인지적, 정의적으로 피드백 하는 것을 목적으로 하였다.

개발된 내용은 10명의 전문가 집단의 2차에 걸친 델파이 조사를 거쳐 수정 및 보완되었으며, 타당도 검증을 통해 타당성을 입증하였다.

본 논문에서 개발한 문제 해결 과정, 성취기준, 평가 요소, 평가 기준을 참고 한다면 문제 해결 과정 중심의 컴퓨팅 사고력을 평가하고 교수·학습 평가를 설계하는데 용이해질 것으로 보인다. 이를 토대로 컴퓨팅 사고력 향상 및 평가에 기여할 수 있기를 기대한다.

<제목 차례>

I. 서론	1
1. 연구의 필요성 및 목적	1
2. 연구의 내용	3
II. 이론적 배경	4
1. 컴퓨팅 사고력(Computational Thinking)	4
1) 컴퓨팅 사고력 정의	4
2) 컴퓨팅 사고력 구성요소	6
2. 문제 해결 과정	8
3. 과정중심 수행평가	10
4. 2015개정 중등 정보교과 교육과정 및 성취기준	14
1) 2015개정 정보교과 교육과정	14
2) 2015개정 중등 정보교과 교육과정 성취기준	15
III. 문제 해결 과정 중심 컴퓨팅 사고력 평가 모델 개발	17
1. 연구 개발 방법 및 절차	17
1) 컴퓨팅 사고력 기반 문제 해결 과정 개발	18
2) 문제 해결 과정 성취기준 개발	22
3) 평가 요소 추출 및 개발	23
4) 평가 기준 개발	24
2. 전문가 검토 및 결과	26
1) 내용 타당도 검증	26
2) 1차 델파이	29
3) 2차 델파이 및 결과	32

3. 컴퓨팅 사고력 기반 문제 해결 과정	34
4. 문제 해결 과정 성취기준	37
5. 평가 요소	38
6. 평가 기준	40
7. 문제 해결 과정 중심의 컴퓨팅 사고력 평가 모델	43
IV. 결론	47
[참고 문헌]	50
[ABSTRACT]	53
[부록]	56
1. 중학교 컴퓨팅 사고력 과정평가 모델 차시 별 교수·학습 과정안	56
2. 중학교 컴퓨팅 사고력 과정평가 모델 교수학습 활동지	70

<표 차례>

<표 II-1> 컴퓨팅 사고력의 조작적 정의	5
<표 II-2> ISTE&CSTA 컴퓨팅 사고력의 구성요소와 정의	6
<표 II-3> 교육부 컴퓨팅 사고력의 구성요소와 정의	7
<표 II-4> 연구자별 컴퓨팅 사고력 구성요소 및 상호 관련성	7
<표 II-5> 기존의 평가 체제와 과정 중심 수행평가 비교	13
<표 II-6> 2015개정 중학교 정보교과 교육과정 영역별 성취기준	15
<표 III-1> 최숙영의 컴퓨팅 사고력 기반 문제 해결 과정(2016)	19
<표 III-2> 이철현의 컴퓨팅 사고력 기반 문제 해결 과정(2016)	20
<표 III-3> 전용주의 컴퓨팅 사고력 기반 문제 해결 과정(2017)	20
<표 III-4> 유지수의 컴퓨팅 사고력 기반 문제 해결 과정(2019)	21
<표 III-5> 교육부 컴퓨팅 사고력의 구성요소와 정의	22
<표 III-6> 컴퓨팅 사고력 기반 문제 해결 과정 관련 정보교과 성취기준	22
<표 III-7> 블룸의 개정된 텍사노미	24
<표 III-8> 본 연구에서 제시한 평가 기준 위계	25
<표 III-9> 현장 적용 시 평가기준을 검토하는 준거	26
<표 III-10> 델파이 조사에 참여한 전문가 집단	26
<표 III-11> 응답자 수에 따른 내용타당도 비율(CVR)의 최솟값	28
<표 III-12> 협력적 문제해결 기술	30
<표 III-13> 1차 델파이를 통해 수정된 컴퓨팅 사고력 핵심 요소	31
<표 III-14> 전문가 델파이 결과	32
<표 III-15> 제시하는 문제 해결 과정	35
<표 III-16> 문제 해결 과정에 적용되는 컴퓨팅 사고력 핵심 요소	35
<표 III-17> 컴퓨팅 사고력 기반 문제 해결 과정 성취기준	37
<표 III-18> 성취기준 기반 평가 요소	38
<표 III-19> 성취기준 평가 기준	42
<표 III-20> 수행 과제 개발 체크리스트	44

<그림 차례>

[그림 II-1] 현재 상태와 목표 상태 사이의 관계	8
[그림 II-2] Polya의 문제해결 단계	9
[그림 II-3] 창의적 문제 해결 단계	9
[그림 II-4] 교수·학습과 평가가 연계된 순환적 구조에서의 평가	10
[그림 II-5] 과정중심평가와 결과중심평가, 수행평가와 지필평가의 개념 영역	11
[그림 II-6] 과정중심평가와 수행평가의 특징	12
[그림 III-1] 연구 개발 절차	17
[그림 III-2] 평가 요소 추출 방법	23
[그림 III-3] 1차 델파이를 통해 수정된 문제 해결 과정	29
[그림 III-4] 본 연구에서 제시하는 컴퓨팅 사고력 기반 문제 해결 과정	36
[그림 III-5] 문제 해결 과정 중심 컴퓨팅 사고력 평가 모델	43

I. 서론

1. 연구의 필요성 및 목적

세계경제포럼(이하 WEF)은 2016년 다보스포럼을 통해 인공지능(이하 AI), 기계 학습(ML), 로봇공학, 나노기술, 3D프린팅과 유전학, 생명공학기술과 같이 이전에는 서로 단절되어 있던 분야들의 경계가 파괴되고 융·복합을 통해 발전해나가는 ‘기술 혁신’의 패러다임인 4차 산업혁명 시기에 접어들었다고 강조했다. 이러한 변화의 중심에는 소프트웨어(이하 SW)가 있으며 미래창조과학부(2014)는 ‘SW가 혁신과 성장, 가치창출의 중심이 되고 개인·기업·국가의 경쟁력을 좌우하는 SW중심사회가 도래하였다’라고 선언하였다.

SW가 중심이 되는 산업의 융·복합은 일자리의 변화를 야기한다. WEF은 2018년 미래 일자리를 전망한 보고서에서 전통적인 사무직 관련 직업은 감소하고 AI 및 데이터, 정보 보안 등 최신 기술을 이해하고 활용하는 인력에 대한 수요가 크게 증가할 것이라고 보고하였다. 이는 곧 4차 산업혁명을 주도할 미래인재상의 변화를 뜻한다. WEF는 미래인재가 가져야할 핵심 역량으로 비판적 사고력과 문제해결력(Critical thinking/problem solving), 창의력(Creativity), 의사소통능력(Communication), 협업능력(Collaboration)인 4C를 강조하였다. 이러한 핵심역량을 갖춘 미래인재를 양성할 교육으로 SW교육이 주목받고 있다. SW교육의 목표는 컴퓨터 과학의 기본 개념과 원리를 기반으로 다양한 문제를 창의적이고 효율적으로 해결하는 컴퓨팅 사고력을 기르는 교육(교육부, 2015)으로, 우리나라는 2015개정 교육과정을 통해 SW교육의 중요성을 강조하며 초·중등학교에서 단계적으로 소프트웨어 교육을 의무화 하였다.

SW교육의 중요성이 커짐에 따라 SW교육의 목적인 컴퓨팅 사고력을 어떻게 함양하고 평가할 것인가는 중요한 문제로 대두되었다. 컴퓨팅 사고력은 2006년 Jeannette Wing의 ‘Computational Thinking’이라는 논문에서 주목받기 시작했다. Wing은 컴퓨팅 사고력이 컴퓨터 과학 분야의 기술을 활용하여 현실 문제의 효

과적인 해결책을 찾아내는 사고력이며, 3R(읽기, 쓰기, 셈하기)과 더불어 모든 사람이 갖추어야 할 기초 역량이라고 주장하였다. 즉, 컴퓨팅 사고력은 단순히 코딩, 프로그래밍 능력을 기르는 것이 아니라 컴퓨터를 기반으로 한 문제 해결 능력이다(교육부, 2015). 이러한 컴퓨팅 사고력을 함양시키기 위한 다양한 연구가 활발하게 진행되고 있지만, 컴퓨팅 사고력을 어떻게 평가할 것인가에 대한 연구는 비교적 부족한 실정이다.

평가는 자신의 학습수준을 파악하고 자기성찰을 통해 부족한 부분을 보완, 개선하는 관점에서 매우 중요한 영역이다. 최근 평가의 방향은 단순히 학습자의 지식이나 기능의 정도를 측정하는 형태의 평가에서 벗어나 학습자의 인지 구조 변화나 이해 수준을 진단하고 촉진하여 평가 자체가 경험이 되도록 ‘과정’에 초점을 두는 평가 방식으로 변화하고 있다(전경희, 2016). 교육부는 최근 평가의 패러다임에 발맞춰 2015개정교육과정 총론에서 ‘학습의 과정을 중시하는 평가를 강화하여 학생이 자신의 학습을 성찰하도록 하고, 평가 결과를 활용하여 교수·학습의 질을 개선’하도록 표기하여 평가를 학습 도구로써 활용하도록 하였다. 최근 평가 패러다임은 컴퓨팅 사고력의 이상적인 평가 방향과도 같다. 컴퓨팅 사고력은 컴퓨터를 기반으로 해결하고자 하는 문제를 효과적이고 효율적으로 해결하는 절차적 능력으로(이영준 외, 2014) 문제 해결을 위한 일련의 사고 과정에서 발휘되는 역량이다. 따라서 학습자가 문제를 해결해 나가는 과정에서 평가가 이루어져야 한다. 기존의 컴퓨팅 사고력 평가를 위한 평가 도구는 대부분 지필 또는 결과물 위주의 전통적인 평가에 기반을 두고 인지적인 측면에서의 평가가 이루어지고 있다. 이는 문제 해결을 위한 컴퓨팅 사고력의 일련의 사고과정을 종합적으로 평가하는 데에 어려움이 있다. 지필 또는 결과 중심의 평가 방식에서 벗어나, 학습자의 문제 해결 과정을 평가하는데 적합한 최근의 평가 방향으로 변화하여야 한다.

최근 평가의 패러다임인 과정중심평가는 암기위주나 정답 찾기 등 결과 중심의 평가 방식에서 문제 해결 과정을 중시하는 평가로 전환해야 한다는 의미를 가지고 있으며(김순남 외, 2013), 결과 중심에서 발생하는 한계를 극복하고 ‘학습을 위한 평가’, ‘학습으로써의 평가’로 확장된 평가 패러다임이다. 과정중심평가는 교육과정에 해당하는 성취기준을 기반으로 평가 계획을 세우고 교수·학습과 평가가 연계되어 교수·학습 과정에서 학생의 인지적, 정의적인 평가가 이루어지고 평가 과정에서

적절한 피드백을 주어 학습의 질을 개선하기 위해 사용된다(임은영, 2017). 또한 기존 지필평가에서 벗어나 수행 중심의 평가가 강조되고 있다. 수행 중심의 능동적 평가를 수행평가라 하며 이는 개인이 습득한 지식을 실제 문제나 평가 상황에서 얼마나 잘 수행하는지, 어떻게 수행하는지 관찰, 서술 등 다양한 방법을 통해 판단하는 평가 방법이다. 이러한 평가 패러다임에 발맞춰 학습자의 문제 해결 과정 중심의 컴퓨팅 사고력 평가를 위해 과정중심 수행평가를 활용하여 평가를 진행한다면 학습자의 문제 해결 과정을 중점적으로 컴퓨팅 사고력을 평가함과 동시에 적절한 피드백을 통한 컴퓨팅 사고력 향상을 기대할 수 있을 것이다.

2. 연구의 내용

본 연구의 목적은 학습자의 문제 해결 과정에서 발현되는 컴퓨팅 사고력 역량을 평가하기 위해 과정중심평가를 이용하여 컴퓨팅 사고력 평가 모델을 제안하고자 한다. 이를 위한 연구의 주요 내용 및 방법은 다음과 같다.

첫째, 2015개정 정보교과 교육과정과 컴퓨팅 사고력 기반의 문제 해결 과정 선행 연구를 분석하여 ‘문제 해결 과정’ 정의한다.

둘째, 성취기준을 바탕으로 평가를 구성하는 과정중심평가를 적용하기 위해 2015개정 중등 정보교과 교육과정에서 제시하고 있는 컴퓨팅 사고력 관련 성취기준을 분석하여 ‘문제 해결 과정 성취기준’을 재구성하여 정의한다.

셋째, 학습자의 문제 해결 과정에서 학습자의 수준을 다방면으로 수집하고 객관적인 평가가 이루어지기 위해, 앞서 정의한 성취기준을 분해하여 교수·학습 과정에서 활용할 ‘평가 요소’를 추출하여 제시한다.

넷째, 학습자의 성취기준 도달 여부를 알려주는 실제적인 기준을 제공하기 위하여 ‘평가 기준’을 제시한다.

다섯째, 앞서 제시한 컴퓨팅 사고력 기반의 ‘문제 해결 과정’, ‘문제 해결 과정 성취기준’, ‘평가 요소’, ‘평가 기준’을 전문가 델파이 과정을 거쳐 수정 및 보완하고, 학습자의 문제 해결 과정 중심 컴퓨팅 사고력 평가 모델을 제시한다.

II. 이론적 배경

1. 컴퓨팅 사고력(Computational Thinking)

1) 컴퓨팅 사고력 정의

컴퓨팅 사고력(Computational Thinking, CT)은 문제를 해결하기 위한 기법이 라고 할 수 있다. 문제가 무엇인지 파악하고 그 문제를 해결할 수 있는 방법을 컴퓨터 과학을 기반으로 한 사고방식으로 해결방안을 찾는 과정 및 절차이다.

컴퓨팅 사고력은 Seymour Papert 교수가 1980년대에 처음 사용하고, 2006년 Wing의 ‘Computational Thinking’이라는 논문을 통해 주목받기 시작했다. Wing은 컴퓨팅 사고력이 ‘해결해야 할 문제를 만났을 때 컴퓨터 과학자처럼 사고하여 문제를 해결하는 것’이라고 정의하며 ‘문제해결 과정에서 문제를 단순화하고, 추상화 과정을 통해 문제의 핵심요소를 추출하고 모델링하여, 컴퓨팅 기기를 통한 해법을 자동화하는 능력’이라 하였다. Wing은 컴퓨터 과학이 계산(Computing)에 대한 학문이기 때문에 다음과 같은 특징을 지니고 있다고 설명하고 있다(Wing, 2008).

컴퓨팅 사고력의 핵심은 프로그래밍이 아닌 개념화에 있다.

컴퓨팅 사고력은 단순 반복적인 기술이 아닌 모든 사람이 갖춰야 하는 핵심 역량이다.

컴퓨팅 사고력은 컴퓨터가 아닌 인간의 사고방법이다.

컴퓨팅 사고력은 수학적 사고와 공학적 사고를 보완하고 결합한다.

컴퓨팅 사고력은 인공물이 아닌 아이디어이다.

컴퓨팅 사고력은 모두를 위한 것이다.

한편, 미국의 ISTE(International Society for Technology in Education)와 CSTA(Computer Science Teachers Association)은 K-12에 해당되는 컴퓨팅 사고력을 ‘현대 사회에서 컴퓨팅이 가진 힘과 제약사항을 이해하고, 새로운 지식을

만들어 내거나, 시스템을 설계하여 문제를 해결하고자 하는 모든 분야에 사용될 수 있는 능력'이라 설명하며, 조작적 정의를 정의적 측면과 인지적 측면으로 나누어 <표 II-1>과 같이 정의하였다(ISTA & CSTA, 2011).

<표 II-1> 컴퓨팅 사고력의 조작적 정의(ISTE&CSTA, 2011)

구분	조작적 정의
<p>컴퓨팅 사고력이 갖는 특징 (인지적 측면)</p>	<ul style="list-style-type: none"> • 문제해결을 위해 컴퓨터나 도구를 사용할 수 있도록 문제를 표현하기 • 논리적으로 자료를 조직하고 분석하기 • 모델이나 시뮬레이션 등의 추상화를 통해 자료를 표현하기 • 알고리즘 사고와 같은 일련의 절차를 통해 해결책을 자동화하기 • 가장 효율적이고 효과적인 절차와 자원을 조합하여 목표를 달성하는데 필요한 해결책을 확인하고, 분석하고, 구현하기 • 문제 해결 절차를 다양한 문제로 일반화하고 적용하기
<p>컴퓨팅 사고력의 성향, 태도적 차원 (정의적 측면)</p>	<ul style="list-style-type: none"> • 복잡한 문제를 다루는 데 있어서의 자신감 • 어려운 문제를 다루는 인내력 • 모호성에 대한 허용 • 답이 정해지지 않은 개방형 문제를 다룰 수 있는 능력 • 공동의 목표나 해결책을 달성하기 위해서 다른 사람과 의사소통하고 일할 수 있는 능력

우리나라에서 컴퓨팅 사고력은 2009 개정 교육과정의 정보 교과 목표로 진술되면서 중학교 정보 교육과 컴퓨터 교육학의 중요한 패러다임으로 자리 잡았다. 최근 2015개정 교육과정에서 발표한 SW 교육과정에서는 컴퓨팅 사고력을 '컴퓨터 과학의 기본 개념과 원리, 컴퓨팅 시스템을 활용하여 실생활 및 다양한 학문분야의 문제를 이해하고 창의적으로 해법을 구현하여 적용할 수 있는 능력'이라 제시하고 그 역량으로 '추상화 능력', '자동화 능력', '창의·융합 능력'을 두고 있다.

각 정의를 통해 컴퓨팅 사고력의 개념을 정리해 보면 컴퓨팅 사고력은 학습자가 실생활에서 접할 수 있는 문제 상황에서 컴퓨터 과학의 개념과 원리를 기반으로 문제를 해결하는 사고과정, 절차적 능력이라 하겠다.

2) 컴퓨팅 사고력 구성요소

컴퓨팅 사고력에 대한 다양한 정의가 존재하는 만큼 컴퓨팅 사고력을 구성하는 구성요소 또한 다양하게 정의된다.

Wing(2008)은 컴퓨팅 사고력을 크게 실제 세계의 문제를 해결 가능한 형태로 표현하기 위한 사고력인 추상화(Abstraction)와 추상화 과정을 통해 만들어진 문제 해결 모델을 컴퓨터가 이해할 수 있는 프로그래밍 언어로 표현하여 인간이 처리하기 어려운 많은 양의 반복된 작업이나 시뮬레이션을 실시하는 자동화(Automation)로 구성되어 있다고 제시하였다(Wing, 2008).

미국의 ISTE & CSTA(2011)는 컴퓨팅 사고력 구성요소를 자료수집(Data Collection), 자료분석(Data Analysis), 자료표현(Data Representation), 문제분해(Problem Decomposition), 추상화(Abstraction), 알고리즘 및 절차(Algorithm and Procedures), 자동화(Automation), 시뮬레이션(Simulation), 병렬화(Parallelization)로 제시하였다(ISTE & CSTA, 2011).

<표 II-2> ISTE&CSTA 컴퓨팅 사고력의 구성요소와 정의(ISTE & CSTA,2011)

요소	정의
자료수집	문제의 이해와 분석을 토대로 문제를 해결하기 위한 자료를 모으는 단계
자료분석	수집된 자료와 문제에 주어진 자료를 세심히 분류하고 분석하는 단계
자료표현	문제의 자료 내용을 그래프, 차트, 단어, 이미지 등으로 표현하는 단계
문제분해	문제를 해결하기 위해 문제를 나누어 분석하는 단계
추상화	문제의 복잡도를 줄이기 위해 기본 주요 개념의 정의를 설정하는 단계
알고리즘 및 절차	지금까지의 문제를 해결하기 위한 과정을 순서적 단계로 표현하는 단계
자동화	순차적으로 나열하고 표현한 내용을 컴퓨팅 기기를 이용하여 문제해결의 최선책을 선택하는 단계
시뮬레이션	복잡하고 어려운 해결책이나 현실적으로 실행이 불가능한 해결책을 선택하기 위해 모의실험을 하는 단계
병렬화	목표를 달성하기 위한 작업을 동시에 수행하도록 자원 구성

교육부(2015)는 소프트웨어교육 운영지침을 통해 컴퓨팅 사고력의 구성요소를 자료 수집, 자료 분석, 구조화, 추상화(분해, 모델링, 알고리즘), 자동화(코딩, 시뮬레이션), 일반화로 나누어 제시하였다(교육부,2015).

<표 II-3> 교육부 컴퓨팅 사고력의 구성요소와 정의(교육부·KERIS, 2015)

요소		정의
자료 수집		문제해결에 필요한 자료를 모으기
자료 분석		자료의 이해, 패턴 찾기, 결론을 도출하기
구조화		문제를 그래프, 차트, 그림 등으로 시각화하기
추상화	분해	문제를 관리 가능한 수준의 작은 문제로 나누기
	모델링	문제 해결을 위한 핵심요소를 추출하고, 모델 만들기
	알고리즘	문제를 해결하기 위한 일련의 단계를 알고리즘으로 표현하기(절차적 표현)
자동화	코딩	프로그래밍 언어를 이용해, 문제해결과정을 자동화하기
	시뮬레이션	프로그램(소프트웨어) 실행하기
일반화		문제해결과정을 다른 문제에 적용하기

각 학자 및 기관의 컴퓨팅 사고력 구성요소를 살펴보면 그 의미는 대동소이하며 추상화 과정과 자동화 과정 의미의 세분화 정도의 차이만 있는 것으로 보인다. 컴퓨팅 사고력 구성요소의 상호 연관성을 살펴보면 다음과 같다.

<표 II-4> 연구자별 컴퓨팅 사고력 구성요소 및 상호 관련성

Wing(2006)	ISTE &CSTA(2011)	교육부,KERIS(2015)	
추상화	자료수집	자료수집	
	자료분석	자료분석	
	자료표현	구조화	
	문제분해	추상화	분해
	추상화		모델링
	알고리즘 및 절차		알고리즘
자동화	자동화	자동화	코딩
	시뮬레이션		시뮬레이션
	병렬화		
	-	일반화	

2. 문제 해결 과정

우리는 일상생활 속에서 수많은 문제를 접하고 해결해 나간다. 그렇다면 문제란 무엇일까? 문제가 있다는 것 내가 목표로 하는 상태와 현재의 상태가 같지 않은 상태이다. 즉, 문제란 현재 상태와 목표 상태의 갭(gap)이다. 여기서 현재의 상태는 ‘어떻게 되어 있는가?’하는 상황, 목표 상태는 ‘어떻게 되었으면 좋겠는가?’하는 전망, 갭은 두 상태간의 차이이다. 문제를 인식하기 위해서는 현재 상태와 목표 상태의 차이를 인식할 수 있어야 한다. 문제를 인식하지 못하는 이유는 현재 상태와 목표 상태의 차이를 인식하지 못하기 때문이다. 따라서 현재 상태와 목표 상태를 인식하여 갭을 발견할 수 있으면 남들이 문제라고 인식하지 못하는 것을 문제로 인식하여 문제를 발견할 수도 있다. 우리는 이를 문제 인식이라 한다.

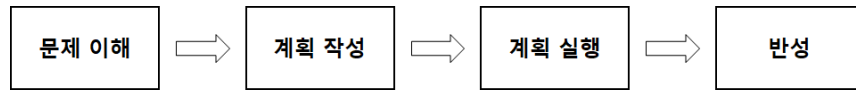


[그림 II-1] 현재 상태와 목표 상태 사이의 관계

현재 상태와 목표 상태간의 갭이 없으면 문제가 없다는 것이다. 갭을 없애는 과정에서 우리는 일반적으로 갭이 발생한 원인을 도출해 내고 갭을 줄일 수 있는 솔루션을 마련하여 문제를 해결한다. 이러한 과정이 바로 문제 해결 과정이다. 즉, 문제를 해결한다는 것은 현재 상태와 목표 상태간의 갭을 없애는 것이다. 문제 해결 과정을 보여주는 대표적인 예는 Polya(1957)이 제시한 문제 해결 단계와 Osborn(1953)이 제시한 창의적 문제 해결 모델이다.

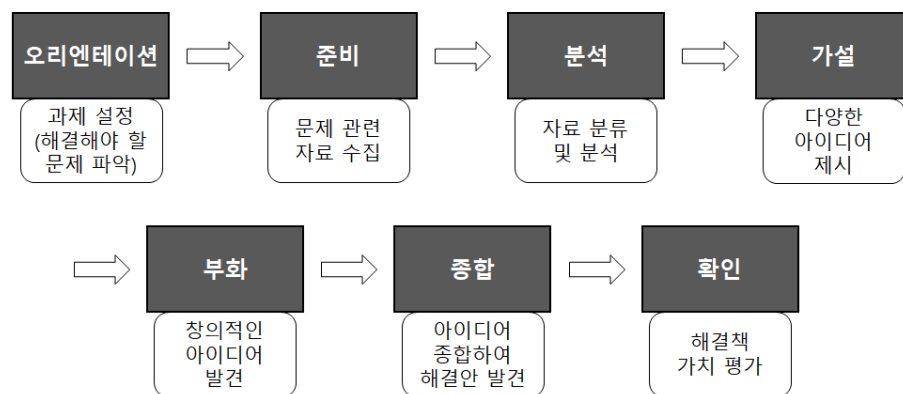
Polya(1957)는 문제를 해결한다는 것은 자신이 배운 지식과 원리를 응용하여 여러 가지 새로운 상황에서 당면하는 문제들의 해결 방안을 발견하는 것이라 말하며 문제 해결 단계를 4단계로 나누어 [그림 II-2]와 같이 제시하였다. 문제 이해 단계에서는 문제에서 구하려는 것과 주어진 것을 알고, 문제를 분석하는 단계이다. 계획 작성 단계에는 문제에서 주어진 것과 구하려는 것 사이의 관계를 파악하는 단

계로 여러 가지 문제 해결을 위한 전략을 이용한다. 계획 실행 단계에는 앞서 계획한 문제 해결 계획에 따라 실행 및 적용하는 단계이며, 마지막 반성 단계에는 문제 해결 과정을 되돌아보고 성찰하는 시간을 가진다.



[그림 II -2] Polya의 문제해결 단계(1957)

한편, 문제 해결 과정에서 창의성이 주목을 받음에 따라 창의적인 문제해결 방법에 대한 관심이 높아졌다. 최초의 창의적 문제해결(Creative problem solving: CPS)모델은 Alex Osborn이 제안한 것으로 [그림 II -3]과 같이 7단계로 기술하였다(Osborn, 1953). Osborn은 문제 해결을 위해서는 먼저 해결해야 할 문제가 무엇인지 파악하는 것이 중요하며, 해결해야 할 문제가 무엇인지 파악이 되면 그와 관련한 자료를 수집, 분류, 분석해야 한다고 하였다. 다음 단계로 다양한 아이디어가 나올 수 있도록 분위기를 조성하고 창의적 아이디어를 발견하여 해결안 설립 및 평가 단계가 이루어져야 한다 하였다. Osborn은 다양하고 독특한, 창의적인 아이디어가 많이 나올 수 있는 발산적 사고와, 생성해 낸 아이디어와 해결책을 분석하고 다듬어 가는 수렴적 사고를 강조하였다.

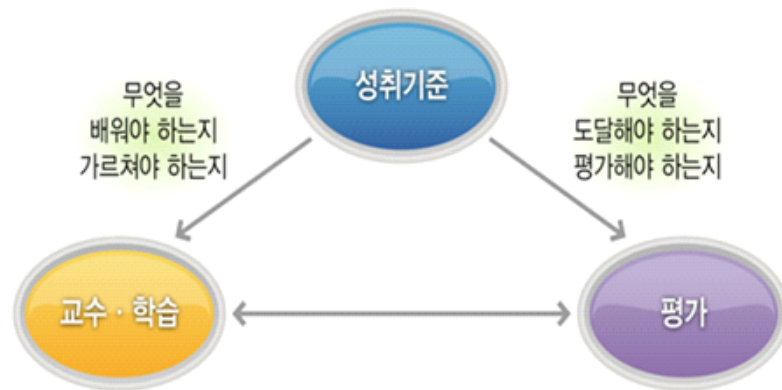


[그림 II -3] 창의적 문제 해결 단계(Osborn,1953)

이후 Polya와 Osborn이 각각 제시한 문제 해결 과정은 학자들에 의해 보완되어 다양한 유형이 존재하지만 일반적으로 문제의 인식 및 이해에서 출발해 해결안을 도출해 내고 해결안을 실행하여 평가로 끝나는 구조를 가지고 있다.

3. 과정중심 수행평가

미래사회에 요구되는 인재는 단편적인 지식을 가지고 있는 인재가 아닌, 지식을 활용하여 문제를 해결하는 능력, 지식과 지식을 융합하여 새로운 것을 창조하는 능력, 비판적인 사고를 가지고 사회를 바라보는 능력, 새로운 아이디어를 창출하는 능력, 자기 주도적 학습 태도와 협업 및 의사소통 능력 등이 요구되는 사회이다. 이러한 미래인재상의 변화는 교육의 변화와 함께 평가 패러다임의 변화를 야기하였다. 이에 우리나라는 2015개정 교육과정에서 미래사회가 요구하는 핵심역량을 함양하여 바른 인성을 갖춘 창의·융합 인재를 양성하기 위해 과정중심평가를 강조하였다.

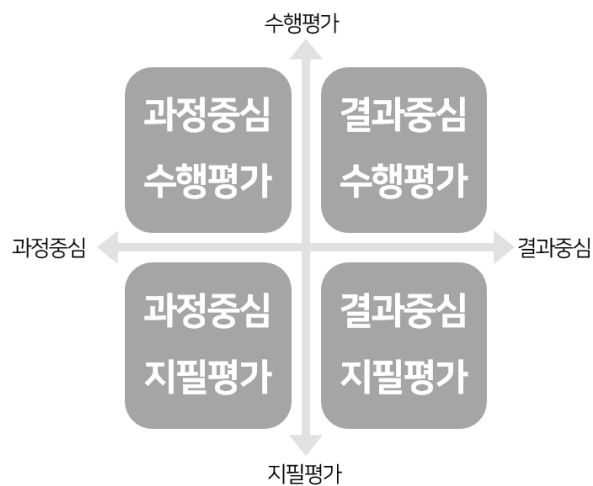


[그림 II-4] 교수·학습과 평가가 연계된 순환적 구조에서의 평가

※ 출처 : 교육부·한국교육과정평가원(2014)

한국교육과정평가원(2017)은 과정중심평가를 “교육과정의 성취기준에 기반한 평가 계획에 따라 교수·학습 과정에서 학생의 변화와 성장에 대한 자료를 다각도로

수집하여 적절한 피드백을 제공하는 평가”라고 정의하였다. 즉, 과정중심평가는 기존 평가에 새로운 의미가 더해진 것으로 평가를 단지 측정(measure)의 의미에 국한하지 않고, 교수·학습과 평가가 통합된 모습으로 전개된다는 점에서 기존의 평가 방식과는 차별되는 개념이다(전경희, 2016). 이러한 관점에서 본다면 과정중심평가는 ‘학습 결과에 대한 평가(assessment of learning)’에서 ‘학습을 위한 평가(assessment for learning)’ 또는 ‘학습으로써의 평가(assessment as learning)’로, 교수·학습과 연계된 평가라고 할 수 있다.

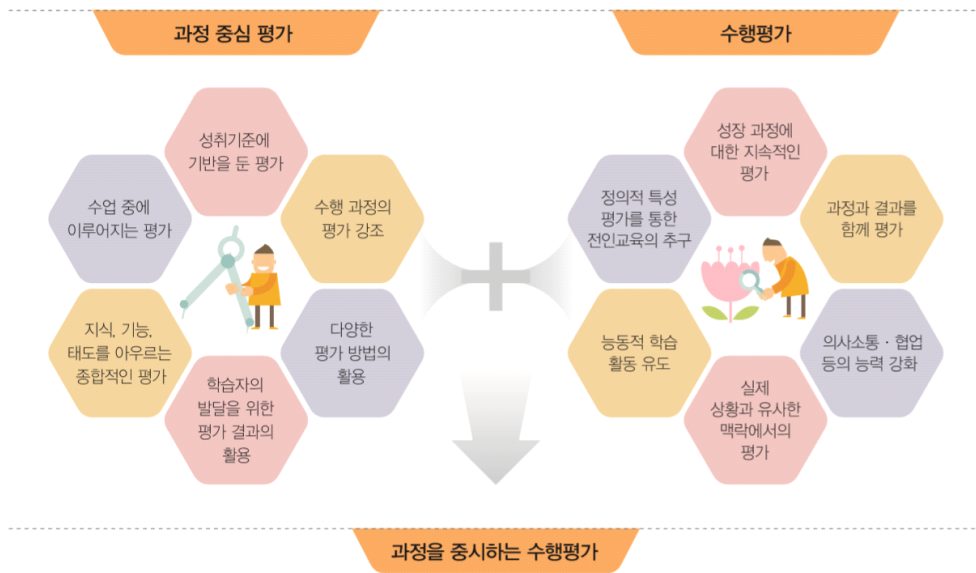


[그림 II -5] 과정중심평가와 결과중심평가, 수행평가와 지필평가의 개념 영역

※ 출처 : 과정중심평가의 특징과 의미에 관한 연구(임종현, 최원석, 2018) 재인용

과정중심평가는 학교 현장에서 수행평가의 개념으로 많이 활용되지만, 이는 과정중심평가를 좁은 의미로 바라본 것이다. 과정중심평가는 수행평가와 지필평가로 나눌 수 있으며 이와 반대되는 의미인 결과중심평가도 수행평가와 지필평가로 나눌 수 있다. 예를 들어 교사가 학습자의 수행 과정을 중점으로 관찰하여 평가를 한다면 ‘과정중심 수행평가’이고, 학습자의 서술형, 논술형 등의 지필평가를 한다면 ‘과정중심 지필평가’가 되는 것이다. 이와 반대로 교사가 학습자의 수행을 통한 결과물을 가지고 평가를 진행한다면 ‘결과중심 수행평가’가 되고, 중간고사, 기말고사 등 학업 성취의 결과를 가지고 평가를 진행한다면 ‘결과중심 지필평가’가 되는 것이다(임종현, 최원석, 2018).

최근 평가의 패러다임은 결과중심과 지필평가가 아닌, 과정중심과 수행평가를 지향한다. 수행평가는 개인이 습득한 지식, 기능이나 기술을 실제 생활이나 평가 상황에서 얼마나 잘 수행하는지, 어떻게 수행하는지를 서술, 관찰, 면접 등의 다양한 방법을 통해 종합적으로 판단하는 평가를 의미(성태제, 2014)하는 것으로 학습의 결과뿐만 아니라 과정을 중시하는 평가이다.



[그림 II -6] 과정중심평가와 수행평가의 특징

* 출처 : 교육부·한국교육과정평가원(2017)

수행평가가 가지는 장점은 첫째, 학생들이 스스로 과제를 해결해 나감으로써 단편적인 지식을 암기하는데 그치지 않고, 능동적으로 답을 구하거나 지속적으로 탐구하고 창의적인 사고력을 키우는데 효과적이다. 둘째, 수행평가는 산출물을 만들어 내므로 과정과 결과를 동시에 평가할 수 있다. 셋째, 단편적인 평가가 아니라 학습자의 변화와 발달과정을 종합적으로 판단하기 위해 지속적인 평가가 이루어진다. 넷째, 모둠활동을 통하여 의사소통, 협업 등의 사회적 기술 능력 강화에 도움이 된다. 다섯째, 실제 상황과 유사한 문제 상황을 통하여 지식이나 기술을 활용한 문제 해결력을 기를 수 있다. 여섯째, 학생의 인지적인 평가뿐만 아니라 정의적인 평가를 통해 전인적인 평가가 가능하다(전경희, 2016). 과정중심 수행평가는 학습자의 문제 해결 과정에 집중하고 교수·학습을 극대화 하는 평가, 학습자의 학습 수준에

대한 정보 수집을 위한 도구 및 과정으로서의 평가이며 기존 결과 지향적 평가에서 문제 해결 과정, 학습 과정 지향적 평가로 그 강조점이 바뀌는 것을 의미한다. 기존의 평가 체제와 과정중심 수행평가를 비교하면 <표 II-5>와 같다.

<표 II-5> 기존의 평가 체제와 과정 중심 수행평가 비교(부산광역시교육청,2016)

내용	기존의 평가 체제	과정중심수행평가
평가체제	<ul style="list-style-type: none"> • 상대평가 • 양적평가 	<ul style="list-style-type: none"> • 절대평가 • 질적평가
평가목적	<ul style="list-style-type: none"> • 선발, 분류, 배치 • 한 줄 세우기 	<ul style="list-style-type: none"> • 지도, 조언, 개선 • 여러 줄 세우기
평가내용	<ul style="list-style-type: none"> • 선언적(결과적, 내용적)지식 학습의 결과 중시 • 학문적 지능의 구성 요소 	<ul style="list-style-type: none"> • 절차적(과정적, 방법적) 지식 학습의 결과 및 과정 중시 • 실천적 지능의 구성 요소
평가방법	<ul style="list-style-type: none"> • 선택형 문항을 사용한 지필평가 중심 • 일회적 평가 • 객관성, 일관성, 공정성 강조 	<ul style="list-style-type: none"> • 다양한 평가 방법 고려 • 지속적, 종합적인 평가 • 객관성, 일관성, 공정성을 비롯하여 전문성, 타당성, 적합성 강조
평가시기	<ul style="list-style-type: none"> • 학습 활동이 종료되는 시점 • 교수·학습과 평가 활동 분리 	<ul style="list-style-type: none"> • 학습 활동의 모든 과정 • 교수·학습과 평가 활동 통합
교사의 역할	<ul style="list-style-type: none"> • 지식의 전달자 	<ul style="list-style-type: none"> • 학습의 안내자·촉진자
학생의 역할	<ul style="list-style-type: none"> • 수동적인 학습자 • 지식의 재생산자 	<ul style="list-style-type: none"> • 능동적인 학습자 • 지식의 창조자
교수·학습	<ul style="list-style-type: none"> • 교사 중심 • 인지적 영역 중심 • 암기 위주 • 기본 학습능력 강조 	<ul style="list-style-type: none"> • 학생 중심 • 인지적·정의적 영역 모두 강조 • 탐구 위주 • 창의성 등 고등 사고능력 강조

기존 평가 체제는 측정, 분류, 배치가 목적이며 결과를 중시하지만, 과정중심 수행평가 체제는 지도, 조언, 개선이 목적이며 과정을 중시하는 것을 알 수 있다. 또한 기존의 평가 체계에서의 교사의 역할은 학습자를 평가하고 지식을 전달하는 역할이지만 과정중심 수행평가에서 교사의 역할은 학습자의 문제 해결 과정에서 평가를 통해 학습자의 부족한 점을 찾아내고 의사소통을 통해 적절한 피드백을 주는 안내자, 촉진자의 역할을 한다.

4. 2015개정 중등 정보교과 교육과정 및 성취기준

1) 2015개정 정보교과 교육과정

우리나라에서 정보 교과 관련 교육과정이 도입된 시기는 2차 교육과정에서 전자 계산 교육으로 처음 시작이 되었다. 그 후 6차 교육과정에서 초·중·고등학교까지 컴퓨터 관련 내용과 과목이 정착되었으며, 2007 개정 교육과정에 들어서 중·고등 교과의 명칭을 '정보'로 통일하고 교과의 내용을 컴퓨터 과학의 기본 개념과 원리를 바탕으로 구성되었다. 2009 개정 교육과정에서는 논리적 사고력, 창의적 문제 해결력, 계산적 사고력을 강조했으며 2015개정 정보교과 교육과정에 들어서 컴퓨팅 사고력을 기반으로 문제 해결 역량을 기르기 위한 내용으로 구성되었다.

2015개정 정보교과 교육과정의 목표 및 성격은 정보와 정보처리기술을 올바르게 활용하고, 새로운 지식과 정보, 기술을 창의적으로 생성하고 협력적으로 문제를 해결하는 능력을 갖춘 인재를 양성하는 것이다(교육부, 2015).

소프트웨어 교육의 일반적인 목표는 학생들의 컴퓨팅 사고력을 향상시키는데 있다. 2015개정 정보과 교육과정에서도 컴퓨팅 사고력을 핵심 역량으로 선정하여 정보문화소양과 협력적 문제해결력을 갖추고 컴퓨팅 사고력으로 문제를 창의적이고 효율적으로 해결하는 역량을 기르기 위한 것을 목표로 하고 있다. 중학교 '정보'교과의 목표는 다음과 같다(교육부, 2015).

정보사회의 특성을 올바르게 이해하고 정보윤리를 실천할 수 있는 태도를 기른다.

정보기술을 활용하여 문제 해결에 필요한 자료와 정보를 수집하고 효율적으로 구조화하는 능력과 태도를 기른다.

컴퓨터과학의 기본 개념과 원리에 따라 실생활의 문제를 추상화하여 해법을 설계하고 프로그래밍 과정을 통해 소프트웨어로 구현하여 자동화할 수 있는 능력을 기른다.

컴퓨팅 시스템의 구성 및 동작원리를 이해하고 다양한 입·출력 장치와 프로그래밍을 통해 문제 해결에 적합한 피지컬 컴퓨팅 시스템을 구성하는 능력을 기른다.

2) 2015개정 중등 정보교과 교육과정 성취기준

2015개정교육과정 정보 교과의 내용은 ‘정보문화’, ‘자료와 정보’, ‘문제 해결과 프로그래밍’, ‘컴퓨팅 시스템’ 영역으로 구분되며, 각 영역의 성취기준을 통해 핵심 역량을 신장하는데 중점을 두고 있다. 여기서 성취기준이란 교과를 통해 배우고 학생들이 할 수 있어야 할, 또는 할 수 있기를 기대하는 능력이며, 내용(지식)과 기능(수행 능력, 태도)으로 구성되어 있다. 정보 교과 영역별 성취기준은 <표 II-6> 과 같다.

<표 II-6> 2015개정 중학교 정보교과 교육과정 영역별 성취기준(교육부, 2015)

영역	성취기준
정보문화	9정01-01 정보기술의 발달과 소프트웨어가 개인의 삶과 사회에 미친 영향과 가치를 분석하고 그에 따른 직업의 특성을 이해하여 자신의 적성에 맞는 진로를 탐색한다.
	9정01-02 정보사회 구성원으로서 개인의 정보와 저작권 보호의 중요성을 인식하고 개인정보 보호, 저작권 보호 방법을 실천한다.
	9정01-03 정보사회에서 개인이 지켜야 하는 사이버 윤리의 필요성을 이해하고 사이버 폭력 방지와 게임·인터넷·스마트폰 중독의 예방법을 실천한다.
자료와 정보	9정02-01 디지털 정보의 속성과 특징을 이해하고 현실 세계에서 여러 가지 다른 형태로 표현되고 있는 자료와 정보를 디지털 형태로 표현한다.
	9정02-02 인터넷, 응용 소프트웨어 등을 활용하여 문제 해결을 위한 자료를 수집하고 관리한다.
	9정02-03 실생활의 정보를 표, 다이어그램 등 다양한 형태로 구조화하여 표현한다.
추상화와 알고리즘	9정03-01 실생활 문제 상황에서 문제의 현재 상태, 목표 상태를 이해하고 목표 상태에 도달하기 위해 수행해야 할 작업을 분석한다.
	9정03-02 문제 해결에 필요한 요소와 불필요한 요소를 분류한다.
	9정03-03 논리적인 문제 해결 절차인 알고리즘의 의미와 중요성을 이해하고 실생활 문제의 해결과정을 알고리즘으로 구상한다.
	9정03-04 문제 해결을 위한 다양한 방법과 절차를 탐색하고 명확하게 표현한다.
프로그래밍	9정04-01 사용할 프로그래밍 언어의 개발 환경 및 특성을 이해한다.
	9정04-02 다양한 형태의 자료를 입력 받아 처리하고 출력하기 위한 프로그램을 작성한다.
	9정04-03 변수의 개념을 이해하고 변수와 연산자를 활용한 프로그램을

	작성한다.
	9정04-04 순차, 선택, 반복의 개념과 원리를 이해하고 세 가지 구조를 활용한 프로그램을 작성한다.
	9정04-05 실생활 문제 해결을 위한 소프트웨어를 협력하여 설계, 개발, 비교 분석한다.
컴퓨팅 시스템	9정05-01 컴퓨팅 시스템을 구성하는 하드웨어와 소프트웨어의 역할을 이해하고 유기적인 상호 관계를 분석한다.
	9정05-02 센서를 이용한 자료 처리 및 동작 제어 프로그램을 구현한다.

정보문화 영역의 성취기준은 정보사회의 특성을 이해하고 소프트웨어의 중요성, 가치 인식과 정보사회 구성원으로서 갖추어야 할 정보윤리, 정보보호 능력 함양을 위한 개인정보, 저작권 보호, 사이버 윤리를 실천하는 데 중점을 두고 있다.

자료와 정보 영역의 성취기준을 살펴보면 주로 정보기술을 활용하여 정보를 탐색, 생산, 관리하는 능력과 태도를 함양하는데 중점을 두고 있다. 즉 문제 해결에 필요한 자료를 수집, 관리하고 정보를 효과적으로 전달하기 위해 구조화는 역량을 기르는 영역이다.

추상화와 알고리즘 영역은 실생활의 문제를 추상화 하는 과정을 거쳐 해결하기 쉬운 형태로 만들고 문제 해결을 위한 알고리즘을 설계하는 능력과 태도를 함양하는데 중점을 둔 영역이다.

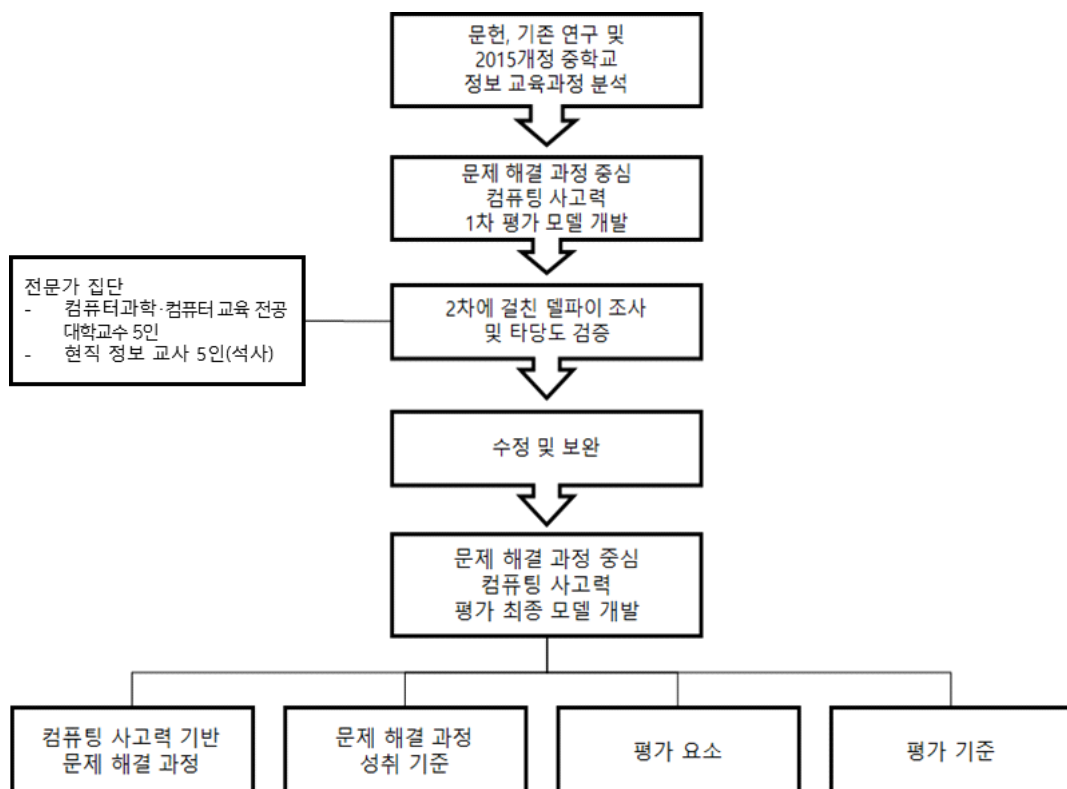
프로그래밍 영역은 추상화와 알고리즘을 통해 설계한 문제 해결 모델을 자동화 하는 능력을 함양하는 데 중점을 두어 설정하고 있다. 이 과정에서 프로그램의 입력과 출력, 변수와 연산, 실행 흐름, 제어 구조 등의 기본 개념과 원리를 적용하는 것이 중요하다.

컴퓨팅 시스템 영역은 주로 컴퓨팅 시스템의 구성과 동작 원리의 이해, 센서를 통한 자료의 입력과 처리, 동작 제어를 위한 프로그램을 설계하고 개발하는데 중점을 둔 영역이다.

III. 문제 해결 과정 중심 컴퓨팅 사고력 평가 모델 개발

컴퓨팅 사고력의 의미와 최근 평가의 패러다임 측면에서 보면 컴퓨팅 사고력의 평가는 학습자가 문제를 해결해 나가는 과정에서 이루어지는 것이 바람직하다. 이를 위해 본 논문에서 개발할 내용은 다음과 같다. 첫째, 학습자가 문제를 해결하는 과정에서 발현되는 컴퓨팅 사고력을 평가하기 위해 컴퓨팅 사고력 기반의 ‘문제해결 과정’을 정의한다. 둘째, 결과가 아닌 과정에 초점을 맞추고, 평가 자체가 교수·학습 과정이 되는 과정중심 수행평가 적용을 위해 ‘성취기준’을 정의한다. 셋째, 성취기준을 기반으로 학습자의 수준을 다방면으로 수집하고 객관적인 평가가 이루어지기 위해 ‘평가요소’를 제시한다. 넷째, 학습자의 성취기준 도달 여부를 알려주는 실질적인 기준을 제공하기 위해 성취기준을 기반으로 ‘평가기준’을 제시한다.

1. 연구 개발 방법 및 절차



[그림 III-1] 연구 개발 절차

본 연구는 학습자의 문제 해결 과정에서 발현되는 컴퓨팅 사고력을 평가하기 위하여 과정중심 수행평가 방식을 적용하였다. 이를 위해 컴퓨팅 사고력 기반의 ‘문제 해결 과정’, ‘문제 해결 과정 성취기준’, ‘평가 요소’, ‘평가 기준’을 개발하였다. 개발한 요소는 컴퓨터과학·컴퓨터교육 전공의 대학 교수 5인, 컴퓨터 과학·컴퓨터 교육 전공의 석사 학위를 가진 현직 정보교사 5인으로 구성된 전문가 집단의 2차에 걸친 델파이 검증을 통해 평가 모델을 수정 및 보완하고 최종 모델을 개발하여 연구를 완성하였다.

본 연구에서 개발한 문제 해결 과정 중심 컴퓨팅 사고력 평가 개발 절차를 도식화 하면 [그림 III-1]과 같다. 제시된 내용을 살펴보면, 학습자의 컴퓨팅 사고력은 문제를 해결하는 과정에서 발현되므로 컴퓨팅 사고력 기반 문제 해결 과정을 기존 문헌과 연구를 분석하여 제시하였다. 또한 과정중심 수행평가가 이루어지기 위해서는 성취기준의 확립이 중요하므로, 2015개정 중학교 정보교과 분석과 기존 문헌연구를 통해 문제 해결 과정 성취기준을 제시하였다. 마지막으로 학습자에 대한 컴퓨팅 사고력 평가와 피드백의 근거가 되기 위한 평가 요소와 평가 기준은 성취기준을 토대로 분석하여 제시하여 1차 평가 모델을 제시하였다. 개발한 ‘문제 해결 과정’, ‘문제 해결 과정 성취기준’, ‘평가 요소’, ‘평가 기준’은 전문가 델파이를 통해 수정, 보완되었다.

1) 컴퓨팅 사고력 기반 문제 해결 과정 개발

수행평가는 학습자가 습득한 지식, 기능이나 기술을 실제 생활이나 인위적 평가 상황에서 얼마나 잘 수행하는지, 어떻게 수행하는지 수행과정에 초점을 둔 평가 방법이다. 컴퓨팅 사고력 또한 문제 상황에서 컴퓨터 과학의 개념과 원리를 기반으로 문제를 해결하는 사고과정, 절차적 능력이다. 즉, 수행평가는 컴퓨팅 사고력을 기반으로 한 문제 해결 과정에서 평가하기 적절한 평가 방법이다. 이를 토대로 본 논문은 컴퓨팅 사고력 기반의 문제 해결 과정을 정의하고 학습자의 문제 해결 과정에서 과정중심 수행평가를 적용하고자 한다. 컴퓨팅 사고력 기반으로 문제 해결 과정을 정의한 논문을 살펴보면 다음과 같다.

최숙영(2016)은 문제해결의 관점에서 컴퓨팅 사고력 증진을 위한 교수학습에 대한 연구를 통해 컴퓨팅 사고력이 컴퓨터를 기반으로 하는 문제 해결에 관한 하나

의 사고 과정으로 정의될 수 있기 때문에 일반적인 문제 해결의 과정에서 관련된 컴퓨팅 사고력 개념들을 살펴보는 것은 컴퓨팅 사고력을 이해하는데 도움이 될 수 있다 하였다. 최숙영은 이 논문을 통해 컴퓨팅 사고력의 핵심 요소를 뽑아내고 핵심 요소들이 문제 해결 과정에서 어떻게 사용되는지 기술하며 일반적인 문제 해결 과정을 ‘문제인식’, ‘문제분석’, ‘해결책고안’, ‘문제 해결의 실천’, ‘문제 해결의 평가’ 5단계로 나누고, 각 단계에 컴퓨팅 사고력 핵심 요소를 배치하였다.

<표 III-1> 최숙영의 컴퓨팅 사고력 기반 문제 해결 과정(2016)

문제 해결 과정		CT 핵심 요소
문제인식	해결하려고 하는 문제가 무엇인지 파악하는 단계	자료수집
		자료분석
문제분석	문제에서 중요한 핵심 요소들을 뽑아내고 커다란 문제를 해결 가능한 작은 단계로 구성하는 단계	자료분석
		문제분해
		추상화
해결책고안	분석된 내용을 토대로 해결 방법을 고안하고 가장 적절한 방법을 선택하는 단계	알고리즘 및 절차
문제해결의 실천	고안된 해결책을 직접 실행해보고 실행단계에서 발생하는 문제들을 검토하고 해결책을 수정 보완하는 단계	자동화
		디버깅
		시뮬레이션
문제해결의 평가	문제해결 단계에서 어려웠던 점, 후에 이와 유사한 문제를 수행하게 될 때 적용할 수 있는 기법, 개선해야 될 점 등을 살펴보는 단계	평가
		일반화

이철현(2016)은 소프트웨어 교육을 위한 컴퓨팅 사고력 기반 문제 해결 모형 (CT-PS Model)을 제시하며 소프트웨어 교육의 목적인 컴퓨팅 사고력 증진을 위하여 컴퓨팅 사고력을 적용할 수 있는 다양한 문제를 발굴하고, 문제를 해결하는 과정에서 컴퓨팅 사고력 핵심 요소를 경험하고 습득함으로써 컴퓨팅 사고력을 향상시킬 수 있다고 하였다. 이철현은 문제 해결 과정을 ‘문제분석’, ‘추상화’, ‘구현’ 3 단계로 제시하고 컴퓨팅 사고력 핵심 요소를 ‘분해’, ‘패턴인식’, ‘추상화’, ‘알고리즘’, ‘자동화’로 제시하였다.

<표 III-2> 이철현의 컴퓨팅 사고력 기반 문제 해결 과정(2016)

문제 해결 과정		CT 핵심 요소
문제분석	문제분해하기	주어진 문제의 규모를 줄이거나 일부를 선택하는 등 작게 나눈다.
	문제해결 핵심 아이디어 찾기	문제 분해를 통해 해결의 핵심 아이디어를 찾는다.
추상화	패턴 찾기	핵심 아이디어를 통해 반복되는 패턴을 찾는다. 경우에 따라 패턴이 없는 문제도 있을 수 있다.
	알고리즘설계	문제 해결의 핵심 아이디어와 패턴을 토대로 문제 해결 절차를 의사코드나 순서도로 구성한다.
구현	코딩	적절한 프로그래밍 요소를 이용하여 추상화 단계에서 설계한 알고리즘에 따라 코딩을 진행한다.
	테스트	코딩한 내용을 실행하여 결과를 확인한다.
	디버깅	오류가 발생할 경우 원인을 찾아서 수정한다.

전용주(2017)는 새로운 교육과정의 소프트웨어 교육을 위한 컴퓨팅 사고력 기반의 창의적 문제해결(CT-CPS) 수업 모형을 개발하여 2015 개정 정보교육과정의 목표인 창의적 문제 해결력을 위한 문제 해결 과정 수업 모형을 제시하였다. 창의적 문제 해결 과정의 각 단계에서 컴퓨팅 사고력의 핵심 요소들이 사용되어 실생활의 문제를 컴퓨팅 사고력을 통해 창의적으로 해결해 나가는 과정을 수업의 단계로 표현하였으며, 창의적 문제 해결 단계를 ‘문제인식 및 분석’, ‘아이디어 구상’, ‘설계’, ‘구현 및 평가’로 나누어 제시하였다.

<표 III-3> 전용주의 컴퓨팅 사고력 기반 문제 해결 과정(2017)

문제 해결 과정		CT 핵심 요소
문제인식 및 분석	일상생활 속에서 문제를 인식하고 관련 자료를 수집, 분석하여 보고서, 프리젠테이션 또는 기타 시각화된 자료로 제시	자료수집
		자료 분석
아이디어 구상	인식한 문제를 해결하기 위한 다양한 방법에 관한 구상	추상화 문제분해
설계	구상된 아이디어를 다양한 시행착오와 시도를 통해	알고리즘 및 절차

	최적안을 선택하고 이를 해결하는 알고리즘을 구체적으로 설계	
구현 및 평가	설계된 해결책 중 최선의 방법을 택하여 프로그램을 통해 자동화, 시뮬레이션을 통한 공유	자동화
		시뮬레이션
		병렬화

유지수(2019)는 컴퓨팅 사고력이 문제를 해결하는 과정이라 설명하며 일상생활에서 마주치는 문제를 포함한 대다수의 문제는 일반적인 문제 해결 과정인 Polya의 문제 해결 단계(Poly, 1957)의 각 단계에서 컴퓨팅 사고력이 사용된다면 좀 더 의미해질 것으로 보고 ‘문제 인식’, ‘문제 분석’, ‘해결방법 탐색’, ‘해결방법 실천’, ‘평가’ 5단계를 제시하며 각 단계에서 사용되는 컴퓨팅 사고력을 제시하였다.

<표 III-4> 유지수의 컴퓨팅 사고력 기반 문제 해결 과정(2019)

문제 해결 과정		CT핵심 요소
문제 인식	상황을 정확히 인식하고 그 원인을 찾음	문제 분해
문제 분석	문제를 이해하기 쉬운 형태로 표현	문제 분해 추상화
해결방법 탐색	논리적이고 체계적으로 문제 해결 방법을 탐색	자료 수집 자료 분석 자료 표현 알고리즘과 절차
해결방법 실천	문제 해결 방법을 적용	시뮬레이션
평가	문제 해결 과정을 돌아봄	자동화 병렬화

최숙영, 이철현, 전용주, 유지수의 컴퓨팅 사고력 기반 문제 해결 과정을 살펴보면 문제의 인식 및 분석 과정을 거쳐 해결책을 탐색하고 알고리즘을 설계하여 프로그래밍을 통해 자동화 해결 방법을 실천한 후 공유 및 평가의 과정을 통해 문제를 해결 하였다. 또한, 각 연구를 살펴보면 문제 해결 단계마다 사용되는 컴퓨팅 사고력 핵심 요소를 표기하고 있다. 본 연구 또한 컴퓨팅 사고력 핵심 요소를 표기하기 위하여 교육부가 소프트웨어교육 운영지침에서 제시한 컴퓨팅 사고력의 세부 구성요소의 정의를 바탕으로 개발하여 제시하였다.

<표 III-5> 교육부 컴퓨팅 사고력의 구성요소와 정의(교육부·KERIS, 2015)

요소		정의
자료수집		문제 해결에 필요한 자료를 모으기
자료 분석		자료의 이해, 패턴 찾기, 결론을 도출하기
구조화		문제를 그래프, 차트, 그림 등으로 시각화하기
추상화	분해	문제를 관리 가능한 수준의 작은 문제로 나누기
	모델링	문제 해결을 위한 핵심요소를 추출하고, 모델 만들기
	알고리즘	문제를 해결하기 위한 일련의 단계를 알고리즘으로 표현하기(절차적 표현)
자동화	코딩	프로그래밍 언어를 이용해, 문제 해결과정을 자동화하기
	시뮬레이션	프로그램(소프트웨어)를 실행하기
일반화		문제 해결과정을 다른 문제에 적용하기

2) 문제 해결 과정 성취기준 개발

본 연구는 학습자의 문제 해결 과정에서 컴퓨팅 사고력을 평가하기 위해 결과중심이 아닌 과정중심 수행평가 방식을 이용한다. 과정중심 수행평가는 결과에 초점을 맞춘 과거의 평가방식을 지양하고 ‘학습을 위한’ 평가, ‘학습으로써의’ 평가로 확장된 평가 패러다임이다. 이와 같은 과정중심 수행평가가 실천되기 위해서는 교육 평가와 교육과정, 교수·학습이 분리될 수 없다. 즉, 과정중심 수행평가가 제대로 작동하기 위해서는 교육과정 내용으로 수행 과제를 제시 및 교수·학습 과정을 설계하고 무엇을 어떻게 평가할지 평가 방안을 설계하는 교육과정-교수·학습-교육평가가 연계되는 평가 방법이다. 교육과정은 성취기준과 직결되므로 성취기준을 준거로 교수·학습 계획과 평가 계획을 세워야 한다. 본 논문은, 컴퓨팅 사고력 기반의 문제 해결 과정에서 학습자들이 할 수 있는, 할 수 있기를 바라는 성취기준을 정의하기 위해 2015개정 정보 교과 교육과정의 영역별 성취기준을 분석하여 컴퓨팅 사고력과 직접적인 연관이 있는 성취기준을 뽑아내고 컴퓨팅 사고력 기반 문제 해결 과정의 정의에 맞게 재구성하여 개발하였다.

<표 III-6> 컴퓨팅 사고력 기반 문제 해결 과정 관련 정보교과 성취기준(교육부, 2015)

관련 성취 기준
9정03-01 실생활 문제 상황에서 문제의 현재 상태, 목표 상태를 이해하고 목표 상태에 도달하기 위해 수행해야 할 작업을 분석한다.
9정02-02 인터넷, 응용 소프트웨어 등을 활용하여 문제 해결을 위한 자료를 수집하고

- 관리한다.
- 9정02-03 실생활의 정보를 표, 다이어그램 등 다양한 형태로 구조화하여 표현한다.
- 9정03-02 문제 해결에 필요한 요소와 불필요한 요소를 분류한다.
- 9정03-03 논리적인 문제 해결 절차인 알고리즘의 의미와 중요성을 이해하고 실생활 문제의 해결과정을 알고리즘으로 구상한다.
- 9정03-04 문제 해결을 위한 다양한 방법과 절차를 탐색하고 명확하게 표현한다.
- 9정04-01 사용할 프로그래밍 언어의 개발 환경 및 특성을 이해한다.
- 9정04-02 다양한 형태의 자료를 입력 받아 처리하고 출력하기 위한 프로그램을 작성한다.
- 9정04-03 변수의 개념을 이해하고 변수와 연산자를 활용한 프로그램을 작성한다.
- 9정04-04 순차, 선택, 반복의 개념과 원리를 이해하고 세 가지 구조를 활용한 프로그램을 작성한다.
- 9정04-05 실생활 문제 해결을 위한 소프트웨어를 협력하여 설계, 개발, 비교 분석한다.

2015개정 중학교 정보교과 교육과정에서 제시하고 있는 성취기준은 핵심 요소 추출과 관련된 성취기준을 강조하고 문제 분해와 직접 관련된 성취기준은 존재하지 않았기 때문에 문제 분해 영역에 대한 성취기준 내용 추가하였으며, 자동화 과정에서 디버깅과 관련한 성취기준 내용 추가, 공유 및 평가 단계와 관련한 성취기준을 기존 문헌과 연구 내용을 분석하여 재구성하였다.

3) 평가 요소 추출 및 개발

02-01 **문제와 관련된 자료**를 수집, 저장, 관리, 공유하고 **표, 다이어그램** 등으로 구조화 하여 표현한다.

내용	기능
문제와 관련된 자료 표, 다이어그램	수집하기 저장하기 관리하기 공유하기 구조화하기 표현하기
학습 요소 추출	
<ul style="list-style-type: none"> • 문제와 관련된 자료 수집하기 • 문제와 관련된 자료 저장하기 • 문제와 관련된 자료 관리하기 • 문제와 관련된 자료 공유하기 • 문제와 관련된 자료를 표, 다이어그램 등으로 구조화하여 표현하기 	
평가 요소 추출	
<ul style="list-style-type: none"> • 문제와 관련된 자료를 수집할 수 있는가? • 수집한 자료를 컴퓨터로 저장, 관리, 공유할 수 있는가? • 문제와 관련된 자료를 표, 다이어그램 등으로 구조화하여 표현할 수 있는가? 	

[그림 III-2] 평가 요소 추출 방법

성취기준은 교육내용을 통해 배워야 할 지식과 그 지식을 통해 할 수 있어야 할, 할 수 있기를 기대하는 수행능력을 표현한 것으로 내용(지식)과 기능(수행능력, 태도)으로 구성되어 있다. 즉, 성취기준은 무엇을 가르칠 것인지, 무엇을 평가할 것인지에 대한 교육내용인 것이다. 이를 토대로 본 연구에서는 평가 요소를 추출하기 위해 성취기준을 내용과 기능으로 분해하여 평가 요소를 추출하였다. 평가 요소 추출 방법은 [그림 III-2]와 같다.

본 연구에서 개발한 성취기준 '02-01 문제와 관련된 자료를 수집, 관리, 공유하고 표, 다이어그램 등으로 구조화 하여 표현한다.'에서 내용은 '문제와 관련된 자료', '표, 다이어그램'이며, 기능은 '수집하기', '저장하기', '관리하기', '공유하기', '구조화하기', '표현하기'로 나눌 수 있다. 이를 통해 학습요소를 추출하면 '문제와 관련된 자료 수집하기', '문제와 관련된 자료 저장하기', '문제와 관련된 자료 관리하기', '문제와 관련된 자료 공유하기', '문제와 관련된 자료를 표, 다이어그램 등으로 구조화하여 표현하기'로 추출할 수 있으며 이를 통해 평가 요소를 추출하면 '문제해결에 필요한 자료를 수집할 수 있는가?', '수집한 자료를 컴퓨터로 저장, 관리, 공유할 수 있는가?', '문제와 관련된 자료를 표, 다이어그램 등으로 구조화하여 표현할 수 있는가?'로 추출할 수 있다. 이와 같은 방법으로 각 성취기준의 평가 요소를 추출하였다.

4) 평가 기준 개발

평가 기준이란 평가 활동에서 학생들이 어느 정도의 수준에 도달했는지를 판단하기 위한 실질적인 기준 역할을 할 수 있도록, 각 성취기준에 도달한 정도를 구분하고 각 도달 정도에 따라 학생들이 무엇을 할 수 있고 무엇을 알고 있는지를 기술한 것을 의미한다. 평가기준은 학생들의 성취 수준을 교사가 판단할 수 있도록 행동 동사를 명확하게 기술하는 것이 필요하다. 본 연구에서는 학생들의 수행에 의한 성취도를 행동동사를 통해 객관적으로 평가하기 위해 블룸의 개정된 디지털 텍사노미를 사용하여 평가 기준을 진술하려고 한다.

<표 III-7> 블룸의 개정된 텍사노미(2009)

고차원적 사고력(Higher Order Thinking Skills)	
창조하기 (Creating)	설계하기(Designing), 구성하기(Constructing), 계획하기(Planning), 제작하기(Producing), 발명하기(Inventing), 창안하기(Devising), 만들기(Making)
평가하기 (Evaluating)	확인하기(Checking), 가설화하기(Hypothesizing), 비평하기(Critiquing), 실험하기(Experimenting), 판단하기(Judging), 시험하기(Testing), 탐색하기(Detecting), 관찰하기(Monitoring)
분석하기 (Analysing)	비교하기(Comparing), 구조화하기(Organizing), 해체하기(Deconstructing), 속성 파악하기(Attributing), 개요화하기(Outlining), 찾기(Finding), 체계화하기(Structuring), 통합하기(Integrating)
응용하기 (Applying)	시행하기(Implementing), 수행하기(Carrying out), 사용하기(Using), 실행하기(Executing), 구현하기(Implementing)
이해하기 (Understanding)	해석하기(Interpreting), 요약하기(Summarising), 추론하기(Infering), 주석하기(Paraphrasing), 분류하기(Classifying), 비교하기(Comparing), 설명하기(Explaining), 예시하기(Exemplifying)
기억하기 (Remembering)	인지하기(Recognizing), 열거하기(Listing), 묘사하기(Describing), 알아보기(Identifying), 검색하기(Retrieving), 이름 붙이기(Naming), 배치하기(Locating), 찾기(Finding)
저차원적 사고력(Lower Order Thinking Skills)	

성취기준에 대한 평가 기준은 ‘상/중/하’ 세 가지 수준으로 나누어 기술하고자 한다. 상 수준은 성취기준을 바탕으로 우수하게 문제를 해결할 수 있는 수준, 중 수준은 성취기준을 바탕으로 문제 해결이 가능한 수준, 하 수준은 성취기준을 바탕으로 문제 해결을 하지 못하는 수준으로 기준을 두고 나누어 제시하였다.

<표 III-8> 본 연구에서 제시한 평가 기준 위계

평가 기준	일반적 특성
상	성취기준을 바탕으로 우수하게 문제를 해결할 수 있는 수준
중	성취기준을 바탕으로 문제 해결이 가능한 수준
하	성취기준을 바탕으로 문제 해결을 하지 못하는 수준

성취 기준을 기반으로 한 평가기준안이 작성되면 전문가 검토를 통해 수정 과정을 거쳐 타당성을 확보해야 하며, 현장 적용 시 <표 III-9> 와 같은 검토 준거를 토대로 지속해서 수정 및 보완이 이루어져야 한다.

<표 III-9> 현장 적용 시 평가기준을 검토하는 준거(교육과정평가연구, 2009)

준거	비고
• 적은 수의 기준이 설정되어 숙달이 이루어질 수 있게 되어있는가?	
• 발달 단계에 적합하고 노력하면 달성이 가능한 것인가?	
• 명확하고 구체적인가?	
• 성취기준에 직접 관련이 있는가?	
• 학년별 수준에 따라 강조되는 내용과 일치하는가?	

2. 전문가 검토 및 결과

본 연구는 학습자의 문제 해결 과정 중심의 컴퓨팅 사고력 평가를 적용하기 위해 ‘컴퓨팅 사고력 기반 문제 해결 과정’, ‘문제 해결 과정 성취기준’, ‘평가 요소’, ‘평가 기준’을 개발하였으며, 전문가 검토를 통해 타당성을 확보하고자 하였다. 이를 위해 컴퓨터 과학·교육을 전공한 대학 교수 박사 5인과 현직 정보교사 5인으로 구성된 전문가 집단을 대상으로 e-mail을 통해 2차에 걸친 델파이 과정을 거쳤다.

<표 III-10> 델파이 조사에 참여한 전문가 집단

구분	직책	전공	학위	경력
전문가 1	교수	컴퓨터 과학	박사	15~20년
전문가 2	교수	컴퓨터 과학	박사	10~15년
전문가 3	교수	컴퓨터 교육	박사	10~15년
전문가 4	조교수	컴퓨터 과학	박사	10~15년
전문가 5	조교수	컴퓨터 교육	박사	10~15년
전문가 6	고등학교 교사	컴퓨터 교육	석사	15~20년
전문가 7	고등학교 교사	컴퓨터 교육	석사	10~15년
전문가 8	중학교 교사	컴퓨터 교육	석사	10~15년
전문가 9	중학교 교사	컴퓨터 교육	석사	10~15년
전문가 10	중학교 교사	컴퓨터 교육	석사	10~15년

1) 내용 타당도 검증

기존 문헌 및 연구와 2015개정 정보교과 교육과정 분석을 통해 개발한 ‘컴퓨팅 사고력 기반 문제 해결 과정’, ‘문제 해결 과정 성취기준’, ‘평가 요소’, ‘평가 기준’의 결과물에 대한 내용 타당도를 산출하기 위하여 델파이 설문 통계가 이루어졌다. 전문가 설문의 내용은 ‘문제 해결 과정’, ‘문제 해결 과정 각 단계별 정의’, ‘문제

해결 과정 컴퓨팅 사고력 핵심 요소', '문제 해결 과정 성취기준', '성취기준 기반의 평가 요소', '평가 기준 위계', '평가 기준 행동 기술'에 대한 질문으로 이루어졌으며, 전문가 델파이 조사를 통한 자료 통계 분석은 기본적으로 평균(M), 중앙값(Md), 표준편차(SD), 1사분위수(Q_1), 3사분위수(Q_3)를 산출하고 수렴도, 합의도, 안정도와 내용 타당도 비율(Content Validity Ratio : CVR)를 계산하여 산출하였다.

수렴도는 전문가의 의견이 한 점으로 수렴하였을 때 0의 값을 가진다. 즉, 수렴도는 0에 가까울수록 타당함을 의미하는 것이다. 수렴도를 구하는 공식은 다음과 같다(이종성, 2001)

$$\text{수렴도} = \frac{Q_3 - Q_1}{2}$$

합의도는 3사분위수와 1사분위수가 일치하여 완전의 합의 되었을 때 1의 값을 가지고, 의견의 편차가 클수록 감소한다. 즉, 합의도는 1에 가까울수록 타당함을 의미하는 것이다. 합의도를 구하는 공식은 다음과 같다(이종성, 2001).

$$\text{합의도} = 1 - \frac{Q_3 - Q_1}{Md}$$

안정도는 설문과정에서 응답자의 응답 결과 차이가 적어 응답의 일치성이 높은 경우 안정도가 확보되었다고 본다. 안정도를 측정하기 위한 방법으로 변이계수(Coefficient of Variation)가 사용되며 이를 구하는 공식은 다음과 같다(노승용, 2006).

$$\text{안정도} = \frac{SD}{M}$$

안정도는 0에 가까울수록 일치성이 높아 안정도가 확보되었다고 보며, 안정도가 0.5 이하인 경우 추가적인 설문이 필요 없고, 0.5~0.8인 경우 비교적 안정적이라고 판단, 0.8 이상인 경우 안정적이지 못한 수준으로 추가적인 설문이 필요한 것으로 판단한다(노승용, 2006).

내용 타당도를 측정하는 방법 중 가장 널리 사용되는 방법은 Lawshe(1975)가 개발한 내용 타당도 비율(CVR)이다. CVR은 다음과 같은 공식으로 구할 수 있다.

$$CVR = \frac{N_e - \frac{N}{2}}{\frac{N}{2}}$$

여기서 N 은 전체 평정자수 이며, N_e 는 타당하다고 응답한 평정자의 수이다. CVR은 설문에 응답한 응답자 수에 따라 최솟값을 제시하고 있다.

<표 III-11> 응답자 수에 따른 내용타당도 비율(CVR)의 최솟값(Lawshe,1975)

응답자 수	CVR 최솟값
5	.99
6	.99
7	.99
8	.75
9	.78
10	.62
11	.59
12	.56
13	.54
14	.51
15	.49
20	.42
25	.37
30	.33
35	.31
40	.29

본 연구에서는 설문 응답을 ‘전혀 그렇지 않다’는 1점, ‘그렇지 않다’는 2점, ‘보통이다’는 3점, ‘그렇다’는 4점, ‘매우 그렇다’는 5점으로, 5점 척도 응답을 받았다. 타당하다고 응답하는 기준은 ‘그렇다’, ‘매우 그렇다’로, 2가지 응답 경우를 기준으로 잡았다. 또한 응답자는 10명의 전문가가 응답하였으며, CVR 최솟값 0.62로 기준을 잡았다.

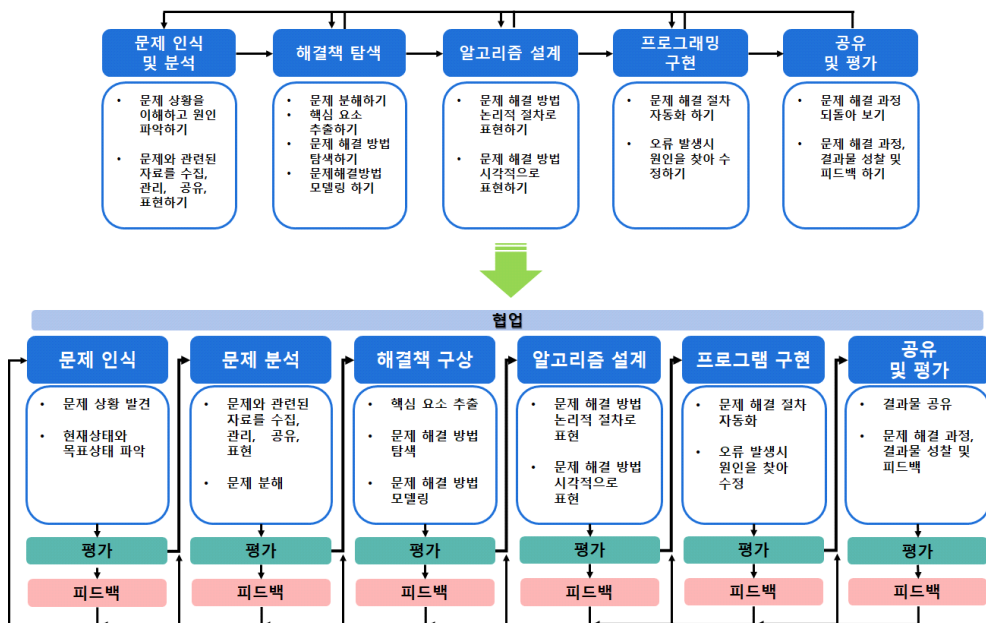
2) 1차 델파이

1차 델파이 조사에서는 CVR값 0.62 이하인 부분과 의견의 합의가 이루어지지 않은 곳을 중점으로 피드백이 이루어졌다. 전문가 집단의 의견을 수렴을 수렴하고 수정 및 보완하였으며, 1차 전문가 집단의 주된 피드백은 다음과 같다.

첫째, 문제 인식과 문제 분석을 구분해야 한다. 문제를 발견하고 현재 상태와 목표 상태의 차이를 인식하는 과정은 중요하다. 문제의 원인을 파악하고 문제 분석을 통해 문제와 관련한 자료를 수집, 분석하여 문제를 분해하고 해야 할 작업들을 파악하여야 한다.

둘째, 알고리즘을 설계하기 전에 모델링 작업이 필요하다. 모델링은 현실의 문제를 컴퓨터 세계로 가져가는 핵심 절차로, 알고리즘은 모델링을 통해 설계된 모델에 따라 완전히 달라진다. 따라서 해결책이 탐색되었으면 모델링 과정을 거쳐 해당 문제를 어떻게 컴퓨터 세계로 가져가고 표현할 것인지를 결정해야한다.

셋째, 문제 해결 단계별 컴퓨팅 사고력 핵심 요소를 세분화 하여 표기할 필요가 있다. 교육부가 제시한 컴퓨팅 사고력 핵심 요소로 평가, 공유의 개념을 담기 어려운 부분이 있으며, 문제 상황을 이해하고 원인 파악을 위한 추상화 과정에서 추상화(모델링)라 표기하는 것은 모델링 이라는 단어로 혼돈을 불러올 수 있다.



[그림 III-3] 1차 델파이를 통해 수정된 문제 해결 과정

1차 델파이 과정을 통해 수정된 문제 해결 과정은 [그림 Ⅲ-3]과 같다. 문제를 발견하고 현재 상태와 목표 상태를 이해하는 과정은 중요한 과정이므로 ‘문제 인식 및 분석’ 단계를 ‘문제 인식’, ‘문제 분석’으로 나누어 제시하였다. 또한 문제 해결을 위한 문제 해결 방법 모델링 과정을 ‘해결책 구상’단계에 추가하여 제시하였다. 또한, 현재의 문제 해결 단계에 대한 평가를 통해 다음 단계로 넘어가거나 이전 단계나 현재 단계에서 수정이 이루어지는 형태로 수정이 이루어 졌으며 문제 해결 단계에는 협업이 베이스가 되는 것을 추가하였다. 협업은 새로운 개념이 아니라 이미 협동학습이나 모둠을 구성하여 문제를 해결해나가는 형태로 학교 현장에서 많이 활용되어온 개념이다. PISA(2015)는 협력적 문제해결력을 ‘두 명 이상의 학습자가 문제해결에 요구되는 이해와 노력을 공유하고, 그 해결책까지 도달하기 위해 그들의 지식, 기술, 노력을 모아서 문제를 풀고자 하는 과정에서 효과적으로 몰두하는 개인의 능력’이라 표현하며 협력적 문제해결력의 핵심요소 3가지를 제시하기도 하였다. PISA가 제시한 핵심요소와 문제 해결 과정 적용 기술은 <표 Ⅲ-12>와 같다.

<표 Ⅲ-12> 협력적 문제해결 기술(PISA,2015)

		협력적 문제해결 역량		
		(1) 공유된 이해의 확립 및 유지	(2) 문제해결을 위한 적절한 행동	(3) 팀조직 구축 및 유지 관리
문 제 해 결 과 정	(A) 탐색 및 이해	(1A) 팀원들의 관점과 능력 찾아내기	(2A) 목표에 부합된 협력적 상호 작용 유형 찾기	(3A) 문제해결을 위한 역할 이해 하기
	(B) 표현 및 형식화	(1B) 공유된 표현 만들기 문제의 미에 대해 협상하기	(2B) 완성할 과제를 구체화하여 표현하기	(3B) 역할과 팀 조직 표현하기
	(C) 계획 및 실행	(1C) 수행역할에 대한 팀원 의사 소통하기	(2C) 계획 실행하기	(3C) 참여의 규칙 따르기
	(D) 모니터링 및 반영	(1D) 공유된 이해에 대한 점검과 수정하기	(2D) 행동결과에 대한 모니터링 과 문제해결성과 평가하기	(3D) 모니터링, 피드백 제공, 팀 조직 및 역할 적응하기

‘공유된 이해의 확립 및 유지’ 역량은 학생들이 각자 문제에 대해 어떤 지식을 가지고 있는지와 함께 문제에 대한 시각 및 팀원의 관점을 인지하는 역량이다. ‘문제 해결을 위한 적절한 행동’ 역량은 문제 해결을 위해 팀원간의 문제의 제한점을 이해하고, 팀의 목표를 설정하고, 과제를 수행하여 최종적으로 결과를 점검해 나가는 행동과 관련된 역량이다. ‘팀을 조직하고 유지’ 역량은 자신과 다른 팀원의 역할을 이해하고, 팀원에게 적합한 역할을 수행하며, 팀을 조직하고 점검하며 문제 해결에 장애가 되는 요소를 잘 다룰 수 있는 역량이다. PISA가 제시한 역량을 바탕으로 문제 해결 과정의 단계에 맞게 협력적 문제 해결 과정과 평가에 반영할 수 있다.

컴퓨팅 사고력 핵심 요소는 <표 III-13>와 같이 수정되었다. 기존 컴퓨팅 사고력 핵심 요소의 문제는 문제 해결 과정의 의미와 활동을 모두 만족하지 못하는 데에 있다. 따라서 문제 인식을 위한 ‘추상화(핵심요소 추출)’ 와 공유 및 평가를 위한 ‘평가’요소를 추가하여 의미를 재정의 하였다. 이밖에도 자동화 과정에서 호출의 개념을 추가하였으며 성취기준, 평가요소, 평가기준에서 기술의 모호함에 대한 수정이 이루어 졌다.

<표 III-13> 1차 델파이를 통해 수정된 컴퓨팅 사고력 핵심 요소

1차 컴퓨팅 사고력 핵심 요소		피드백 이후 컴퓨팅 사고력 핵심 요소	
문제 해결 과정	컴퓨팅 사고력 핵심 요소	문제 해결 과정	컴퓨팅 사고력 핵심 요소
문제 인식 및 분석	자료 수집	문제 인식	추상화 (핵심요소 추출)
	자료 분석	문제 분석	자료 수집
	구조화		자료 분석
해결책 탐색	추상화(모델링)	해결책 구상	구조화
	추상화(분해)		추상화(분해)
알고리즘 설계	추상화(모델링)	알고리즘 설계	추상화 (핵심요소 추출)
	추상화(알고리즘)		추상화 (모델링)
	추상화(알고리즘)		추상화(알고리즘)

프로그램 구현	자동화(코딩)	프로그램 구현	자동화(코딩)
	자동화(시뮬레이션)		자동화(시뮬레이션)
공유 및 평가	일반화	공유 및 평가	일반화
			평가

3) 2차 델파이 및 결과

1차 델파이 조사를 통해 전문가 피드백을 수용하여 수정한 결과물을 바탕으로 2차 델파이 조사를 진행하였다. 2차 델파이 조사 결과 <표 III-14>에서 보는 바와 같이 타당도가 확보되었다.

<표 III-14> 전문가 델파이 결과

구분		<i>M</i>	<i>Md</i>	<i>SD</i>	<i>Q1</i>	<i>Q3</i>	수렴도	합의도	<i>CVR</i>	안정도
문제 해결 과정	Q.1-1	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.1-2.1	4.70	5.00	0.46	4.25	5.00	0.38	0.85	1.00	0.10
문제 해결 과정 각 단계별 정의	Q.1-2.2	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.1-2.3	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.1-2.4	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.1-2.5	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.1-2.6	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
문제 해결 과정 컴퓨팅 사고력 핵심 요소	Q.1-3.1	4.30	4.00	0.64	4.00	5.00	0.50	0.75	0.80	0.15
	Q.1-3.2	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.1-3.3	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.1-3.4	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.1-3.5	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.1-3.6	4.70	5.00	0.46	4.25	5.00	0.38	0.85	1.00	0.10
	Q.1-3.7	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.1-3.8	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.1-3.9	4.80	5.00	0.60	5.00	5.00	0.00	1.00	0.80	0.13
	Q.1-3.10	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
문제 해결 과정 성취기준	Q.2-1.1	4.70	5.00	0.46	4.25	5.00	0.38	0.85	1.00	0.10
	Q.2-1.2	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.2-1.3	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.2-1.4	4.60	5.00	0.49	4.00	5.00	0.50	0.80	1.00	0.11
	Q.2-1.5	4.60	5.00	0.49	4.00	5.00	0.50	0.80	1.00	0.11
	Q.2-1.6	4.70	5.00	0.64	5.00	5.00	0.00	1.00	0.80	0.14
	Q.2-1.7	4.80	5.00	0.60	5.00	5.00	0.00	1.00	0.80	0.13
	Q.2-1.8	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.2-1.9	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.2-1.10	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.2-1.11	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.2-1.12	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.3-1.1	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.2	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14

성취기준 기반 평가 요소	Q.3-1.3	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.4	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.3-1.5	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.6	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.3-1.7	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.3-1.8	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.3-1.9	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.10	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.11	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.12	4.80	5.00	0.60	5.00	5.00	0.00	1.00	0.80	0.13
	Q.3-1.13	4.70	5.00	0.46	4.25	5.00	0.38	0.85	1.00	0.10
	Q.3-1.14	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.3-1.15	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.3-1.16	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.3-1.17	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
Q.3-1.18	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00	
Q.3-1.19	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00	
Q.3-1.20	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06	
Q.3-1.21	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08	
성취기준 기반 평가 기준 위계	Q.4-1.1	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-1.2	4.60	5.00	0.66	4.25	5.00	0.38	0.85	0.80	0.14
	Q.4-1.3	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-1.4	4.80	5.00	0.60	5.00	5.00	0.00	1.00	0.80	0.13
	Q.4-1.5	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.4-1.6	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-1.7	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-1.8	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-1.9	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-1.10	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-1.11	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-1.12	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
성취기준 기반 평가 기준 행동 기술	Q.4-2.1	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-2.2	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-2.3	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-2.4	4.80	5.00	0.60	5.00	5.00	0.00	1.00	0.80	0.13
	Q.4-2.5	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-2.6	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-2.7	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.4-2.8	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-2.9	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-2.10	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.4-2.11	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
	Q.4-2.12	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-2.13	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
	Q.4-2.14	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
	Q.4-2.15	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.16	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06	
Q.4-2.17	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00	
Q.4-2.18	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08	
Q.4-2.19	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06	
Q.4-2.20	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06	
Q.4-2.21	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08	

Q.4-2.22	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
Q.4-2.23	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
Q.4-2.24	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.25	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
Q.4-2.26	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
Q.4-2.27	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.28	5.00	5.00	0.00	5.00	5.00	0.00	1.00	1.00	0.00
Q.4-2.29	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
Q.4-2.30	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.31	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.32	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
Q.4-2.33	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.34	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08
Q.4-2.35	4.90	5.00	0.30	5.00	5.00	0.00	1.00	1.00	0.06
Q.4-2.36	4.80	5.00	0.40	5.00	5.00	0.00	1.00	1.00	0.08

수렴도는 0.5를 넘지 않았으며 0.5로 산출된 응답을 살펴보면 평균 응답이 4.3과 4.6으로, 4점 또는 5점 응답으로 의견이 갈린 것을 알 수 있다. 합의도 또한 최솟값 0.75로 안정권에 든 것을 알 수 있다. 내용 타당도 비율인 CVR을 살펴보면, 기준치인 0.62를 모두 넘는 결과가 산출 되었으며, 안정도 또한 최댓값 0.15로 더 이상 설문을 진행하지 않아도 되는 수준으로 산출되었다.

3. 컴퓨팅 사고력 기반 문제 해결 과정

학습자의 수행에 의한 과정 중심 평가를 적용하기 위해 제안하는 컴퓨팅 사고력 기반의 문제 해결 과정은 ‘문제 인식’, ‘문제 분석’, ‘해결책 구상’, ‘알고리즘 설계’, ‘프로그램 구현’, ‘공유 및 평가’ 6단계로 정의하였다.

1단계 ‘문제 인식’ 단계는 일상생활 속에서 발생하는 문제를 발견하고, 문제 상황의 현재 상태와 목표 상태를 이해하여 문제를 인식하는 단계이다.

2단계 ‘문제 분석’ 단계는 문제와 관련된 자료를 수집, 관리, 공유하여 시각화된 자료로 정리하여 제시, 문제의 원인은 파악하고, 문제의 규모를 줄여 분해하는 단계이다.

3단계 ‘해결책 구상’ 단계는 문제 해결을 위해 문제에서 중요한 핵심 요소를 추출하고 패턴을 파악하거나 다양한 해결 방법에 대해 탐색하고 컴퓨터로 자동화하

여 적용할 수 있는 문제 해결 모델을 설계하는 단계이다.

4단계 ‘알고리즘 설계’ 단계는 앞선 단계에서 설계한 문제 해결 모델을 토대로 문제 해결 절차를 의사코드나 순서도로 표현하는 단계이다.

5단계 ‘프로그램 구현’ 단계는 앞서 고안된 알고리즘을 직접 자동화하여 실현해 보고, 발생하는 문제들을 검토하여 수정 및 보완한다.

6단계 ‘공유 및 평가’ 단계는 최종 결과물을 친구들과 공유하고 동료의 피드백을 받으며, 문제 해결 과정에서 어려웠던 점, 개선해야 할 점 등을 확인하고 구현한 문제 해결 방법과 유사한 문제, 다른 문제에 어떻게 적용할지 살펴보는 단계이다.

<표 III-15> 제시하는 문제 해결 과정

문제 해결 과정	정 의
문제 인식	생활 속 문제를 발견하고, 해결하고자 하는 문제 상황의 현재 상태와 목표 상태에 대해 이해 및 정의
문제 분석	문제와 관련된 자료를 수집, 관리, 공유 및 시각화된 자료로 정리하여 문제의 원인을 파악하고 문제를 분석하여 해결 가능한 단위의 문제로 분해
해결책 구상	문제 해결을 위한 핵심 요소 추출을 통하여 문제 해결을 위한 다양한 방법에 대해 구상 후 문제 해결 모델 설계
알고리즘 설계	문제 해결 모델을 토대로 문제 해결 절차를 의사코드나 순서도로 표현
프로그램 구현	문제 해결 절차를 프로그래밍을 통해 자동화하여 구현하고 오류가 발생할 경우 원인을 찾아 수정
공유 및 평가	결과물을 공유하고 문제 해결 단계에서 어려웠던 점, 개선하여야 할 점 등을 확인하고 이와 유사한 문제, 문제 해결 과정을 돌아봄

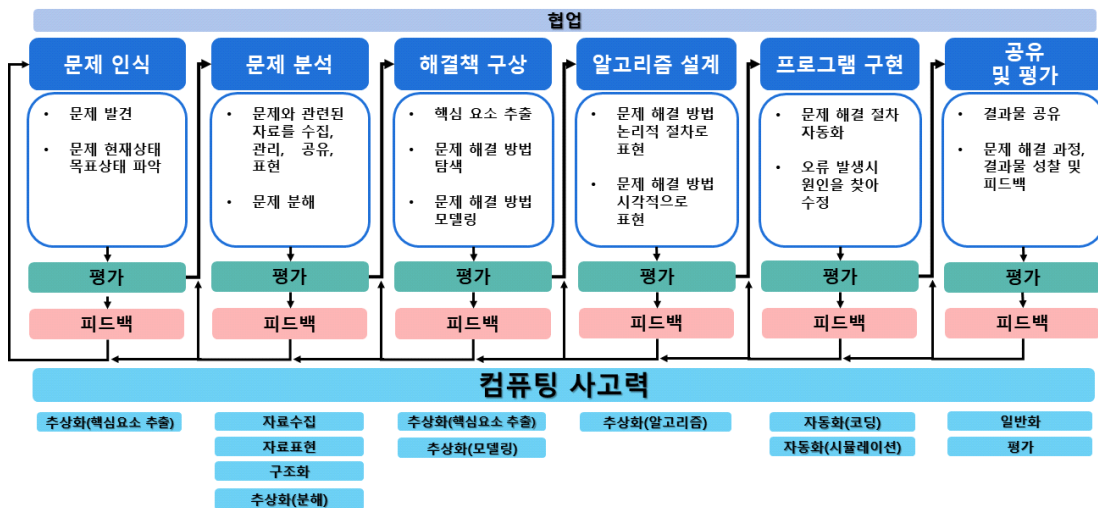
문제 해결 과정의 각 단계에서는 컴퓨팅 사고력의 핵심 요소가 발현되며, 컴퓨팅 사고력 핵심 요소는 교육부가 제시한 컴퓨팅 사고력 구성 요소를 기반으로 개발하여 제시하였다. 각 단계별 핵심 요소는 <표 III-16>과 같다.

<표 III-16> 문제 해결 과정에 적용되는 컴퓨팅 사고력 핵심 요소

문제 해결 과정	컴퓨팅 사고력 핵심 요소	
문제 인식	추상화 (핵심요소 추출)	문제 발견 및 현재 상태와 목표 상태 추출하기

문제 분석	자료 수집	문제와 관련된 자료를 모으기
	자료 분석	자료의 이해, 패턴 찾기, 결론을 도출하기
	구조화	문제를 그래프, 차트, 그림 등으로 시각화하기
	추상화(분해)	문제를 관리 가능한 수준의 작은 문제로 나누기
해결책 구상	추상화 (핵심요소 추출)	문제 해결을 위한 핵심요소를 추출하기
	추상화 (모델링)	문제 해결을 위한 모델 만들기
알고리즘 설계	추상화(알고리즘)	문제를 해결하기 위한 일련의 단계를 알고리즘으로 표현하기(절차적 표현)
프로그램 구현	자동화(코딩)	프로그래밍 언어를 이용해, 문제 해결과정을 자동화하기
	자동화(시뮬레이션)	프로그램(소프트웨어) 실행하기
공유 및 평가	일반화	문제 해결과정을 다른 문제에 적용하기
	평가	결과물을 공유하여 성능을 평가하고 개선할 부분에 대해 논의하기

문제 해결 과정의 각 단계는 유기적이며, 만약 전 단계에서 수정할 사항이 있으면 이동하여 다시 그 과정을 진행할 수 있다. [그림 III-4]는 컴퓨팅 사고력 기반 문제 해결 과정과 적용되는 컴퓨팅 사고력 핵심 요소를 도식화한 그림이다.



[그림 III-4] 본 연구에서 제시하는 컴퓨팅 사고력 기반 문제 해결 과정

4. 문제 해결 과정 성취기준

2015개정 중학교 정보 교과 내용 및 성취기준을 기반으로 개발하고, 2차에 걸친 전문가 검토 및 피드백을 통해 개발된 컴퓨팅 사고력 기반 문제 해결 과정 성취기준은 <표 III-17>와 같다.

<표 III-17> 본 연구에서 제시하는 컴퓨팅 사고력 기반 문제 해결 과정 성취기준

문제 해결 과정	성취기준
문제 인식	01-01 실생활 문제 상황에서 문제를 발견하고 현재 상태, 목표 상태를 이해한다.
문제 분석	02-01 문제와 관련된 자료를 수집, 저장, 관리, 공유하고 표, 다이어그램 등으로 구조화 하여 표현한다.
	02-02 문제의 복잡성을 줄이기 위해 해결 가능한 작은 단위의 문제로 분해한다.
해결책 구상	03-01 문제 해결에 필요한 패턴이나 핵심요소를 추출하고 문제 해결을 위한 아이디어를 제시하거나, 현재의 문제와 유사한 다른 문제의 해결 방법을 탐색한다.
	03-02 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위해 모델링 한다.
알고리즘 설계	04-01 문제 해결을 위한 다양한 방법과 절차를 탐색하고 시각적으로 명확한 알고리즘으로 표현한다.
	04-02 문제 해결 모델에 대한 알고리즘을 제어 구조를 이용하여 논리적으로 설계한다.
프로그램 구현	05-01 설계한 알고리즘을 구현할 프로그래밍 언어를 선택하고, 프로그래밍 언어의 개발 환경 및 특성에 맞게 자동화하여 구현한다.
	05-02 제어 구조, 호출, 변수와 연산자 등을 활용하여 자료를 입력을 받아 처리하고 출력하는 프로그램을 작성한다.
	05-03 오류 발생 시 프로그램이나 알고리즘을 분석하여 오류의 원인을 발견하고 실행 가능한 형태로 수정한다.
공유 및 평가	06-01 문제를 해결함에 있어 역할 분담, 자신의 임무 수행, 팀원 간에 의사소통 등 협업하여 프로그램을 개발 하였는지 확인한다.
	06-02 결과물을 서로 공유 및 평가하고 문제 해결 과정에 대하여 성찰 및 피드백 하는 태도를 취한다.

5. 평가 요소

본 연구에서 활용할 평가 요소는 성취기준을 기반으로 추출하였으며, 2차에 걸친 전문가 검토 및 피드백을 통해 수정된 성취기준 기반 평가 요소는 <표 III-18>과 같다.

<표 III-18> 성취기준 기반 평가 요소

성취기준	평가 요소
01-01 실생활 문제 상황에서 문제를 발견하고 현재 상태, 목표 상태를 이해한다.	01-01-01 실생활 문제 상황에서 문제를 발견할 수 있는가?
	01-01-02 문제의 현재 상태와 목표 상태를 설명할 수 있는가?
02-01 문제와 관련된 자료를 수집, 저장, 관리, 공유하고 표, 다이어그램 등으로 구조화 하여 표현한다.	02-01-01 문제와 관련된 자료를 수집할 수 있는가?
	02-01-02 수집한 자료를 컴퓨터로 저장, 관리, 공유할 수 있는가?
	02-01-03 문제와 관련된 자료를 표, 다이어그램 등으로 구조화하여 표현할 수 있는가?
02-02 문제의 복잡성을 줄이기 위해 해결 가능한 작은 단위의 문제로 분해한다.	02-02-01 문제를 해결 가능한 단위의 문제로 분해할 수 있는가?
03-01 문제 해결에 필요한 패턴이나 핵심요소를 추출하고 문제 해결을 위한 아이디어를 제시하거나, 현재의 문제와 유사한 다른 문제의 해결 방법을 탐색한다.	03-01-01 문제나 자료에서 반복되는 패턴이나 핵심요소를 추출할 수 있는가?
	03-02-02 패턴이나 핵심 요소를 통해 문제 해결을 위한 아이디어를 제시하거나, 현재 문제와 유사한 문제의 해결 방법을 탐색하여 적용할 수 있는가?
03-02 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위해 모델링한다.	03-02-01 문제 해결을 위한 아이디어를 구조화 하고 문제 해결을 위한 모델을 설계할 수 있는가?
04-01 문제 해결을 위한 다양한 방법과 절차를 탐색하고 시각적으로	04-01-01 다양한 문제 해결 방법과 절차를 탐색하여 비교하고 분석할 수 있는가?
	04-01-02 문제 해결 절차를 글, 그림 등을 이용하여

명확한 알고리즘으로 표현한다.	알고리즘으로 명확하게 표현할 수 있는가?
04-02 문제 해결 모델에 대한 알고리즘을 제어 구조를 이용하여 논리적으로 설계한다.	04-02-01 문제 해결 알고리즘을 제어 구조를 이용하여 논리적으로 설계할 수 있는가?
05-01 설계한 알고리즘을 구현할 프로그래밍 언어를 선택하고, 프로그래밍 언어의 개발 환경 및 특성에 맞게 자동화하여 구현한다.	05-01-01 설계한 알고리즘을 프로그래밍 언어의 개발 환경 및 특성에 맞게 자동화하여 구현할 수 있는가?
05-02 제어 구조, 호출, 변수와 연산자 등을 활용하여 자료를 입력을 받아 처리하고 출력하는 프로그램을 작성한다.	05-02-01 제어구조, 호출, 변수, 연산자 등을 사용함에 있어 논리적인 오류 없이 프로그램을 구현할 수 있는가?
	05-02-02 다양한 형태의 자료를 입력 받아 처리한 후 원하는 출력 값을 얻는 프로그램을 구현할 수 있는가?
05-03 오류 발생 시 프로그램이나 알고리즘을 분석하여 오류의 원인을 발견하고 실행 가능한 형태로 수정한다.	05-03-01 오류의 원인을 체계적으로 분석하여 찾아낼 수 있는가?
	05-03-02 오류를 프로그램 실행 가능한 형태로 수정할 수 있는가?
06-01 문제를 해결함에 있어 역할 분담, 자신의 임무 수행, 팀원 간에 의사소통 등 협업하여 프로그램을 개발 하였는지 확인한다.	06-01-01 문제 해결 과정에서 각자의 역할을 나누고, 할당된 과제를 수행하여 과제의 목표를 달성할 수 있는가?
	06-01-02 문제 해결을 위한 아이디어, 의견을 나누고 공유하며 지속적인 의사소통을 할 수 있는가?
06-02 결과물을 서로 공유 및 평가하고 문제 해결 과정에 대하여 성찰 및 피드백 하는 태도를 취한다.	06-02-01 동료들과 결과물을 공유하여 잘된 점, 아쉬운 점 등을 이야기 할 수 있는가?
	06-02-02 문제 해결 과정을 성찰하고 이와 유사한 문제를 찾아 논의하여 피드백 할 수 있는가?

6. 평가 기준

본 연구에서는 학습자의 문제 해결 과정에서 발생하는 행동을 보고 교사가 판단할 수 있도록 행동 동사를 명확하게 기술하고, 평가 기준을 상/중/하 3단계 위계로 나누어 평가 기준을 제시하였다. 2차에 걸친 전문가 검토와 피드백을 통해 수정된 성취기준 기반 평가 기준은 <표 III-19>과 같다.

<표 III-19> 성취기준 평가 기준

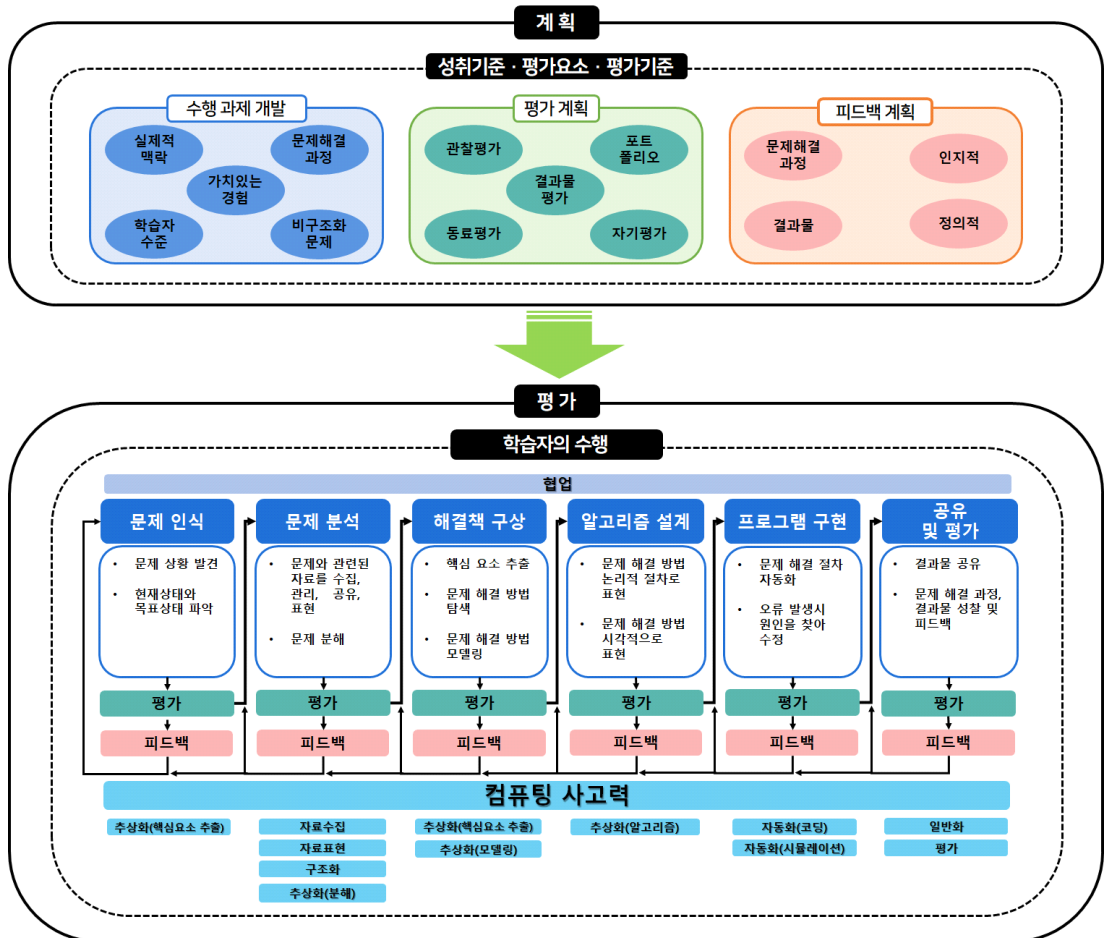
성취 기준	평가 기준	
01-01 실생활 문제 상황에서 문제를 발견하고 현재 상태, 목표 상태를 이해한다.	상	실생활에서 문제를 발견하고 문제 상황에서 현재 상태와 목표 상태를 설명하고 문제가 발생한 원인을 추론할 수 있다.
	중	실생활에서 문제를 발견하고 문제 상황에서 현재 상태와 목표 상태를 설명할 수 있다.
	하	실생활에서 문제를 발견하였지만, 문제 상황에서 현재 상태와 목표 상태를 설명할 수 없다.
02-01 문제와 관련된 자료를 수집, 저장, 관리, 공유하고 표, 다이어그램 등으로 구조화 하여 표현한다.	상	문제와 관련된 자료를 검색하여 자료의 종류와 특성에 맞게 분류하여 컴퓨터에 저장, 관리 및 공유할 수 있고, 수집한 자료를 표, 다이어그램 등 시각적으로 표현하여 문제의 원인을 파악할 수 있다.
	중	문제와 관련된 자료를 검색하여 컴퓨터에 저장, 관리 및 공유할 수 있고, 수집한 자료를 표, 다이어그램 등 시각적으로 표현할 수 있다.
	하	문제와 관련된 자료를 검색하여 컴퓨터에 저장하지 못하거나 표, 다이어그램 등 시각적인 형태로 표현하지 못한다.
02-02 문제의 복잡성을 줄이기 위해 해결 가능한 작은 단위의 문제로 분해한다.	상	문제를 해결 가능한 단위의 하위 문제로 나누어 설명하고, 하위 문제 간의 연결성을 설명할 수 있다.
	중	문제를 해결 가능한 단위의 하위 문제로 나누고, 하위 문제를 설명할 수 있다.
	하	문제를 해결 가능한 단위의 하위 문제로 나누지 못한다.
03-01 문제 해결에 필요한 패턴이나 핵심요소를 추출하고 문제 해결을 위한 아이디어를	상	문제와 자료에서 패턴이나 핵심 요소를 추출하여 문제 해결 방법을 위한 아이디어를 찾아 제시할 수 있고, 문제와 유사한 이전의 문제 해결 방법을 탐색하여 재구성하여 적용할 수 있다.
	중	문제와 자료에서 문제 해결에 필요한 부분을 추출하

제시하거나, 현재의 문제와 유사한 다른 문제의 해결 방법을 탐색한다.		여 문제 해결 방법을 찾거나, 문제와 유사한 이전의 문제 해결 방법을 찾을 수 있다.
	하	문제와 자료를 분석 하였지만 패턴이나 핵심 요소를 찾지 못하거나, 현재의 문제와 유사한 다른 문제의 해결 방법을 찾지 못한다.
03-02 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위해 모델링 한다.	상	분해된 문제를 구조화 하여 표현하고, 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델을 설계할 수 있다.
	중	문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 방법을 말할 수 있다.
	하	문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 방법을 말하지 못한다.
04-01 문제 해결을 위한 다양한 방법과 절차를 탐색하고 시각적으로 명확한 알고리즘으로 표현한다.	상	다양한 문제 해결 방법과 절차를 탐색하고 비교하여 효율적인 절차를 선택하고 글, 그림 등을 이용하여 시각적인 형태로 쉽고 명확하게 표현할 수 있다.
	중	문제 해결 방법과 절차를 시각적인 형태로 쉽고 명확하게 표현할 수 있다.
	하	문제 해결 방법과 절차를 시각적으로 표현하였지만 명확하게 표현하지 못한다.
04-02 문제 해결 모델에 대한 알고리즘을 제어 구조를 이용하여 논리적으로 설계한다.	상	제어 구조를 이용하여 논리적인 알고리즘을 설계할 수 있다.
	중	제어 구조를 이용하여 논리적인 알고리즘을 설계함에 있어 불필요한 부분이 포함되어 있거나, 필요한 부분이 누락되었다.
	하	제어 구조를 이용하여 논리적인 알고리즘을 설계하지 못한다.
05-01 설계한 알고리즘을 구현할 프로그래밍 언어를 선택하고, 프로그래밍 언어의 개발 환경 및 특성에 맞게 자동화하여 구현한다.	상	설계된 알고리즘을 프로그래밍 언어의 개발 환경 및 특성에 맞게 구조적, 논리적으로 동일하게 자동화하여 구현할 수 있다.
	중	프로그래밍 언어의 개발 환경 및 특성에 맞게 구현 하였지만, 설계된 알고리즘 일부분만 구현할 수 있다.
	하	설계된 알고리즘을 자동화하여 구현하지 못한다.
05-02 제어 구조, 호출, 변수와 연산자 등을	상	프로그램의 목적에 맞게 다양한 형태의 자료를 입력 받아 처리하기 위해 순차, 선택, 반복 구조와 호출,

활용하여 자료를 입력을 받아 처리하고 출력하는 프로그램을 작성한다.		변수, 연산자를 논리적으로 사용하여 원하는 결과를 출력하는 프로그램을 구현할 수 있다.
	중	순차, 선택, 반복 구조와 호출, 변수, 연산자를 사용하여 프로그램의 목적에 맞게 구현 하였지만 불필요한 블록이나 객체가 포함되어 있다.
	하	프로그램의 목적에 맞게 자료를 입력 받아 처리하고 결과를 출력하는 프로그램을 구현하지 못하였다.
05-03 오류 발생 시 프로그램이나 알고리즘을 분석하여 오류의 원인을 발견하고 실행 가능한 형태로 수정한다.	상	오류 발생 시 지속적으로 시험을 시도하고, 코드를 확인하는 등 체계적으로 오류의 원인을 찾아내고 실행 가능한 형태로 수정할 수 있다.
	중	무작위적 오류 발견을 통해 오류의 원인을 찾아내고, 오류를 실행 가능한 형태로 수정할 수 있다.
	하	오류가 발생 하였지만 오류의 원인을 찾지 못해 오류를 수정하지 못한다.
06-01 문제를 해결함에 있어 역할 분담, 자신의 임무 수행, 팀원 간에 의사소통 등 협업하여 프로그램을 개발 하였는지 확인한다.	상	문제 해결 과정에서 역할 분담을 하였으며, 서로의 과제를 확인하여 공통의 목표를 달하고, 지속적으로 아이디어나 의견을 공유하여 의사소통 하였다.
	중	문제 해결 과정에서 팀원 간의 의견을 공유가 되지 않아, 불균형 적인 역할 분담으로 한쪽에 과중된 과제가 주어진 상태에서 프로그램을 개발하였다.
	하	문제 해결 과정에서 역할 분담이 되지 않고, 팀원 간의 의견을 공유하지 않았다.
06-02 결과물을 서로 공유 및 평가하고 문제 해결 과정에 대하여 성찰 및 피드백 하는 태도를 취한다.	상	동료 평가를 통해 결과물의 장단점과 개선점에 대해 적극적으로 의견을 제시하고 있으며, 동료의 피드백과 자기평가를 바탕으로 자신의 문제 해결 과정을 되돌아보며 성찰하고 피드백 하는 태도를 취한다.
	중	동료 평가를 통해 결과물의 장단점과 개선점에 대해 간단한 의견을 제시하고 있지만, 동료의 피드백과 자기평가를 바탕으로 성찰하는 태도가 미흡하다.
	하	동료 평가에서 결과물에 대한 의견을 제시하지 않고, 자신의 문제 해결 과정을 성찰하지 않는다.

7. 문제 해결 과정 중심의 컴퓨팅 사고력 평가 모델

본 연구는 학습자가 컴퓨팅 사고력을 기반으로 문제를 해결하는 과정에서 과정 중심평가 방식을 적용하여 컴퓨팅 사고력을 평가하기 위한 평가 모델을 [그림 III-5]와 같이 제시한다.



[그림 III-5] 문제 해결 과정 중심 컴퓨팅 사고력 평가 모델

평가자는 본 논문에서 개발한 성취기준을 기반으로 수행 과제를 개발하고, 평가 계획, 피드백 계획을 세운다. 수행 과제는 학습자의 수준을 고려하여 수행 과제가 실생활 문제 상황에서 수행 능력을 평가할 수 있도록 개발하며, 문제가 컴퓨터로 문제를 해결 할 수 있는지 고려해야 한다. 또한 학습자가 수행 평가에 참여하는 동안 인지적·정의적 측면에서 긍정적이고 가치 있는 경험이 되어야 하며, 문제를 해

결하기 위한 다양한 시도를 할 수 있는 형태로 개발되어야 한다. 평가자는 학습자의 참여 동기와 흥미, 교육 환경 등을 고려하여 교수·학습 계획을 세운다. 2015개정 교육과정 제시하는 학습자 수행과제 개발에서 고려해야할 사항을 참고한 수행 과제 개발 체크리스트는 <표 III-20>와 같다.

<표 III-20> 수행 과제 개발 체크리스트

항목	내용	예	아니오
교육과정 및 성취기준 적합성	• 학습자의 수준이나 범위에서 벗어나지 않는가?		
	• 성취기준에서 의도하는 능력이 해당 과제로 타당하게 평가될 수 있는가?		
고등 사고 능력의 평가	• 수행 과제가 분석력, 종합력, 비판력 등의 종합적인 고등 사고 능력을 평가 하는데 적절한가?		
	• 수행 과제가 학생이 경험할 수 있는 실제적인 상황을 반영하고 있는가?		
과제의 공정성	• 수행 과제가 성별, 지역, 문화적인 측면에서 특정 학습자에게 유리하거나 불리하지 않는가?		
참여 동기 및 흥미 고려	• 수행 과제가 학습자의 능력을 충분히 발휘할 수 있도록 학습자의 동기와 흥미를 고려한 것인가?		
과제의 실행 가능성	• 수행 과제는 교육 환경을 고려할 때 충분히 실행 가능한가?		
	• 수행 과제는 컴퓨터를 통해 해결할 수 있는 과제인가?		

평가 계획은 본 논문에서 개발한 평가 요소와 평가 기준을 고려하여 지식, 기능 태도의 구체적인 평가 요소, 평가 척도로 사용하고 채점 기준을 세운다. 평가 기준은 학습자에게 미리 안내되어야 하며 학습자의 문제 해결 과정에서 관찰평가, 자기평가, 포트폴리오 등의 방법으로 평가를 진행하고 수행을 통한 결과물에 대해 동료평가, 결과물 평가 등의 평가 계획을 수립한다.

피드백 계획은 학습자의 수행 과정에 대한 관찰과 기록을 통해 진행한다. 학습자

에게 부족한 점은 무엇인지, 보완할 수 있는 부분은 무엇인지 기록과 관찰을 통해 추출하고 적절한 피드백을 주어야 한다. 피드백은 현재 학습자의 수준과 도달해야 할 수행 수준 간의 차이를 자세하게 알려주어 학습의 성장을 지원한다. 피드백은 학습자에 문제 해결 과정과 수행의 결과에 대해 정의적, 인지적인 측면에서 이루어져야 한다.

문제 해결 과정 중심 컴퓨팅 사고력 평가를 위한 교수·학습 과정은 문제를 발견하는 것으로부터 시작한다. 실생활에서 겪을 수 있는 문제 상황에서 문제를 인식하고 문제의 원인을 발견하는 능력을 기른다.

문제 분석 단계에서는 문제와 관련한 자료를 인터넷, 인터뷰 등 다양한 방법으로 수집하고 팀원과 공유한다. 이 때 팀원 간의 역할분담을 통해 효율적으로 자료를 수집하고 관리, 표현하도록 한다. 또한 팀원과 소통을 통해 문제를 분석하여 문제를 해결 가능한 단위의 문제로 분해하는 활동을 한다.

해결책 구상 단계에서는 문제 해결을 위한 핵심요소 또는 반복되는 패턴을 찾고 해결아이디어를 구상하기 위해 브레인스토밍, 마인드맵 등 다양하게 생각할 수 있는 환경을 만들어 주고, 해결해야 할 문제와 비슷한 문제 해결 과정을 검색 및 탐색하여 찾고 적용할 수 있도록 한다. 아이디어 및 해결책을 바탕으로 컴퓨터로 자동화하여 구현할 수 있도록 문제 해결 모델을 설계한다.

알고리즘 설계 단계에서는 문제 해결 모델을 기반으로 자동화할 스토리 보드를 작성하고, 알고리즘을 시각적으로 표현해 보는 활동을 가진다. 이 때, 역할분담을 하여 활동을 진행하도록 한다.

프로그램 구현 단계에서는 시각적으로 표현한 스토리 보드 및 알고리즘을 자동화 하여 프로그램으로 구현하는 활동을 가진다. 팀 활동의 경우 하나의 프로그램을 만들며 한 명이 주도적으로 프로그래밍 할 경우가 생기므로, 알고리즘 설계에서 충분한 대화를 거친 이후 프로그램 구현 단계에서는 각자 프로그래밍 하며, 구현 과정에서 지속적으로 의사소통 하도록 유도한다.

공유 및 평가 단계에서는 팀 별로 만든 작품을 자신의 클래스에 업로드 하여 공유하거나 발표하는 활동을 가진다. 팀별로 잘된 점, 부족한 점을 이야기하고, 결과물을 피드백 하여 수정하며, 자신의 문제 해결 과정을 되돌아보고 이야기하는 시간을 가진다.

문제 해결 과정의 각 단계에서 학습자의 컴퓨팅 사고력에 대한 자기 평가, 관찰 평가, 포트폴리오 등을 통해 평가가 이루어지며 평가 과정에서 평가자는 학습자의 부족한 부분에 대하여 적절하게 피드백 해주어 학습자의 성장을 돕는다.

IV. 결론

소프트웨어 교육의 목적인 컴퓨팅 사고력을 어떻게 평가할 것인가는 매우 중요한 문제이다. 최근 평가의 방향은 단순히 지식의 습득과 결과를 중심으로 평가하는 것이 아닌, 습득한 지식 기반의 문제 해결 능력과 수행 능력을 평가하기 위해 학습자가 문제를 해결해 나가는 과정에서 평가를 진행하고 있다. 컴퓨팅 사고력 또한 컴퓨터 지식을 기반으로 문제를 해결해 나가는 역량으로, 문제 해결 과정에서 평가가 이루어져야 한다. 이를 위하여 본 연구는 학습자의 컴퓨팅 사고력을 문제를 해결하는 과정 중심에서 평가하기 위하여 과정중심 수행평가 방법을 활용한 모델을 제시하였다.

본 연구를 통해 얻은 시사점은 다음과 같다.

첫째, 컴퓨팅 사고력은 컴퓨터 과학의 기본 원리 및 지식을 토대로 일련의 문제를 해결해 나가는 과정이므로 컴퓨팅 사고력의 학습 및 평가는 학습자의 문제 해결 과정에서 함께 이루어지는 것이 바람직하다. 이를 위해 문제 해결 관점에서의 컴퓨팅 사고력 연구와 평가 연구가 이루어져야 한다.

둘째, 기존의 지필 및 결과중심의 평가가 아닌 학생의 수행과정에서 평가하고 평가 과정에서 교사의 적절한 피드백을 통해 학생의 성장을 도와야 한다. 이를 위해 평가 과정에서 학습자의 컴퓨팅 사고력을 향상시킬 수 있는 방안에 대한 연구가 이루어져야 한다.

셋째, 협력적 문제해결에 대한 연구가 활발히 이루어져야 한다. 2015 정보교과 교육과정에서 제시하는 정보교과의 핵심역량은 ‘정보문화소양’, ‘협력적 문제해결력’, ‘컴퓨팅 사고력’이다. 이처럼 협력적 문제해결력의 중요성과 당위성은 강조되고 있으나, 그 적용방법과 평가에 대한 구체적인 방법은 부족한 실정이다. 본 논문 또한 협력적 문제해결의 적용을 위해 PISA에서 제시한 핵심역량을 제시하고 있지만 구체적인 적용과 평가방법에 대한 연구가 이루어져야 할 필요가 있다.

본 연구는 문제 해결 과정 중심의 컴퓨팅 사고력 평가를 위해 컴퓨팅 사고력 기반의 ‘문제 해결 과정’, ‘문제 해결 과정 성취기준’, ‘평가 요소’, ‘평가 기준’을 개

발하고 2차에 걸친 전문가 델파이조사를 통해 수정 및 보완하여 제시하였다. 또한 개발한 내용들의 타당도 검사를 진행하여 타당도를 확보하였다.

본 연구를 통해 기대할 수 있는 효과는 다음과 같다.

첫째, 결과 중심의 컴퓨팅 사고력 평가가 아닌 문제 해결 과정 중심의 컴퓨팅 사고력 평가이다. 학습자가 컴퓨팅 사고력을 기반으로 문제를 해결해 나가는 과정에서 인지적, 정의적으로 학습자의 컴퓨팅 사고력 수준을 평가할 수 있을 것이다.

둘째, 교육현장에 적용이다. 최근 교육부가 과정중심평가와 수행평가를 강조하며 학교 현장에서 과정중심 수행평가를 진행하기 위해 노력하고 있다. 현장에서는 평가의 준거가 교사에 의해 개발되기 때문에 교사의 역량에 따라 평가의 결과가 달라질 수 있다. 본 연구를 활용하여 좀 더 객관적인 과정중심 수행평가가 이루어질 것 기대한다.

셋째, 기존 컴퓨팅 사고력 평가 도구와 함께 사용하여 좀 더 객관적으로 학습자의 부족한 부분을 찾아내고 평가할 수 있을 것이다. 컴퓨팅 사고력을 평가하기 위한 다양한 방법들이 존재하며, 각 방법들은 장단점을 가지고 있다. 하나의 방법만으로 학습자의 수준을 파악하는 것이 아닌 다양한 방법을 적용하여 적용한다면 좀 더 정확하게 학습자의 수준을 평가할 수 있을 것이다.

넷째, 평가를 통한 컴퓨팅 사고력 향상이다. 학습자는 컴퓨팅 사고력 기반 문제 해결 과정을 경험하고 직접 문제를 해결하여, 그 과정에서 평가자에게 부족한 부분을 피드백 받음으로써 컴퓨팅 사고력이 향상될 수 있을 것이다.

본 연구를 바탕으로 후속 연구를 위한 제언은 다음과 같다.

첫째, 본 연구에서 개발한 내용은 제1급 감염병인 코로나19로 인해 사회적 거리두기 및 모임 자제로 학생들과 함께 수업을 진행하여 적용하기 어려운 부분이 있었다. 추후 코로나19에 대한 감염 경보가 풀린 후 적용하여 그 효과성을 검증할 필요가 있다.

둘째, 평가기준의 수정 및 보완이다. 본 논문에서 개발한 평가기준은 기존 문헌 분석과 연구를 바탕으로 개발된 성취기준을 통해 개발되었다. 블룸의 디지털 텍사노미를 활용한 행동동사로 기술하고, 2차에 걸친 전문가 피드백 과정을 거치며 평가의 객관성과 타당성을 확보하였지만 추후 현장에 적용을 하며 수정 및 보완이 이루어져야 한다.

셋째, 본 연구는 2015개정 중학교 교육과정 정보교과를 기준으로 개발하였기 때문에 내용 및 평가 기준이 중학교 정보교과에 맞춰져 있다. 중학교 정보교과 이하 또는 이상 수준의 학습자 컴퓨팅 사고력을 객관적으로 평가하기에 한계가 있다. 따라서 초등 기술·가정교과 교육과정, 고등 정보교과 교육과정을 분석하여 각 수준에 맞도록 성취기준, 평가 요소, 평가 기준을 수정하여 개발되어야 할 필요성이 있다.

[참고 문헌]

- 강오한 (2019). 정보·컴퓨터 중등교사 임용시험에서 프로그래밍 실기 평가의 개선 방안. 한국컴퓨터교육학회 논문지. 22(6), 1-9.
- 교육부 (2015). 2015개정 정보과 교육과정. 교육부.
- 교육부 (2015). 2015개정교육과정 총론. 교육부.
- 교육부, KERIS (2015). 소프트웨어 운영지침 개발 정책 연구. 교육부.
- 교육부 (2017). 과정을 중시하는 수행평가 어떻게 할까요?. 한국교육과정평가원.
- 김경훈 (2016). 2015개정 교육과정에 따른 정보과 평가 기준 개발 연구. 한국 교육과정평가원.
- 김국현 (2009). 성취기준의 개념 혼란과 도덕과 성취기준 및 평가기준의 설정 원리와 방법. 교육과정평가연구. 12(3), 125-151.
- 김미송 (2018). 초등예비교사를 위한 컴퓨팅 사고력 자기평가 방법. 정보교육 회 논문지. 22(1), 61-70.
- 김민자 (2016). 비전공자 프로그래밍 수업 창의적 산출물의 컴퓨팅 사고력 기반 평가 루브릭 개발. 한국컴퓨터교육학회 논문지. 20(2), 1-9.
- 김민정 (2017). 컴퓨팅사고력 측정에 사용되고 있는 도구분석을 통한 새로운 검사 도구 개발방향 제시. 한국컴퓨터교육학회 논문지. 20(6), 17-25.
- 김성숙 외 (2018). 우리나라 학생들의 PISA 2015 협력적 문제해결력과 협력적 태도에 대한 관계 비교. 교육과정평가연구. 21(3), 155-179.
- 김순남 외 (2013). 창의인재 육성을 위한 학생평가 정책 연구. 한국교육개발원.
- 노승용 (2006). 델파이 기법 : 전문적 통찰로 미래예측하기. 국토연구원.
- 미래창조과학부 (2014). SW중심사회 실현전략.
- 박주연 (2017). 초·중등 SW교육의 평가요소 개발. 한국컴퓨터교육학회 논문지. 20(6), 47-59.
- 박혜영 외 (2014). 협력적 문제해결력 교수·학습 및 평가를 위한 PISA와 ATC21S의 특징 비교 분석. 학습자중심교과교육연구. 14(9), 439-462.
- 서인철 (2009). Pair Programming을 이용한 프로그래밍 수업이 학업 성취도 및 교우관계에 미치는 영향. 건국대학교 교육대학원, 석사학위논문.

- 성태제 (2014). 현대교육평가(4판). 서울:학지사.
- 세계경제포럼 (2018). The Future of Jobs Reprot 2018.
- 오마에 겐이치 (2005). 맥킨지 문제 해결의 기술 (김영철 역). 일빛 출판.
- 유지수 (2019). 초등학생을 위한 문제해결력 기반 컴퓨팅 사고력 평가 문항 개발 및 적용. 서울교육대학교 교육전문대학원.
- 이민우, 김성식 (2019). 문제해결 프로그래밍 교육을 위한 범용 컴퓨팅 사고력 척도 개발 연구. 한국컴퓨터교육학회 논문지. 22(5), 67-77.
- 이영준 외 (2014). 초중등 단계 Computational Thinking 도입을 위한 기초 연구. 한국과학창의재단.
- 이은경 (2019). 한국과 미국의 교육과정에 나타난 컴퓨팅 사고력 관련 성취기준 비교 분석. 한국컴퓨터교육학회 논문지, 22(3), 15-23.
- 이은민 (2016). 정보통신정책연구원_4차 산업혁명과 산업구조의 변화. 정보통신 정책연구.
- 이종성 (2001). 델파이방법. 서울:교육과학사.
- 이철현 (2016). 소프트웨어 교육을 위한 컴퓨팅 사고력 기반 문제 해결 모형 (CT-PS Model) 개발. 한국실과교육연구학회, 22(3), 97-117.
- 이태욱 외 (2016). 정보교과교육론. 한빛아카데미.
- 임종헌, 최원석 (2018). ‘과정 중심 평가’의 특징과 의미에 관한 연구: 자유학기제 ‘과정 중심 평가’를 중심으로. 한국교육, 45(3), 31-59.
- 임은영 (2017). 과정 중심 평가의 개념과 의미. https://happyedu.moe.go.kr/happy/bbs/selectHappyArticleImg.do?bbsId=BBSMSTR_000000000192&nttId=7238에서 2019. 11. 05 인출
- 전경희 (2016). 과정중심 수행평가의 방향과 과제. 한국교육개발원.
- 전용주 (2017). 새로운 교육과정의 소프트웨어 교육을 위한 컴퓨팅 사고력 기반 창의적 문제해결(CT-CPS) 수업모형의 개발 및 적용. 박사학위논문, 한국교원대학교 대학원.
- 주여진 (2018). 컴퓨팅 사고력 측정을 위한 추상화 역량 평가 기준 개발. 정보교육학회논문지, 22(3), 375-383.
- 최숙영 (2016). 문제해결의 관점에서 컴퓨팅 사고력 증진을 위한 교수학습에 대

- 한 연구. 한국컴퓨터교육학회 논문지, 19(1), 53-62.
- 최숙영 (2019). 컴퓨팅 사고력 평가에 관한 시스템 매핑 기반 국내 문헌 고찰. 한국컴퓨터교육학회 논문지, 22(6), 19-33.
- 최정원, 이영준 (2014). 컴퓨팅 사고력 평가 방안 설계. 한국컴퓨터정보학회 학술 발표논문집, 22(2), 177-178.
- 최형신 (2014). Computational Thinking역량 계발을 위한 수업 설계 및 평가 루브릭 개발. 한국정보교육학회 논문지, 18(1), 57-64.
- 행정안전부 지방행정연수원 (2008). 문제 해결 역량 과정.
- ISTE & CSTA (2011). Computational Thinking Leadership Toolkit 1st edition.
- Lawshe (1975). A QUANTITATIVE APPROACH TO CONTENT VALIDITY.
- Osborn, A. F. (1953). Applied imagination: Principles and procedures of creative thinking. NewYork: Charles Scribner's Sons
- OECD (2017a). PISA 2015 results (volume V): Collaborative problem solving. Paris: OECD Publishing.
- Polya, G. (1986). 어떻게 문제를 풀 것인가? (우정호 역). 서울:(주)천재교육(원본출판:1956).
- Wing M. Jeannette (2006). Computational Thinking. COMMUNICATIONS OF THE ACM March 2006, 49(3).

[ABSTRACT]

Problem Solving Process-Oriented Computational Thinking Assessment

JeongHun Lee

Graduate School of Education Jeju National University
(Major in Computer Education)

Supervised by Professor Jungwon Cho

The Ministry of Science, ICT and Future Planning declared that software has become central to innovation, growth, and value creation and has arrived software-driven society that affects individuals, businesses, and countries. So the proportion of software has increased in future jobs and core competencies of future talent, the importance of software education has also expanded. Korea emphasized the importance of software education in the 2015 revised curriculum and designated information subjects as compulsory subjects. The purpose of software education is to improve computational thinking that solves problems based on the basic principles of computer science. As software education grows in importance, there are active discussions about how to foster computational thinking, but there is relatively little discussion about how to evaluate computational thinking.

How to assess computing power is a very important issue. As assessments are areas where grasped self-learning levels and complemented and improved the deficiencies through self-reflection, assessments in software should enable them to improve computational thinking through assessments. Unlike previous evaluations, the direction of the recent evaluation has been changed from the evaluation of

learning results to the evaluation for learning. Into the direction of this changed assessment, 'The process-Focused Assessment' is rising up. 'The process-Focused Assessment' is an evaluation direction that strengthens the assessment that puts emphasis the course of learning, so that students reflect on their learning through perform, and uses the evaluation to give feedback to students. The evaluation that best reflects this is a 'fair-based performance evaluation' method that assesses how well and how individuals perform the knowledge acquired in real life or in evaluation situations. Computational thinking ability is the ability to be expressed in the problem solving process, and the evaluation of computational thinking ability should also be centered on the student problem solving process.

This thesis aims to utilize 'The Process-Focused Assessment' method for evaluating the computational thinking that is revealed in the problem-solving process of students.

For this, this study was conducted in the following procedures.

First, to define the computational thinking-based problem-solving course and the key elements of computational thinking, the relevant research trends, literature and prior research was analyzed, and compared and arranged, and the computational thinking-based problem-solving course was defined.

Second, in order to apply 'The Process-Focused Assessment' method, the achievement standards for computational thinking-based problem-solving processes were defined. The achievement standards were developed by analyzing that focus on the revised curriculum in 2015 informatics subjects.

Third, the evaluation elements were extracted by disassembling and analyzing the developed achievement standards into learning and evaluation contents. Evaluation elements are the basis for observing,

evaluating, and feedback the learners.

Fourth, to present practical criterion to assess what level a student has reached in the assessment activity presented the assessment standards. The assessment standards are divided into three phases upper/mid/lower and behavioral verbs are used.

Problem-solving process-centered computational thinking assessment was aimed design the teaching and learning contents and evaluation method based on the developed achievement standards. And it aimed evaluate the learner observation, the resulting product, and portfolio in various ways based on assessment factors and assessment standards in the problem-solving process of learner, and do the feedback of cognitive and emotional during the assessment process.

Developed content was modified and supplemented after two rounds of Delphi surveys by a group of 10 experts, and the validity was demonstrated through a validity assessment.

If you refer to the achievement standards, assessment factors, and assessment standards developed in this thesis, it will be easier to evaluate the computational thinking at the problem-solving process-centered and design the teaching and learning assessment. Based on this, I hope to contribute to the improvement and evaluation of computational thinking.

[부록]

1. 중학교 컴퓨팅 사고력 과정평가 모델 차시 별 교수·학습 과정안

< 1~2차시 지도안 >

학습 목표	<ul style="list-style-type: none"> ◦ 실생활 문제를 인식하여 문제의 현재 상태와 목표 상태를 파악하여 말할 수 있다. ◦ 문제와 관련된 자료를 수집, 관리, 저장, 공유, 분석하여 시각각적으로 구조화하여 표현하고, 문제의 복잡성을 줄이기 위해 문제를 해결 가능한 단위의 문제로 분해할 수 있다. ◦ 문제의 패턴이나 핵심 요소를 추출하여 문제 해결을 위한 다양한 방법에 대한 구상하고, 컴퓨터로 구현하기 위한 문제 해결 모델을 설계할 수 있다.
------------------	--

평가 계획	성취 기준	<ul style="list-style-type: none"> ■ 실생활 문제 상황에서 문제를 발견하고 현재 상태, 목표 상태를 이해한다. ■ 문제와 관련된 자료를 수집, 저장, 관리, 공유하고 표, 다이어그램 등으로 구조화 하여 표현한다. ■ 문제의 복잡성을 줄이기 위해 해결 가능한 작은 단위의 문제로 분해한다. ■ 문제 해결에 필요한 패턴이나 핵심 요소를 추출하고 문제 해결을 위한 아이디어를 제시하거나, 현재의 문제와 유사한 다른 문제의 해결 방법을 탐색한다. ■ 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위해 모델링 한다. 	
	평가 기준	평가 요소	평가 기준
	<ul style="list-style-type: none"> • 실생활 문제 상황에서 문제를 발견할 수 있는가? • 문제의 현재 상태와 목표 상태를 설명할 수 있는가? 	상	실생활에서 문제를 발견하고 문제 상황에서 현재 상태와 목표 상태를 설명하고 문제가 발생한 원인을 추론할 수 있다.
		중	실생활에서 문제를 발견하고 문제 상황에서 현재 상태와 목표 상태를 설명할 수 있다.
		하	실생활에서 문제를 발견하였지만, 문제 상황에서 현재 상태와 목표 상태를 설명할 수 없다.
	<ul style="list-style-type: none"> • 문제와 관련된 	상	문제와 관련된 자료를 검색하여 자료

	<ul style="list-style-type: none"> 자료를 수집할 수 있는가? 수집한 자료를 컴퓨터로 저장, 관리, 공유할 수 있는가? 문제와 관련된 자료를 표, 다이어그램 등으로 구조화하여 표현할 수 있는가? 		의 종류와 특성에 맞게 분류하여 컴퓨터에 저장, 관리 및 공유할 수 있고, 수집한 자료를 표, 다이어그램 등 시각적으로 표현하여 문제의 원인을 파악할 수 있다.
		중	문제와 관련된 자료를 검색하여 컴퓨터에 저장, 관리 및 공유할 수 있고, 수집한 자료를 표, 다이어그램 등 시각적으로 표현할 수 있다.
		하	문제와 관련된 자료를 검색하여 컴퓨터에 저장하지 못하거나 표, 다이어그램 등 시각적인 형태로 표현하지 못한다.
	<ul style="list-style-type: none"> 문제를 해결 가능한 단위의 문제로 분해할 수 있는가? 	상	문제를 해결 가능한 단위의 하위 문제로 나누어 설명하고, 하위 문제 간의 연결성을 설명할 수 있다.
		중	문제를 해결 가능한 단위의 하위 문제로 나누고, 하위 문제를 설명할 수 있다.
		하	문제를 해결 가능한 단위의 하위 문제로 나누지 못한다.
	<ul style="list-style-type: none"> 문제나 자료에서 반복되는 패턴이나 핵심요소를 추출할 수 있는가? 패턴이나 핵심 요소를 통해 문제 해결을 위한 아이디어를 제시하거나, 현재 문제와 유사한 문제의 해결 방법을 탐색하여 적용할 수 있는가? 	상	문제와 자료에서 패턴이나 핵심 요소를 추출하여 문제 해결 방법을 위한 아이디어를 찾아 제시할 수 있고, 문제와 유사한 이전의 문제 해결 방법을 탐색하여 재구성하여 적용할 수 있다.
		중	문제와 자료에서 문제 해결에 필요한 부분을 추출하여 문제 해결 방법을 찾거나, 문제와 유사한 이전의 문제 해결 방법을 찾을 수 있다.
		하	문제와 자료를 분석 하였지만 패턴이나 핵심 요소를 찾지 못하거나, 현재의 문제와 유사한 다른 문제의 해결 방법을 찾지 못한다.

		<ul style="list-style-type: none"> 문제 해결을 위한 아이디어를 구조화 하고 문제 해결을 위한 모델을 설계할 수 있는가? 	상	분해된 문제를 구조화 하여 표현하고, 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델을 설계할 수 있다.
			중	문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 방법을 말할 수 있다.
			하	문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 방법을 말하지 못한다.
평가 시기	수업 중, 수업 후			
평가 방법	자기평가, 관찰평가, 포트폴리오			
산출물	학습지, 자료 저장 파일			

단계	학습 요소	교수.학습 활동	자료 및 유의점
도입 (10분)	동기유발 (전체)	■ 스마트폰 과의존 슬롬비 사고 문제에 대한 뉴스 시청 <ul style="list-style-type: none"> 뉴스 시청 후 이에 대하여 이야기해보기 <ul style="list-style-type: none"> - 어떤 내용인가? - 슬롬비 사고에 대한 문제를 예방할 수 있을까? 	♣ 참고 동영상 https://www.yna.co.kr/view/MYH20180912003200038 ♣ 슬롬비로 인한 사고 급증 뉴스를 보며 생활 속 문제에 대한 호기심을 갖도록 유도하며 문제의 원인에 대하여 생각해 보도록 한다.
	문제확인 (전체)	■ 문제 알아보기 <div style="border: 1px dashed black; padding: 5px; text-align: center;"> 스마트폰 과의존에 의한 슬롬비 사고 문제를 해결해 봅시다. </div>	
	학습 목표 안내 (전체)	■ 학습 목표 확인 <div style="border: 1px dashed black; padding: 5px;"> <ul style="list-style-type: none"> 실생활 문제를 인식하여 문제의 현재 상태와 목표 상태 및 문제의 원인을 파악하여 말할 수 있다. 문제와 관련된 자료를 수집, 관리, 공유, 분석하여 시각각적으로 구조화하여 표현하고, 문제의 복잡성을 줄이기 위해 문제를 해결 가능한 단위의 문제로 분해할 수 있다. 문제의 패턴이나 핵심 요소를 추출하여 문제 </div>	

		<p>해결을 위한 다양한 방법에 대한 구상하고, 컴퓨터로 구현하기 위한 문제 해결 모델을 설계할 수 있다.</p> <p>■ 모둠 구성하기 ○ 2인 1조로 모둠 구성하기</p>	
전개1 (10분)	문제인식 (모둠)	<p>■ 문제를 소프트웨어로 해결하기 위해 정의하기</p> <ul style="list-style-type: none"> ○ 문제의 원인 파악하기 ○ 현재 상태와 목표 상태 확인하기 ○ 목표 상태로 도달하기 위해 소프트웨어를 이용한 해결방법 토의하기 ○ 최종 문제 정의 ex) <ul style="list-style-type: none"> - 스마트폰 과의존 예방 애니메이션을 만들어 슬픔비 사고 문제를 해결하겠다. - 스마트폰 강제 종료 프로그램을 만들어 슬픔비 사고 문제를 해결하겠다. 	<p>♣다양한 아이디어가 나올 수 있도록 분위기 조성</p>
	학습활동 안내 (전체)	<p style="text-align: center;">【주요 학습 활동】</p> <p>[활동1] 문제 해결을 위한 자료 수집, 관리하기 [활동2] 수집한 자료 공유, 분석, 표현하기 [활동3] 문제의 복잡성을 줄이기 위해 문제 분해하기</p>	
	활동1 (모둠)	<p>■ [활동1] 문제 해결을 위한 자료 수집, 관리하기</p> <ul style="list-style-type: none"> ○ 문제 해결을 위해 필요한 자료가 무엇인지 생각해 보기 <ul style="list-style-type: none"> - 문제의 현황, 원인, 해결에 필요한 자료나 정보 웹 검색 - 필요한 자료나 정보 저장, 관리하기 	<p>♣역할 분담을 하여 직접 웹 검색을 통해 필요한 자료나 정보를 수집하도록 한다. (자료 수집, 관리 활동에 대한 관찰 평가)</p>
전개2 (40분)	활동2 (모둠)	<p>■ [활동2] 수집한 자료 공유, 분석, 표현하기</p> <ul style="list-style-type: none"> ○ 수집한 자료를 모둠별로 공유하기 ○ 자료나 정보를 분석하여 필요에 맞게 시각적으로 표현하기 <ul style="list-style-type: none"> - 표, 다이어그램, 그림 등으로 구조화 하여 표현 	<p>♣자료의 표현 방식을 예를 들어 설명하고 자료의 종류와 특성에 맞게 표현하도록 유도, 피드백 (자료 공유, 분석, 표현에 대한 관찰 평가)</p>

<p>활동3 (모둠)</p>	<p>■ [활동3] 문제의 복잡성을 줄이기 위해 문제 분해하기</p> <ul style="list-style-type: none"> ○ 문제 해결을 위해 소프트웨어로 구현 했을 때, 기능이나 순서를 구분해서 토의해보기 <p>ex)</p> <ul style="list-style-type: none"> - 애니메이션의 경우 꼭 들어가야 할 장면, 각 장면의 기능 또는 내용 표기 - 프로그램의 경우 만들고자 하는 프로그램의 기능을 나누어 생각해 보기 <ul style="list-style-type: none"> ○ 마인드맵, 그림 등으로 자유롭게 표현하기 <p>■ 자기 평가</p> <table border="1" data-bbox="488 909 1126 1503"> <thead> <tr> <th></th> <th>아니다</th> <th>보통이다</th> <th>그렇다</th> </tr> </thead> <tbody> <tr> <td>나는 문제의 원인과 함께 현재 상태와 목표 상태를 설명 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 소프트웨어로 문제를 해결하기 위한 방법을 제시 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 문제 해결을 위한 자료를 수집 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 수집한 자료를 컴퓨터에 저장하고, 사용하기 쉽게 응용 소프트웨어를 이용하여 관리 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 수집한 자료를 친구와 공유 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 수집한 자료를 분석하여 표, 그림 등으로 보기 쉽게 표현 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 기존의 문제를 해결 가능한 작은 문제로 분해하였다.</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		아니다	보통이다	그렇다	나는 문제의 원인과 함께 현재 상태와 목표 상태를 설명 하였다.				나는 소프트웨어로 문제를 해결하기 위한 방법을 제시 하였다.				나는 문제 해결을 위한 자료를 수집 하였다.				나는 수집한 자료를 컴퓨터에 저장하고, 사용하기 쉽게 응용 소프트웨어를 이용하여 관리 하였다.				나는 수집한 자료를 친구와 공유 하였다.				나는 수집한 자료를 분석하여 표, 그림 등으로 보기 쉽게 표현 하였다.				나는 기존의 문제를 해결 가능한 작은 문제로 분해하였다.				<p>♣문제 분해를 예를 들어 설명하고 자유로운 의견이 나올 수 있도록 토의 분위기를 조성한다.</p> <p>♣분해한 문제를 구조화하여 분해 문제 사이의 관계를 표현할 수 있도록 도와주고, 모둠을 돌아다니면서 학생들의 문제 분해 의견을 들어보고 피드백 해준다.(문제 분해 활동에 대한 관찰 평가)</p> <p>♣평가 요소를 토대로 자기 평가를 재구성하여 제시한다.</p>
	아니다	보통이다	그렇다																															
나는 문제의 원인과 함께 현재 상태와 목표 상태를 설명 하였다.																																		
나는 소프트웨어로 문제를 해결하기 위한 방법을 제시 하였다.																																		
나는 문제 해결을 위한 자료를 수집 하였다.																																		
나는 수집한 자료를 컴퓨터에 저장하고, 사용하기 쉽게 응용 소프트웨어를 이용하여 관리 하였다.																																		
나는 수집한 자료를 친구와 공유 하였다.																																		
나는 수집한 자료를 분석하여 표, 그림 등으로 보기 쉽게 표현 하였다.																																		
나는 기존의 문제를 해결 가능한 작은 문제로 분해하였다.																																		
<p>학습활동 안내 (전체)</p>	<p style="text-align: center;">【주요 학습 활동】</p> <div style="border: 1px solid black; padding: 5px;"> <p>[활동4] 문제 해결을 위한 핵심 요소 추출하기</p> <p>[활동5] 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델 설계하기</p> </div>																																	
<p>활동4 (모둠)</p>	<p>■ [활동4] 문제 해결을 위한 패턴 또는 핵심 요소 추출하기</p> <ul style="list-style-type: none"> ○ 분해한 문제의 장면, 또는 기능에 필요한 자료, 정 	<p>♣핵심 요소 추출을 예를 들어 설명하고 핵심 요소에 대한 자유로운 의견이 나올 수 있도록</p>																																

<p>전개2 (35분)</p>	<p>활동5 (모둠)</p>	<p>보는 어떤 것들이 있는지 생각해보기</p> <ul style="list-style-type: none"> ○ 반복되고 있는 패턴이나 필요한 오브젝트, 변수, 연산, 입력, 출력은 어떤 것들이 있는지 생각해보기 ○ 강조할 것은 무엇이고 단순화 할 것은 무엇인지 생각해보기 ○ 이와 비슷한 문제 해결 방법을 탐색하기 <p>■ [활동4] 문제 해결 모델 설계</p> <ul style="list-style-type: none"> ○ 문제 분해, 핵심 요소 추출을 바탕으로 프로그래밍으로 구현할 문제 해결 방법을 정리하기 ○ 문제 해결을 위한 모델 설계하기 <p>■ 자기 평가</p> <table border="1" data-bbox="488 887 1126 1305"> <thead> <tr> <th></th> <th>아니다</th> <th>보통이다</th> <th>그렇다</th> </tr> </thead> <tbody> <tr> <td>나는 문제의 반복되는 패턴이나 핵심 요소를 추출 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 문제 해결을 위한 아이디어를 제시하거나, 유사한 문제 해결 방법을 탐색하여 적용 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 분해된 문제를 핵심요소를 바탕으로 구조화하여 표현하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 문제 해결 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델을 설계 하였다.</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		아니다	보통이다	그렇다	나는 문제의 반복되는 패턴이나 핵심 요소를 추출 하였다.				나는 문제 해결을 위한 아이디어를 제시하거나, 유사한 문제 해결 방법을 탐색하여 적용 하였다.				나는 분해된 문제를 핵심요소를 바탕으로 구조화하여 표현하였다.				나는 문제 해결 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델을 설계 하였다.				<p>토의 분위기를 조성한다.</p> <p>♣모둠을 돌아다니면서 학생들의 핵심요소 추출 의견을 들어보고 피드백 해준다.(핵심요소 추출 활동에 대한 관찰 평가)</p> <p>♣문제 해결 모델 설계.</p> <p>♣평가 요소를 토대로 자기 평가를 재구성하여 제시한다.</p>
	아니다	보통이다	그렇다																				
나는 문제의 반복되는 패턴이나 핵심 요소를 추출 하였다.																							
나는 문제 해결을 위한 아이디어를 제시하거나, 유사한 문제 해결 방법을 탐색하여 적용 하였다.																							
나는 분해된 문제를 핵심요소를 바탕으로 구조화하여 표현하였다.																							
나는 문제 해결 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델을 설계 하였다.																							
<p>정리 (5분)</p>	<p>정리 (전체) 차시예고</p>	<p>■ 학습활동 정리하기</p> <ul style="list-style-type: none"> ○ 활동을 하면서 느낀 점 이야기하기 ○ 문제 인식 및 분석, 해결책 탐색의 중요성에 대하여 이야기하기 <p>■ 차시예고</p> <ul style="list-style-type: none"> ○ 다음 프로젝트 활동 안내하기 <ul style="list-style-type: none"> - 스토리 보드, 알고리즘 설계 - 프로그램 제작 																					

< 3~4차시 지도안 >

학습 목표	<ul style="list-style-type: none"> ◦ 문제 해결을 위한 다양한 방법과 절차를 탐색한다. ◦ 문제 해결 모델에 대한 알고리즘을 명확하고 논리적인 알고리즘으로 표현한다. ◦ 문제 해결 알고리즘을 개발 환경에 맞게 자동화 한다.
------------------	---

평가 계획	성취 기준	<ul style="list-style-type: none"> ■ 문제 해결을 위한 다양한 방법과 절차를 탐색하고 시각적으로 명확한 알고리즘으로 표현한다. ■ 문제 해결 모델에 대한 알고리즘을 제어 구조를 이용하여 논리적으로 설계한다. 		
	평가 기준	평가 요소	평가 기준	
		<ul style="list-style-type: none"> • 다양한 문제 해결 방법과 절차를 탐색하여 비교하고 분석할 수 있는가? • 문제 해결 절차를 글, 그림 등을 이용하여 알고리즘으로 명확하게 표현할 수 있는가? 	상	다양한 문제 해결 방법과 절차를 탐색하고 비교하여 효율적인 절차를 선택하고 글, 그림 등을 이용하여 시각적인 형태로 쉽고 명확하게 표현할 수 있다.
			중	문제 해결 방법과 절차를 시각적인 형태로 쉽고 명확하게 표현할 수 있다.
			하	문제 해결 방법과 절차를 시각적으로 표현하였지만 명확하게 표현하지 못한다.
		<ul style="list-style-type: none"> • 문제 해결 알고리즘을 제어 구조를 이용하여 논리적으로 설계할 수 있는가? 	상	제어 구조를 이용하여 논리적인 알고리즘을 설계할 수 있다.
			중	제어 구조를 이용하여 논리적인 알고리즘을 설계함에 있어 불필요한 부분이 포함되어 있거나, 필요한 부분이 누락되었다.
	하		제어 구조를 이용하여 논리적인 알고리즘을 설계하지 못한다.	
	평가 시기	수업 중, 수업 후		
	평가 방법	자기평가, 관찰평가, 포트폴리오		
산출 물	학습지, 자료 저장 파일			

단계	학습 요소	교수·학습 활동	자료 및 유의점				
도입 (5분)	<p>동기유발 (전체)</p> <p>학습 목표 안내 (전체)</p>	<p>■ 전시 학습 내용 확인</p> <ul style="list-style-type: none"> ○ 지난 시간에 만들기로 한 프로그램 확인 ○ 모듈 별로 수집한 자료 확인 <p>■ 학습 목표 확인</p> <div style="border: 1px dashed black; padding: 5px;"> <ul style="list-style-type: none"> ○ 문제 해결을 위한 다양한 방법과 절차를 탐색한다. ○ 문제 해결 모델에 대한 알고리즘을 명확하고 논리적인 알고리즘으로 표현한다. ○ 문제 해결 알고리즘을 개발 환경에 맞게 자동화 한다. </div>					
전개1 (45분)	<p>학습활동 안내 (전체)</p> <p>활동1 (모둠)</p> <p>활동2 (모둠)</p>	<p style="text-align: center;">【주요 학습 활동】</p> <div style="border: 1px solid black; padding: 5px;"> <p>[활동1] 문제 해결 과정 스토리 보드 작성하기</p> <p>[활동2] 스토리 보드 장면 알고리즘 작성하기</p> </div> <p>■ [활동1] 문제 해결 과정 스토리 보드 작성하기</p> <ul style="list-style-type: none"> ○ 문제 해결 방법을 소프트웨어로 구현 시 필요한 장면이나 기능을 스토리 보드로 표현하기 <ul style="list-style-type: none"> - 지난 시간에 설계한 모델을 토대로 표기 - 시간의 흐름을 알 수 있도록 스토리로 표기 <p>■ [활동2] 스토리 보드 장면 알고리즘 작성하기</p> <ul style="list-style-type: none"> ○ 작성한 스토리 보드 장면에 해당하는 알고리즘 작성 ○ 순차, 선택, 반복 구조 등 제어구조 활용 ○ 변수, 연산자의 표현 ○ 다양한 알고리즘에 대한 토의 ○ 시각적으로 명확하게 표현 <p>■ 자기 평가</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 50%; height: 20px;"></td> <td style="width: 15%;">아니다</td> <td style="width: 15%;">보통이다</td> <td style="width: 15%;">그렇다</td> </tr> </table>		아니다	보통이다	그렇다	<p>♣서로 토의하며 각 장면을 스케치 하도록 분위기 조성</p> <p>♣알고리즘 논리적, 시각적 표기 예시를 보여줌</p> <p>♣다양한 문제 해결 절차를 이야기 하도록 유도, 모듈별로 지속적으로 대화하며 피드백(알고리즘 작성 활동에 대한 관찰 평가)</p> <p>♣평가 요소를 토대로 자기 평가를 재구성하여 제시한다.</p>
	아니다	보통이다	그렇다				

		<table border="1"> <tr> <td>나는 문제 해결을 위한 다양한 알고리즘을 토의 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 알고리즘을 명확하게 표현 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 순차, 선택, 반복 구조, 변수, 연산자 등을 이용하여 논리적인 알고리즘을 설계 하였다.</td> <td></td> <td></td> <td></td> </tr> </table>	나는 문제 해결을 위한 다양한 알고리즘을 토의 하였다.				나는 알고리즘을 명확하게 표현 하였다.				나는 순차, 선택, 반복 구조, 변수, 연산자 등을 이용하여 논리적인 알고리즘을 설계 하였다.				
나는 문제 해결을 위한 다양한 알고리즘을 토의 하였다.															
나는 알고리즘을 명확하게 표현 하였다.															
나는 순차, 선택, 반복 구조, 변수, 연산자 등을 이용하여 논리적인 알고리즘을 설계 하였다.															
전개2 (45분)	학습활동 안내 (전체)	<p style="text-align: center;">【주요 학습 활동】</p> <div style="border: 1px solid black; padding: 5px;"> <p>[활동3] 알고리즘 자동화하기 1</p> </div>	<p>♣추후 작품 공유를 수월하게 하기 위하여 '엔트리' 프로그램 로그인을 하고, 학급 그룹으로 진행함</p>												
	활동3 (모둠)	<p>■ [활동3] 알고리즘 자동화하기 1</p> <ul style="list-style-type: none"> ○ 설계한 알고리즘을 블록코딩으로 자동화하기 ○ 순차, 선택, 반복 구조, 변수, 연산자 활용하기 ○ 오류가 발생할 경우 원인을 찾아 수정하기 	<p>♣모둠별로 자동화를 진행하지만 한 사람만 코딩하는 것을 막기 위해 각자 자동화를 진행함.</p> <p>♣모듬을 돌아다니면서 학생들의 자동화 과정을 관찰하고 피드백 해준다.(자동화 활동에 대한 관찰 평가)</p>												
정리 (5분)	정리 (전체)	<p>■ 학습활동 정리하기</p> <ul style="list-style-type: none"> ○ 알고리즘을 구상하면서 느낀 점 이야기하기 ○ 알고리즘을 작성하고 자동화 작업을 하면 좋은 점 이야기하기 													
	차시예고	<p>■ 차시예고</p> <ul style="list-style-type: none"> ○ 프로그램 구현 작업 마무리 ○ 완성된 작품 공유 ○ 문제 해결 과정 되돌아보기 													

< 5~6차시 지도안 >

학습 목표	<ul style="list-style-type: none"> ◦ 프로그래밍 언어의 개발 환경 및 특성에 맞게 자료를 입력받아 처리하고 결과를 출력하는 프로그램을 작성한다. ◦ 오류가 발생할 경우 원인을 찾아 실행 가능한 형태로 수정한다. ◦ 결과물을 공유 및 평가하고 문제 해결 과정을 되돌아보며 성찰 및 피드백한다.
------------------	---

성취 기준	<ul style="list-style-type: none"> ■ 설계한 알고리즘을 구현할 프로그래밍 언어를 선택하고, 프로그래밍 언어의 개발 환경 및 특성에 맞게 자동화하여 구현한다. ■ 제어 구조, 호출, 변수와 연산자 등을 활용하여 자료를 입력받아 처리하고 출력하는 프로그램을 작성한다. ■ 오류 발생 시 프로그램이나 알고리즘을 분석하여 오류의 원인을 발견하고 실행 가능한 형태로 수정한다. ■ 문제를 해결함에 있어 역할 분담, 자신의 임무 수행, 팀원 간에 의사소통 등 협업하여 프로그램을 개발 하였는지 확인한다. ■ 결과물을 서로 공유 및 평가하고 문제 해결 과정에 대하여 성찰 및 피드백 하는 태도를 취한다. 		
	평가 계획	평가 요소	평가 기준
평가 기준	<ul style="list-style-type: none"> • 설계한 알고리즘을 프로그래밍 언어의 개발 환경 및 특성에 맞게 자동화하여 구현할 수 있는가? 	상	설계된 알고리즘을 프로그래밍 언어의 개발 환경 및 특성에 맞게 구조적, 논리적으로 동일하게 자동화하여 구현할 수 있다.
평가 기준		중	프로그래밍 언어의 개발 환경 및 특성에 맞게 구현 하였지만, 설계된 알고리즘 일부분만 구현할 수 있다.
평가 기준	<ul style="list-style-type: none"> • 제어구조, 호출, 변수, 연산자 등을 사용함에 있어 논리적인 오류 없이 프로그램을 구현할 수 있는가? • 다양한 형태의 자료 	하	설계된 알고리즘을 자동화하여 구현하지 못한다.
평가 기준		상	프로그램의 목적에 맞게 다양한 형태의 자료를 입력 받아 처리하기 위해 순차, 선택, 반복 구조와 호출, 변수, 연산자를 논리적으로 사용하여 원하는 결과를 출력하는 프로그램을 구현할 수 있다.
평가 기준	중	순차, 선택, 반복 구조와 호출, 변수,	

	를 입력 받아 처리한 후 원하는 출력 값을 얻는 프로그램을 구현할 수 있는가?		연산자를 사용하여 프로그램의 목적에 맞게 구현 하였지만 불필요한 블록이나 객체가 포함되어 있다.
		하	프로그램의 목적에 맞게 자료를 입력 받아 처리하고 결과를 출력하는 프로그램을 구현하지 못하였다.
	<ul style="list-style-type: none"> 오류의 원인을 체계적으로 분석하여 찾아낼 수 있는가? 오류를 프로그램 실행 가능한 형태로 수정할 수 있는가? 	상	오류 발생 시 지속적으로 시험을 시도하고, 코드를 확인하는 등 체계적으로 오류의 원인을 찾아내고 실행 가능한 형태로 수정할 수 있다.
		중	무작위적 오류 발견을 통해 오류의 원인을 찾아내고, 오류를 실행 가능한 형태로 수정할 수 있다.
		하	오류가 발생 하였지만 오류의 원인을 찾지 못해 오류를 수정하지 못한다.
	<ul style="list-style-type: none"> 문제 해결 과정에서 각자의 역할을 나누고, 할당된 과제를 수행하여 과제의 목표를 달성할 수 있는가? 문제 해결을 위한 아이디어, 의견을 나누고 공유하며 지속적인 의사소통을 할 수 있는가? 	상	문제 해결 과정에서 역할 분담을 하였으며, 서로의 과제를 확인하여 공통의 목표를 달하고, 지속적으로 아이디어나 의견을 공유하여 의사소통 하였다.
		중	문제 해결 과정에서 팀원 간의 의견을 공유가 되지 않아, 불균형 적인 역할 분담으로 한쪽에 과중된 과제가 주어진 상태에서 프로그램을 개발하였다.
		하	문제 해결 과정에서 역할 분담이 되지 않고, 팀원 간의 의견을 공유하지 않았다.
	• 동료들과 결과물을 공유하여 잘된 점,	상	동료 평가를 통해 결과물의 장단점과 개선점에 대해 적극적으로 의견을 제

		아쉬운 점 등을 이야기 할 수 있는가?	시하고 있으며, 동료의 피드백과 자기 평가를 바탕으로 자신의 문제 해결 과정을 되돌아보며 성찰하고 피드백 하는 태도를 취한다.
		• 문제 해결 과정을 성찰하고 이와 유사한 문제를 찾아 논의하여 피드백 할 수 있는가?	중 동료 평가를 통해 결과물의 장단점과 개선점에 대해 간단한 의견을 제시하고 있지만, 동료의 피드백과 자기평가를 바탕으로 성찰하는 태도가 미흡하다.
			하 동료 평가에서 결과물에 대한 의견을 제시하지 않고, 자신의 문제 해결 과정을 성찰하지 않는다.
	평가 시기	수업 중, 수업 후	
평가 방법	관찰평가, 포트폴리오		
산출물	학습지, 자료 저장 파일		

단계	학습 요소	교수.학습 활동	자료 및 유의점
도입 (5분)	동기유발 (전체)	■ 전시 학습 내용 확인 <ul style="list-style-type: none"> ○ 지난 시간에 만들기로 한 프로그램 확인 ○ 모둠 별로 수집한 자료 확인 	
	학습 목표 안내 (전체)	■ 학습 목표 확인 <ul style="list-style-type: none"> ○ 프로그래밍 언어의 개발 환경 및 특성에 맞게 자료를 입력받아 처리하고 결과를 출력하는 프로그램을 작성한다. ○ 오류가 발생할 경우 원인을 찾아 실행 가능한 형태로 수정한다. ○ 결과물을 공유 및 평가하고 문제 해결 과정을 되돌아보며 성찰 및 피드백 한다. 	
	학습활동 안내 (전체)	<p style="text-align: center;">【주요 학습 활동】</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;">[활동1] 알고리즘 자동화하기 2</div>	
	활동1	■ [활동1] 알고리즘 자동화하기 2	♣지난 시간에 제작하

<p>전개1 (40분)</p>	<p>(모둠)</p>	<ul style="list-style-type: none"> ○ 설계한 알고리즘을 블록코딩으로 자동화하기 ○ 순차, 선택, 반복 구조, 변수, 연산자 활용하기 ○ 오류가 발생할 경우 원인을 찾아 수정하기 <p>■ 자기 평가</p> <table border="1" data-bbox="544 712 1153 1196"> <thead> <tr> <th></th> <th>아니다</th> <th>보통이다</th> <th>그렇다</th> </tr> </thead> <tbody> <tr> <td>나는 설계한 알고리즘을 개발환경 및 특성에 맞게 자동화하여 구현 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 제어구조, 호출, 변수, 연산자 등을 사용함에 있어 논리적인 오류 없이 프로그래밍 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 자료를 입력 받아 원하는 결과 값을 얻는 프로그램을 구현 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 오류의 원인을 체계적으로 분석하여 찾아내었다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 오류를 프로그램 실행 가능한 형태로 수정 하였다.</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		아니다	보통이다	그렇다	나는 설계한 알고리즘을 개발환경 및 특성에 맞게 자동화하여 구현 하였다.				나는 제어구조, 호출, 변수, 연산자 등을 사용함에 있어 논리적인 오류 없이 프로그래밍 하였다.				나는 자료를 입력 받아 원하는 결과 값을 얻는 프로그램을 구현 하였다.				나는 오류의 원인을 체계적으로 분석하여 찾아내었다.				나는 오류를 프로그램 실행 가능한 형태로 수정 하였다.				<p>던 프로그램에 대한 마무리 작업 유도</p> <p>♣모듬을 돌아다니면서 학생들의 자동화 과정을 관찰하고 피드백 해준다.(자동화 활동에 대한 관찰 평가)</p> <p>♣평가 요소를 토대로 자기 평가를 재구성하여 제시한다.</p>
	아니다	보통이다	그렇다																								
나는 설계한 알고리즘을 개발환경 및 특성에 맞게 자동화하여 구현 하였다.																											
나는 제어구조, 호출, 변수, 연산자 등을 사용함에 있어 논리적인 오류 없이 프로그래밍 하였다.																											
나는 자료를 입력 받아 원하는 결과 값을 얻는 프로그램을 구현 하였다.																											
나는 오류의 원인을 체계적으로 분석하여 찾아내었다.																											
나는 오류를 프로그램 실행 가능한 형태로 수정 하였다.																											
<p>전개2 (45분)</p>	<p>학습활동 안내 (전체)</p> <p>활동2 (모듬)</p> <p>활동3</p>	<p>【주요 학습 활동】</p> <p>[활동2] 완성된 작품 공유하기 [활동2] 다른 모듬이 만든 작품을 보고 평가하기 [활동4] 자기 모듬의 문제 해결 과정을 되돌아 보고 피드백하기</p> <p>■ [활동2] 완성된 작품 공유하기</p> <ul style="list-style-type: none"> ○ 엔트리 학급공유를 통해 모듬별 작품 공유하기 ○ 학급 공유 시 문제 해결 방법 아이디어에 대한 자료 정리 <ul style="list-style-type: none"> - 작품 소개, 작품 아이디어, 문제 분해, 핵심 요소, 문제 해결 모델, 알고리즘 등 <p>■ [활동3] 다른 모듬이 만든 작품을 보고 평가하기</p>	<p>♣자료의 표현 방식을 예를 들어 설명하고 자료의 종류와 특성에 맞게 표현하도록 유도, 피드백 (자료 공유, 분석, 표현에 대한 관찰 평가)</p> <p>♣각 모듬을 모두 평가할 수 있도록 지도(동</p>																								

	<p>(전체)</p> <p>활동4 (모둠)</p>	<ul style="list-style-type: none"> ○ 엔트리 학급 공유에 올라온 모둠별 작품과 작품 설명을 보고 평가하기 <ul style="list-style-type: none"> - 잘된 점, 개선할 점 등 - 자신의 모둠과 비교해보기 <p>■ [활동4] 자기 모둠의 문제 해결 과정을 되돌아보고 피드백하기</p> <ul style="list-style-type: none"> ○ 협업 과정에 어려움은 없었는지 이야기하기 ○ 다른 모둠이 자기 모둠의 작품에 대한 평가한 것을 보고 자기 모둠 작품에서 개선해야 할 점은 무엇인지 이야기하기 ○ 문제 해결 과정에서 좋았던 점, 아쉬웠던 점, 어려웠던 점 등을 이야기하기 ○ 문제 해결 과정과 유사한 문제에 대하여 이야기해보기 <p>■ 자기 평가</p> <table border="1" data-bbox="545 1059 1145 1514"> <thead> <tr> <th></th> <th>아니다</th> <th>보통이다</th> <th>그렇다</th> </tr> </thead> <tbody> <tr> <td>나는 문제 해결 과정에서 팀원과 각자의 역할을 나누고 할당된 과제를 수행하여 공동의 목표를 달성 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 문제 해결을 위한 아이디어, 의견을 팀원과 지속적으로 공유하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 친구들과 결과물을 공유하고 잘된 점, 아쉬운 점을 이야기 하였다.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>나는 문제 해결 과정을 성찰하고 이와 유사한 문제 해결 방법에 대해 이야기 하였다.</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		아니다	보통이다	그렇다	나는 문제 해결 과정에서 팀원과 각자의 역할을 나누고 할당된 과제를 수행하여 공동의 목표를 달성 하였다.				나는 문제 해결을 위한 아이디어, 의견을 팀원과 지속적으로 공유하였다.				나는 친구들과 결과물을 공유하고 잘된 점, 아쉬운 점을 이야기 하였다.				나는 문제 해결 과정을 성찰하고 이와 유사한 문제 해결 방법에 대해 이야기 하였다.				<p>료 평가 - 다른 모둠 결과물에 대한 평가)</p> <p>♣문제 해결 과정을 성찰할 수 있도록 자기평가와 동료평가를 진행함</p>
	아니다	보통이다	그렇다																				
나는 문제 해결 과정에서 팀원과 각자의 역할을 나누고 할당된 과제를 수행하여 공동의 목표를 달성 하였다.																							
나는 문제 해결을 위한 아이디어, 의견을 팀원과 지속적으로 공유하였다.																							
나는 친구들과 결과물을 공유하고 잘된 점, 아쉬운 점을 이야기 하였다.																							
나는 문제 해결 과정을 성찰하고 이와 유사한 문제 해결 방법에 대해 이야기 하였다.																							
<p>정리 (5분)</p>	<p>정리 (전체)</p>	<p>■ 학습활동 정리하기</p> <ul style="list-style-type: none"> ○ 컴퓨팅 사고력 기반의 문제 해결 과정에 대해 느낀 점 이야기하기 																					

2. 중학교 컴퓨팅 사고력 과정평가 모델 교수학습 활동지



학년	
조	
이름	

문제 상황

※ 거리의 무법자 슬몐비 사고 급증

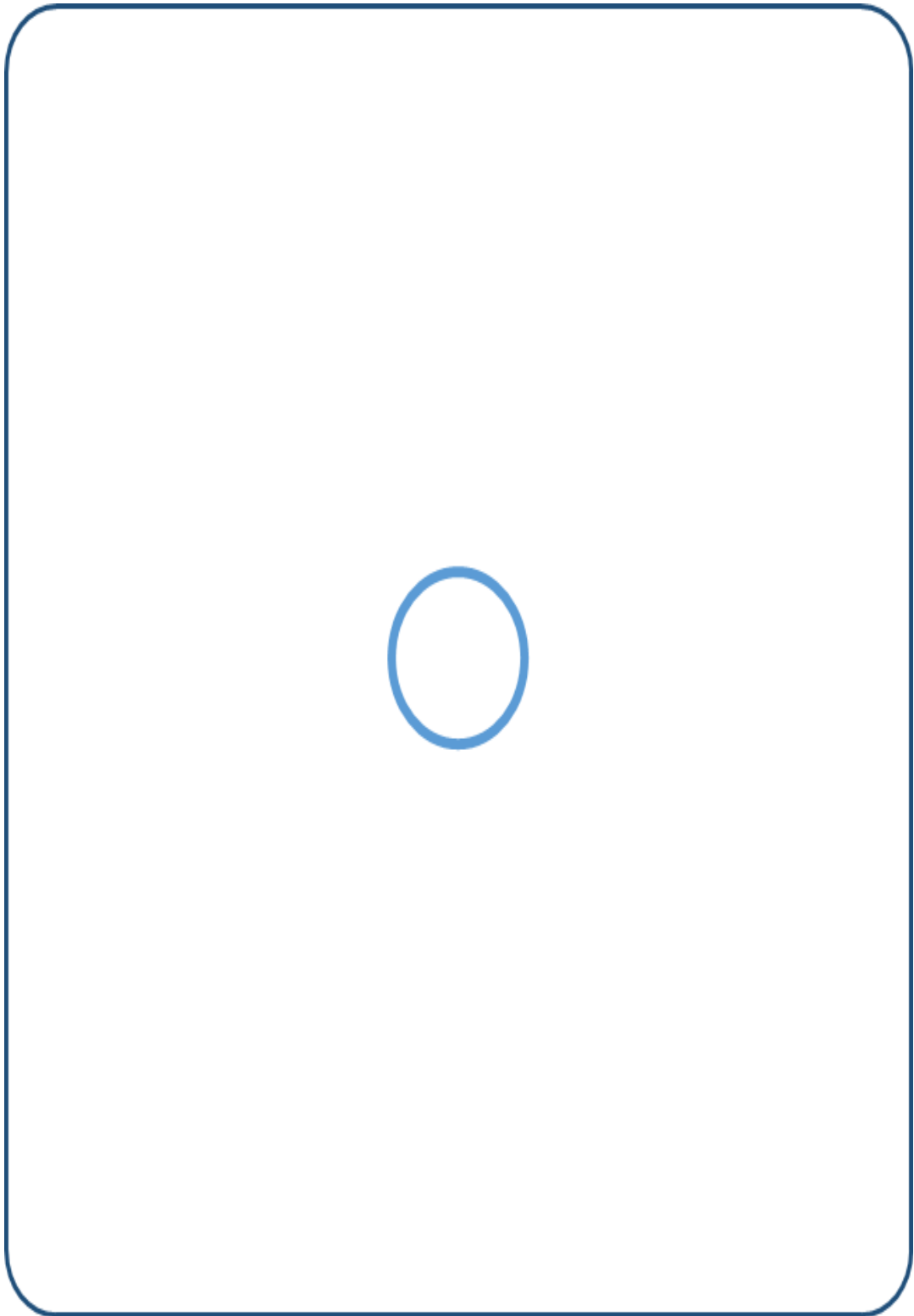
슬몐비라고 들어본 적 있나요? 최근 스마트폰에 빠진 사람이 늘어나며 슬몐비로 인한 사고가 급증하고 있다고 합니다. 다음 뉴스를 보고 슬몐비 문제에 대하여 생각해 봅시다.



<자료출처 : <https://www.yna.co.kr/view/MYH20180912003200038> >

※ 뉴스를 보고 슬몐비 사고 문제의 원인이 무엇인지, 무엇을 어떻게 해결해야 할지에 대하여 이야기해 봅시다.

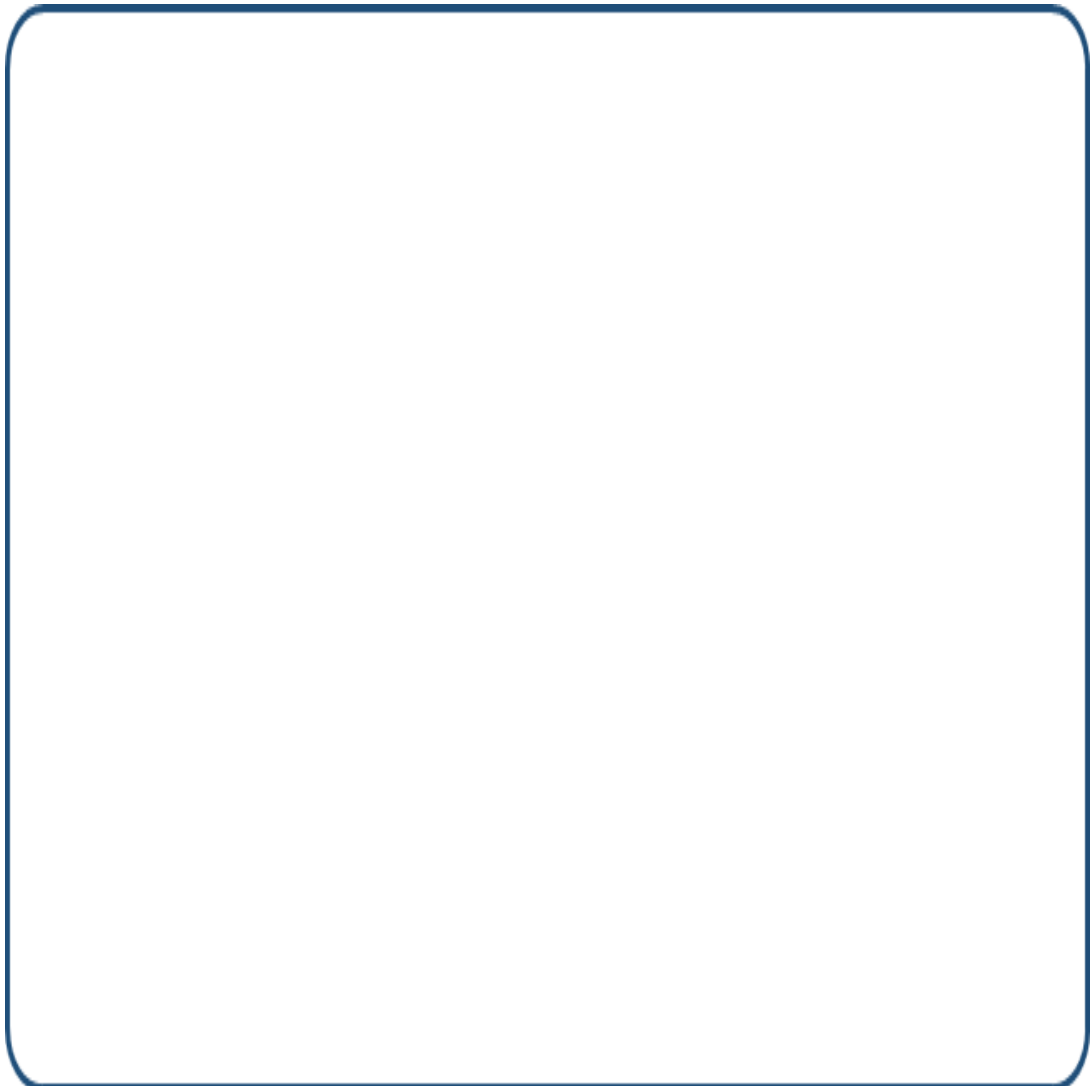
※ 슬몐비 사고를 소프트웨어로 해결하기 위한 아이디어를 생각해 봅시다.



※ 문제 해결을 위한 자료를 인터넷, 설문 등 다양한 방법으로 수집 및 저장하여 관리해 봅시다.



※ 문제 해결을 위해 수집한 자료를 친구와 공유 하고 표, 다이어그램, 그림 등으로 구조화하여 표현해 봅시다.

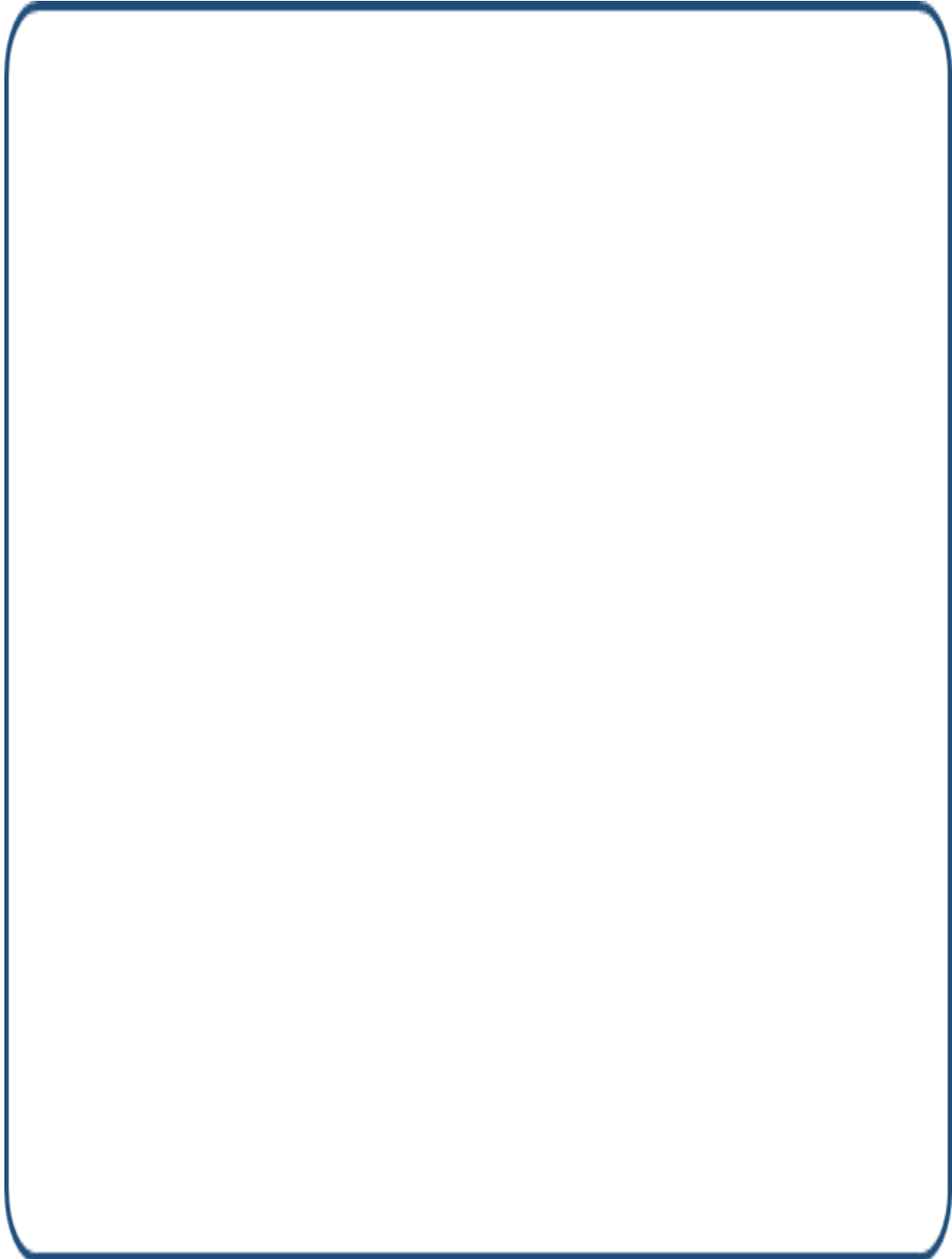


※ 문제 해결을 위한 소프트웨어를 구현 할 때 필요한 기능이나 장면을 분해하여 이야기해 봅시다.

※평가하기!

나는 문제의 원인과 함께 현재 상태와 목표 상태를 설명 하였다.	☆ ☆ ☆
나는 소프트웨어로 문제를 해결하기 위한 방법을 제시 하였다.	☆ ☆ ☆
나는 문제 해결을 위한 자료를 수집 하였다.	☆ ☆ ☆
나는 수집한 자료를 컴퓨터에 저장하고, 사용하기 쉽게 응용 소프트웨어를 이용하여 관리 하였다.	☆ ☆ ☆
나는 수집한 자료를 친구와 공유 하였다.	☆ ☆ ☆
나는 수집한 자료를 분석하여 표, 그림 등으로 보기 쉽게 표현 하였다.	☆ ☆ ☆
나는 기존의 문제를 해결 가능한 작은 문제로 분해하였다.	☆ ☆ ☆

※ 분해된 각각의 문제에서 반복되는 패턴이나 기능과 장면에서 꼭 필요한 자료, 정보, 오브젝트, 변수 등 핵심 요소는 어떤 것이 있을지 이야기 문제 해결을 위한 아이디어를 제시하여 봅시다.



※ 분해된 문제와 패턴, 핵심요소를 바탕으로 문제 해결을 위한 아이디어를 컴퓨터로 구현하기 위한 모델을 설계하여 봅시다.



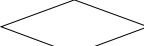


※평가하기!

나는 문제의 반복되는 패턴이나 핵심 요소를 추출 하였다.	☆ ☆ ☆
나는 문제 해결을 위한 아이디어를 제시하거나, 유사한 문제 해결 방법을 탐색 하여 적용 하였다.	☆ ☆ ☆
나는 분해된 문제를 핵심요소를 바탕으로 구조화하여 표현하였다.	☆ ☆ ☆
나는 문제 해결 아이디어를 컴퓨터로 구현하기 위한 문제 해결 모델을 설계 하였다.	☆ ☆ ☆

※ 문제 해결 모델을 바탕으로 다양한 문제 해결 절차를 이야기 하고 문제 해결 과정을 스토리 보드로 작성해 봅시다.

※ 작성한 스토리 보드 별로 알고리즘을 작성해 봅시다.

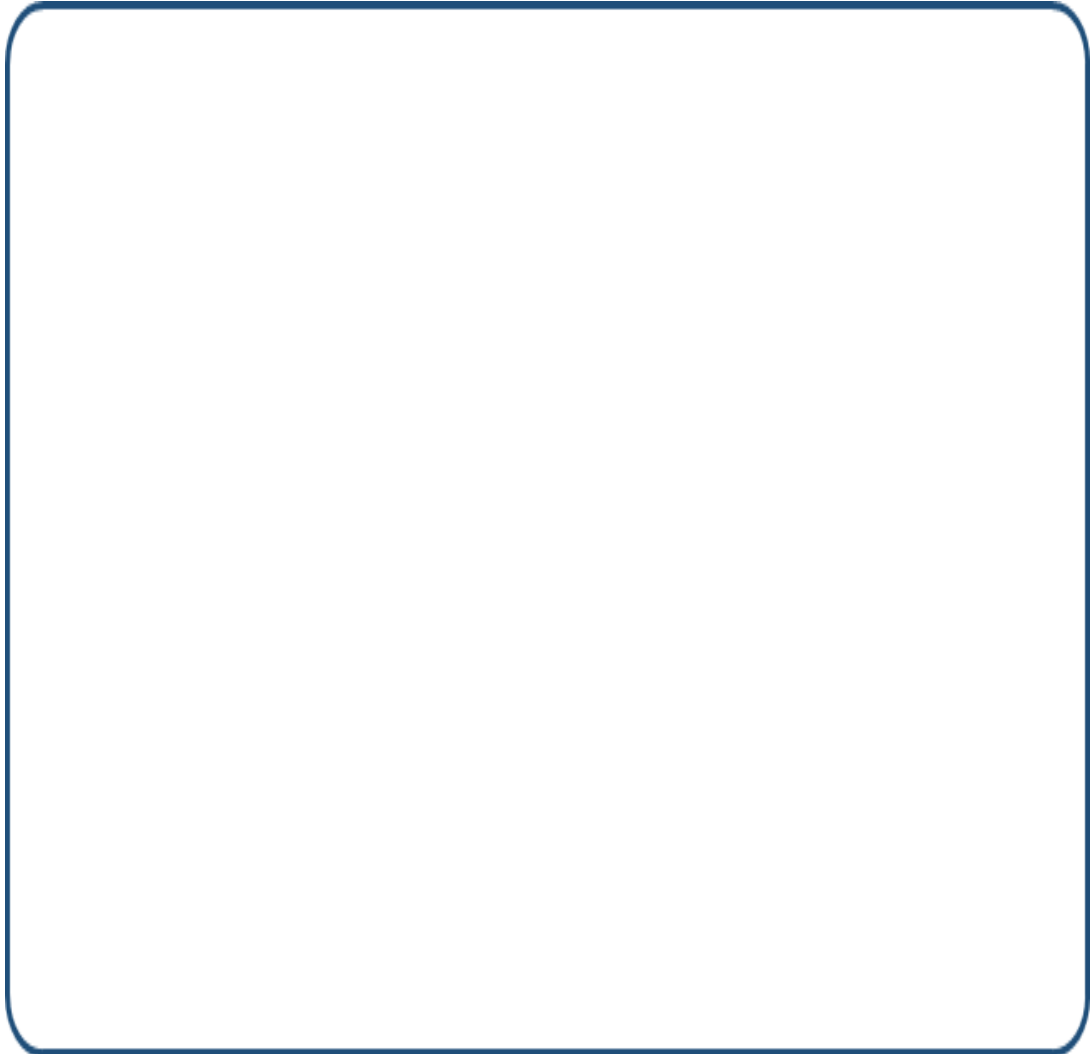
< 알고리즘 표기의 예 >

종류	이름	기호	의미	예
자료 기호	입출력		자료의 입력 또는 출력에 사용	<p style="text-align: center;"><코끼리 냉장고에 넣기></p> <pre> graph TD Start([시작]) --> Open[냉장고 문 열기] Open --> Put[코끼리 넣기] Put --> Close[냉장고 문 닫기] Close --> End([종료]) </pre>
처리 기호	처리		자료를 처리하거나 여러 가지 연산을 처리할 때 사용	
	판단 (조건)		조건에 따라 처리할 때 사용	
선 기호	흐름선		실행 순서, 기호를 연결할 때나 방향을 표시	
특수 기호	단말		순서도의 시작과 끝	

※평가하기!

나는 문제 해결을 위한 다양한 알고리즘을 토의 하였다.	☆ ☆ ☆
나는 알고리즘을 명확하게 표현 하였다.	☆ ☆ ☆
나는 순차, 선택, 반복 구조, 변수, 연산자 등을 이용하여 논리적인 알고리즘을 설계 하였다.	☆ ☆ ☆

※ 설계한 알고리즘을 자동화하여 구현해 보고 완성된 작품을 공유해 봅시다.



※평가하기!

나는 설계한 알고리즘을 개발환경 및 특성에 맞게 자동화하여 구현 하였다.	☆	☆	☆
나는 제어구조, 호출, 변수, 연산자 등을 사용함에 있어 논리적인 오류 없이 프로그래밍 하였다.	☆	☆	☆
나는 자료를 입력 받아 원하는 결과 값을 얻는 프로그램을 구현 하였다.	☆	☆	☆
나는 오류의 원인을 체계적으로 분석하여 찾아내었다.	☆	☆	☆
나는 오류를 프로그램 실행 가능한 형태로 수정 하였다.	☆	☆	☆

※ 다른 모둠의 작품을 보고 나의 작품이 개선되어야 할 점이 무엇이 있는지 생각해 봅시다.



※ 문제 해결 과정에서 친구와 협업하는 과정에서 느낀 점을 이야기 해 봅시다.



※ 문제 해결 과정에서 어려웠던 점, 개선해야 할 점을 이야기 해 봅시다.

※ 나의 문제 해결 과정을 바탕으로 '슬몌비 사고' 문제가 아닌 다른 문제를 해결한다면 어떤 문제를 해결할 수 있을지 이야기 해 봅시다.

※평가하기!

나는 문제 해결 과정에서 팀원과 각자의 역할을 나누고 할당된 과제를 수행하여 공동의 목표를 달성 하였다.	☆	☆	☆
나는 문제 해결을 위한 아이디어, 의견을 팀원과 지속적으로 공유하였다.	☆	☆	☆
나는 친구들과 결과물을 공유하고 잘된 점, 아쉬운 점을 이야기 하였다.	☆	☆	☆
나는 문제 해결 과정을 성찰하고 이와 유사한 문제 해결 방법에 대해 이야기 하였다.	☆	☆	☆