

# 인공지능시대에서의 프로그래밍교육 활성화 방안 연구

전 우 천\*

## ◆ 목 차 ◆

- |                          |                    |
|--------------------------|--------------------|
| 1. 서 론                   | 4. 프로그래밍 교육 활성화 방안 |
| 2. 프로그래밍 교육의 의의          | 5. 결 론             |
| 3. 소프트웨어교육에서의 프로그래밍 교육현황 |                    |

## 1. 서 론

현재 우리는 4차 산업혁명 시대에 살고 있으며 당분간 4차 산업혁명은 우리 사회 및 개인의 삶에 직간접적으로 다양한 영향을 미칠 것으로 예상하고 있다. 4차 산업혁명은 초연결(Hyper-connectivity)와 초지능(Super-intelligence)로 대표된다. 즉 초연결은 다양한 사물이 유선 또는 무선 통신을 통하여 다양한 방법으로 정보를 교류할 수 있는 환경을 말하며, 이에 비해서 초지능은 사물이 일정한 지능을 가지고 있어서 인간의 개입이 없이 일정부분 스스로 판단할 수 있음을 의미한다. 4차 산업혁명을 대표할 수 있는 무인자동차의 경우 초연결과 초지능의 대표적인 제품이라고 할 수 있다. 즉 다양한 센서를 통하여 온도, 습도, 물체 등을 탐지할 수 있으며 위성과의 통신을 통하여 다양한 교통정보를 주고 받을 수 있다. 또한 초지능을 통하여 다양한 판단을 스스로 결정하여 인간 운전자의 개입이 없이 스스로 주행할 수 있는 단계까지 이르렀다.

1차 산업혁명의 경우 1780년대 영국에서 시작되었으며 그 출발점은 Thomas Newcomen이 발명하고 James Watt가 개량한 증기기관이라고 할 수 있다. 증기기관을 바로 증기기관차의 탄생을 알렸고 물류혁명을 가져왔다. 즉 그동안 말 중심의 교통에서 획기적으로 더 많은 거리를 달릴 수 있으며 더 많은 화물을 실고 달릴 수 있어 물류의 이동 확대라는 장점과 더

불어 군사적으로 제국주의의 확대를 가져왔다. 1차 산업혁명의 키워드는 '기계화'라고 표현할 수 있다. 이에 비해서 2차 산업혁명은 1870년대 영국과 미국을 중심으로 일어났으며 내연기관과 전기가 합쳐져서 대량생산이라는 또 다른 혁명을 가져오게 된다. 내연기관과 전기를 바탕으로 하는 대량생산은 많은 노동력이 없이도 많은 제품을 생산할 수 있어서 소위 '부의 부빈익빈'이라는 사회문제를 유발하기도 했으며, 부유 계층의 노동력 착취가 다양한 국가에서 사회적인 이슈로 등장하게 되었다. 3차 산업혁명의 시작시기에 대해서는 다양한 논란이 있으나 일반적으로 1980년대초를 시작으로 보는 시각이 우세하다. 즉 1980년대초 개인용 컴퓨터 즉 PC의 등장과 더불어 정보화 사회가 시작되었다. 정보화 사회는 2차 산업혁명과는 달리 새로운 패러다임을 가지고 왔다. 즉 정보화의 다양하고 창의적인 아이디어를 가지고 얼마든지 부를 창출할 수 있다는 것이다. 2차 산업혁명의 경우 새로운 제품을 상품화하기 위해서는 다양한 자본이 필요했지만 3차 산업혁명의 정보화과정은 다양한 장비와 노동력이 필요로 하지 않기 때문에 빌 게이츠와 같이 새로운 소프트웨어제품을 통하여 단기간에 막대한 부를 창출하는 것이 가능해 졌다. 3차 산업혁명은 누구에게나 평등한 기회를 가져다 주었고 창의적인 아이디어로 얼마든지 성공할 수 있는 혁신적인 패러다임의 변화를 야기시켰다.

4차 산업혁명의 핵심 원동력은 인공지능이라고 할 수 있다. 인공지능은 다양한 학문의 융합적 결정체이

\* 서울교육대학교 컴퓨터교육학과

며 또한 다양한 분야에 적용이 가능한 핵심 키워드로 등장했다. 이제 인공지능은 우리 국가, 사회 및 개인에 있어서 핵심이 되고 있다. 즉 국가적으로는 인공지능관련 산업의 부흥을 통하여 국가경쟁력을 향상시키고, 사회적으로는 다양한 인공지능관련기술을 통하여 사회의 발전에 기여할 수 있다. 예를 들어, 인공지능을 통한 질병관리, 자연재해예보 등 다양한 분야에 사회의 안전 및 통합에 기여할 수 있다. 개인적인 차원에서 인공지능에 대한 소양은 필수화가 되고 있으며 개인의 삶 즉 직업이나 레저 등에 있어서 다양한 부가효과를 발휘할 수 있다. 또한 인공지능시대에 있어서 올바른 정보통신윤리에 대한 이해와 실천의지도 매우 중요한 덕목이 되어가고 있다.

2015년 2월 교육부에서는 소프트웨어교육 운영지침을 발표하여 2000년 이후 실시되고 다양한 개정을 통해 운영되던 ICT교육 즉 정보통신기술교육을 소프트웨어교육으로 대체하여 운영할 것을 예고하고 있다 [교육부2015]. 소프트웨어운영지침에 기초하여 2018년부터는 중학교, 2019년부터는 초등학교에 소프트웨어교육을 전면적으로 실시하게 되었다. 소프트웨어교육의 핵심은 크게 3부분으로 나눌 수 있다.

- 계산적 사고력을 통한 알고리즘 교육
- 프로그래밍 교육
- 정보통신윤리 교육

이중에서 프로그래밍은 소프트웨어교육의 핵심이라고 말할 수 있다. 그동안 정보통신기술교육에서도 프로그래밍 교육을 강조했지만 정보통신기술교육에서는 학습자를 사용자(User)라고 했다면 소프트웨어교육에서는 학습자를 생산자(Maker)로 격상하여 스스로 소프트웨어를 만들 수 있는 능력을 키우는 것이 핵심이기 때문이다.

본 원고에서는 인공지능시대에서의 프로그래밍 교육의 발전방안을 제시하고자 한다. 이를 위해서 먼저 2장에서는 프로그래밍 교육의 다양한 의미를 소개하고, 3장에서는 현재 소프트웨어교육에서의 프로그래밍 교육과정을 소개하며, 4장에서는 프로그래밍 교육의 발전방안을 소개한다. 마지막 5장에서는 결론을 제시한다.

## 2. 프로그래밍 교육의 의의

[김갑수2010]에서는 프로그래밍 교육의 의의를 다음과 같이 정리하여 기술하고 있다. 즉 6가지의 범주에서 인지능력이 향상될 수 있음을 다음과 같이 기술하고 있다.

첫 번째 범주는 프로그래밍 교육을 통하여 수학의 기하학적 개념과 원리를 잘 파악할 수 있는 능력을 향상시킬 수 있다. 즉 프로그래밍 교육을 통하여 수학의 도형과 관련된 라이브러리(Library)를 이용할 수 있으며 이를 통하여 수학적 기하학에 관한 개념과 원리를 배울 수 있는 능력이 향상될 수 있다는 것이다. 또한 변수 또는 수식표현 등을 원하는 형태로 표현할 수 있기 때문에 수학적 기하학적인 지식의 전이가 향상된다는 것이다.

두 번째 범주는 프로그래밍 교육을 통하여 문제 해결능력 및 문제 관리 능력이 향상할 수 있다. 이것은 학생들이 프로그래밍을 통하여 문제를 해결할 수 있는 능력을 키워줄 수 있고 또한 에러를 해결하는 디버깅 과정에서 자연스럽게 문제 해결과정을 전체적으로 관리할 수 있는 능력을 키워줄 수 있다는 것이다.

세 번째 범주는 프로그래밍 교육을 통하여 논리적인 추론과 표현 능력을 향상할 수 있다는 것이다. 컴퓨터 프로그래밍 과정을 통하여 현실 세계의 문제를 프로그래밍 언어로 표현하는 과정에서 자연스럽게 표현능력을 향상시킬 수 있으며 또한 논리적인 작업을 명령어로 표현하면서 자연스럽게 논리적인 사고능력과 추론능력을 향상시킬 수 있다는 것이다.

네 번째 범주는 프로그래밍 교육을 통하여 지식, 생각, 학습의 모델링 능력을 향상시킬 수 있다는 것이다. 즉 프로그래밍 과정은 정보를 가공해야 하고 처리를 해야 함으로 자연스럽게 모델링 능력을 향상시킬 수 있다. 이러한 모델링 능력을 통하여 정보를 가공하여 지식으로 발전시킬 수 있으며, 지식, 생각, 학습능력의 향상을 자연스럽게 기대할 수 있다.

다섯 번째 범주는 프로그래밍 교육을 통하여 인지유형을 만들 수 있다. 즉 프로그래밍이란 어떤 문제를 해결하기 위해서 프로그래머가 실제로 프로그래밍 언어를 통해서 구현하고 실행하여 원하는 결과를 얻는 것이다. 이러한 프로그래밍을 학생들이 직접 경험할

하게 되면 자신만의 인지 유형을 구축할 수 있다.

여섯 번째 범주는 프로그래밍 교육을 통하여 열정과 인내심을 기를 수 있다는 것이다. 프로그래밍이란 완벽한 정답이 나와야 의미가 있기 때문에 열정을 가지고 작업을 해야 하며 또한 결과가 나올 때까지 인내심을 가지고 작업을 해야 한다. 열정과 인내심이 없이는 방대한 프로그래밍 작업을 완성할 수 없기 때문에 열정과 인내는 프로그래머의 핵심 역량이라고 할 수 있다.

한편 [전우천2011]의 연구에서는 프로그래밍과 논리적 사고력의 상관관계를 분석하였다. 즉 서울시내 한 대학부설 과학영재교육원의 정보영재학생들을 대상으로 재학기간 중의 정보성적을 분석한 결과 프로그래밍 능력과 논리적 사고력 간의 유의미한 상관관계가 있음을 밝혀냈다. 즉 프로그래밍 능력이 우수할수록 논리적 사고력도 우수함을 입증하였다.

### 3. 소프트웨어교육에서의 프로그래밍 교육 현황

#### 3.1. 소프트웨어교육 교육목표

2015년 교육부에서 발표한 소프트웨어교육과정에서 초등학교, 중학교 및 고등학교과정에서의 소프트웨어 교육목표는 다음 <표 1>, <표 2>, <표 3>에 각각 표시된다[교육부2015].

<표 1> 초등학교 소프트웨어교육 교육목표

영역	학교급	초등학교
생활과 소프트웨어		소프트웨어가 가져온 생활의 변화를 알고, 정보사회에 필요한 건전한 의식과 태도를 가진다.
알고리즘과 프로그래밍		알고리즘과 프로그래밍을 체험하여 실생활의 다양한 문제를 컴퓨팅 사고로 이해할 수 있다
컴퓨팅과 문제해결		

<표 2> 중학교 소프트웨어교육 교육목표

영역	학교급	중학교
생활과 소프트웨어		소프트웨어 활용의 중요성을 알고, 정보 윤리의 개념을 이해하여 올바른 정보 생활을 실천하고, 정보를 교류할 수 있다.
알고리즘과 프로그래밍		간단한 알고리즘을 설계하고 프로그램을 개발하여 문제를 해결할 수 있다.
컴퓨팅과 문제해결		컴퓨팅 사고력에 기반하여 실생활 문제를 해결할 수 있다.

<표 3> 고등학교 소프트웨어교육 교육목표

영역	학교급	고등학교
생활과 소프트웨어		컴퓨팅 기술과 융합된 다양한 분야를 이해하고, 정보 윤리를 실천하며, 정보기기를 올바르게 조작할 수 있다.
알고리즘과 프로그래밍		알고리즘을 효율적으로 설계하고, 프로그램을 개발하여 창의적으로 문제를 해결할 수 있다.
컴퓨팅과 문제해결		컴퓨팅 사고를 기반으로 다양한 분야와 융합하여 문제를 해결할 수 있다.

#### 3.2. 영역별 지도 유의사항

3가지 영역 즉 ‘생활과 소프트웨어’, ‘알고리즘과 프로그래밍’, 또한 ‘컴퓨팅과 문제해결’에 있어서 지도 유의사항은 다음과 같다[교육부2015].

##### 1) 생활과 소프트웨어 영역

- 초등학교에서는 소프트웨어가 우리 생활에 미치는 변화를 주변의 다양한 사례를 통해 알아보고, 이와 더불어 정보 사회에 필요한 건전한 의식과 태도를 가지도록 지도한다.
- 중학교 <기본과정>에서는 소프트웨어의 종류와 특징을 서로 조사해 보고, 토론 기회를 제공하여 스스로 소프트웨어의 중요성을 인식할 수 있도록 지도한다. 이와 함께 소프트웨어와 관련된 다양한 직업 세계를 알고, 자신의 진로를 생각해 볼 수 있는 시간을 가지도록 지도한다. 또한, 개인정보와 지적 재산권의 개념을 동료들과 서로 설명할 수 있도록 지도한다. 소프트웨어를 운용하는 컴퓨터의 구성요

소를 알고 자료를 처리하는 과정에 대하여 토론하도록 지도한다.

- 중학교 <심화과정>에서는 개인정보와 지적재산권에 대한 이해를 토대로 실제 피해사례를 찾아 발표해 보고, 행동으로 실천할 수 있도록 지도한다. 또한, 컴퓨터의 운영체제와 네트워크의 각 역할을 이해하고, 이를 통해 실제로 정보를 교류해 보는 활동을 수행하도록 지도한다.

## 2) 알고리즘과 프로그래밍 영역

- 초·중학교에서의 알고리즘과 프로그래밍은 나선형 교육과정으로 조직되어 있으므로 학교급별 수준에 맞도록 단순한 요소에서 복잡한 요소로 확장되도록 구성하여 제시한다.
- ‘정보의 유형과 구조화’는 중학교 단계에서 다루며, <기본 과정>에서는 정보의 다양한 유형을 실제 사례를 탐색해 알아보고, 이런 정보들이 디지털로 표현된 사례들을 찾아 토론하도록 지도한다. <심화과정>에서는 이를 토대로 다양한 방식으로 정보를 구조화하는 방법을 알고 실제 문제를 해결하는 데 적용할 수 있도록 지도한다.
- 초·중학교에서의 ‘알고리즘과 프로그래밍’은 학습 동기를 유발할 수 있도록 학생들이 일상에서 경험할 수 있는 소재와 학교급에 맞는 적절한 수준의 문제 환경을 제시하고, 그 문제를 해결하는 과정에 초점을 두고 지도한다. 이를 위해 문제 해결 절차와 방법을 다양한 접근으로 시도해 보고, 이를 알고리즘으로 설계하여 프로그래밍으로 구현해 보는 일련의 활동으로 계획하여 지도한다.
- 문제를 해결하는 과정에서는 학생들이 서로 협력하여 해결할 수 있도록 계획하고, 각자의 역할을 충실히 수행할 수 있도록 지도한다.
- 동료들이 만든 프로그램이나 문제해결 결과에 대해 함께 이야기해보고, 주어진 문제를 해결하는 효율적인 방법에 대해 토론할 수 있도록 지도한다.
- 문제를 해결하는 과정과 알고리즘의 설계, 그리고 프로그래밍의 단계가 서로 관련되어 있음을 알게 하고, 그러한 절차가 컴퓨팅 사고력을 신장하는 과정임을 스스로 인식할 수 있도록 지도한다.

## 3) 컴퓨팅과 문제해결 영역

- 「컴퓨팅과 문제해결」 영역은 「생활과 소프트웨어」와 「알고리즘과 프로그래밍」 영역에서 익힌 개념과 지식을 바탕으로 실생활에서 마주하게 되는 문제를 컴퓨팅 사고력으로 해결할 수 있는 능력을 키울 수 있도록 지도한다. 이를 위하여 실생활의 문제 상황을 제시하고 문제를 인식하고 분석하는 단계에서부터 문제해결 절차에 따라 알고리즘을 설계하고, 이를 프로그램으로 구현하는 일련의 종합 활동이 이루어지도록 한다.
- 특정한 프로그래밍 언어의 습득에 치중되지 않도록 문제해결 절차에 초점을 두어야 하며, 주어진 문제를 해결하기 위하여 협력하여 프로그램을 개발할 수 있도록 지도한다.

## 3.3. 프로그래밍 교육현황

소프트웨어교육의 3가지 핵심영역 즉 ‘생활과 소프트웨어’, ‘알고리즘과 프로그래밍’, 또한 ‘컴퓨팅과 문제해결’은 모두 직접적으로 프로그래밍과 연관되어 있다.

먼저 ‘생활과 소프트웨어’ 영역에서는 소프트웨어의 중요성을 인식하고 다양한 소프트웨어의 활용 사례를 이해하며 직접적으로 현대사회에서 소프트웨어를 직접적으로 제작할 수 있는 프로그래밍 능력이 필수적임을 강조하고 있다.

둘째, ‘알고리즘과 프로그래밍’ 영역은 실제로 프로그래밍을 학생들로 하여금 경험하게 하는 가장 중요한 영역이다. 먼저 알고리즘에 정의와 구성방법을 통해서 프로그래밍에 있어서의 알고리즘의 중요성을 인식함과 더불어 실질적으로 알고리즘을 만드는 방법을 지도하게 한다. 프로그래밍 영역은 알고리즘을 바탕으로 특정한 프로그래밍 언어를 선택하여 프로그래밍을 구축하고 테스트하고 완성할 수 방법을 지도한다.

셋째, ‘컴퓨팅과 문제해결’ 영역은 학생들이 기본적인 프로그래밍 능력을 갖추었다는 전제하여 다양한 실제 생활에서의 문제 해결에 있어서 프로그래밍을 통한 문제해결능력을 강조한다. 예를 들어 자신에게 필요한 스마트 앱을 스스로 제작할 수 있는 능력을 갖게 함과 같다.

## 4. 프로그래밍 교육 활성화 방안

본 장에서는 향후 우리나라에서의 인공지능시대에 있어서 프로그래밍을 활성화할 수 있는 다양한 방안을 다음과 같이 제시한다.

### 1) 적절한 프로그래밍 언어의 선택

소프트웨어교육 운영지침에 의하면 가장 기본적인 프로그래밍 언어에 대해서 직접적인 언급이 없다. 현재 초등학교를 중심으로 다양한 교육용프로그래밍언어(EPL: Educational Programming Language)를 가르치고 있다. 스크래치가 가장 대표적인 언어라고 할 수 있다. 하지만 [지디넷코리아2017]에 따르면 소프트웨어교육에서 추구하는 궁극적인 소프트웨어 생산자(Maker)로 기르기 위해서는 EPL로는 부족하고 고급언어인 자바 또는 C++을 지향해야 하는 데, 초중등학생들에게 자바와 C++은 난이도가 너무 높다는 것이다. 최근 사교육현장에 Python을 통하여 공교육에서의 소프트웨어교육을 대비하는 움직임이 일어나고 있다.

### 2) 다양한 피지컬 컴퓨팅 사례 발굴 및 도입

프로그래밍 교육을 실시할 경우 자칫하면 프로그래밍 언어의 문법만을 강조하여 가르칠 수 있어서 학생들의 계산적 사고력을 향상시키는 것을 등한히 할 수 있다. 계산적 사고력의 향상과 더불어 프로그래밍 능력을 실질적으로 향상시키기 위해서는 무엇보다도 현실적인 문제해결을 지향해야 하며 학교 현장에서 피지컬 컴퓨팅의 다양한 사례를 도입하여 프로그래밍 교육의 현실생활로의 적용 가능성을 교육해야 한다. 이를 위해서는 다양한 피지컬 컴퓨팅 사례를 발굴해야 하며 이와 더불어 단순히 기기만을 다루는 것이 아니라 컴퓨팅 사고력 및 알고리즘 작성능력을 키워주는 다양한 사례를 개발해야 한다.

### 3) 다양한 정보경시대회 지원

프로그래밍 문화 확산은 소프트웨어교육 등 제한된 공교육과정에서 해결하기에는 무리라고 할 수 있다. 공교육과 더불어 다양한 방과후학교 수업, 캠프, 또한 정보경시대회를 개최하고 후원할 필요가 있다. 정보관

련 경시대회의 경우 대부분 2가지 종류로 구분되어 있다. 즉 다양한 응용소프트웨어 제작도구를 통한 산출물대회와 더불어 프로그래밍을 위한 경시대회로 구분할 수 있다. 프로그래밍 경시대회의 경우 대부분의 지원자가 사교육을 통한 프로그래밍 수업을 통해서 준비할 수 밖에 없는 상황이어서 제한된 학생들만이 응시할 수 밖에 없는 문제가 있다.

향후 사교육을 통한 프로그래밍 사전교육이 없이도 계산적 사고력 또는 소프트웨어 사고력을 겨룰 수 있는 다양한 대회를 개최 및 지원할 필요가 있다. 이러한 형태의 대회는 궁극적으로 프로그래밍 위주의 경시대회와도 상생할 수 있는 형태로 발전할 수 있다.

### 4) 교사들을 위한 다양한 연수 기회 제공

초중고등학교 프로그래밍 교육의 경우 정보담당 교사의 수가 절대적으로 부족하고 특히 초등학교의 경우 한정된 시수를 이수한 교사가 대부분이어서 현직 교사로서 프로그래밍을 실질적으로 담당하는 것이 많은 부담이 될 수 있다.

따라서 현직교사에 대한 다양한 프로그래밍 관련 연수기회를 제공하고 다양한 프로그램을 제공할 필요가 있다. 프로그래밍의 경우 단기간의 연수를 받고 바로 학생들을 가르치는 데에는 한계가 있기 때문에 지속적이고 충분한 시간을 할애하는 연수기회를 제공할 필요가 있다.

### 5) 프로그래밍 교수-학습 방법 개발 및 보급

학교현장에서 프로그래밍 수업을 실제적이고 효율적으로 가르치기 위해서는 무엇보다도 프로그래밍 교수-학습방법의 개발 및 보급이 필요하다. 현재 다양한 프로그래밍 수업 모형이 개발되고 있으나 그 효과성에 대한 검증은 많은 실험대상과 실험기간을 필요로 한다.

현재 학교현장에서는 소위 “꺼꾸로 수업(Flipped Learning)” 등 다양한 미래지향 수업이 등장하고 있다. 이와 같은 미래지향 수업모형과 연계하여 학생들의 흥미와 의욕을 높일 수 있는 다양한 프로그래밍 수업 모형 개발이 시급하다.

## 5. 결 론

4차 산업혁명의 도래로 인하여 우리의 삶이 근본적으로 변하고 있다. 즉 국가적, 사회적, 개인적 차원에서 4차 산업혁명은 우리의 경제, 국방, 사회, 직업 등 다양한 분야에 직접적인 영향을 미치고 있으며 우리가 원하던 원하지 않던 4차 산업혁명에 대한 이해는 우리의 삶을 더욱 풍성하게 할 수 있다.

이러한 4차 산업혁명의 중심에는 인공지능이 자리 잡고 있음은 누구도 부인할 수 없는 현실이 되고 있다. 인공지능은 특정한 기술이 아니라 다양한 기술이 융합된 일종의 상위개념으로 4차 산업혁명을 주도하고 있다. 현재 인공지능에 대한 내용이 공교육에 반영되지 못한 실정이며 인공지능 관련내용이 공교육에 반영되기 위해서는 다양한 노력과 더불어 다른 과목과의 갈등도 예상되고 있다.

현실적으로 인공지능을 다룰 수 있는 교과내용은 소프트웨어교육이 가장 가깝고 특히 소프트웨어교육의 핵심인 프로그래밍 교육은 인공지능 이해의 기본 교육이자 정보교육의 핵심이 되고 있다. 2018년부터 의무화 되는 소프트웨어교육을 인공지능시대에 맞게 다양한 방법으로 운영할 필요가 있다.

본 원고에서는 인공지능시대에 프로그래밍 교육을 활성화할 수 있는 방안을 제시하였다. 크게 적절한 프로그래밍 언어의 선택 문제, 다양한 피지컬 컴퓨팅 사례 발굴 및 도입, 다양한 정보경시대회 지원, 교사들을 위한 다양한 연수기회 제공, 프로그래밍 교수-학습 방법 개발 및 보급 등이다.

프로그래밍 문화 확산은 현재 공교육보다는 사교육

에서 활성화되고 있는 형편이며 다양한 경시대회, 소프트웨어교육 관련 캠프, 특별 강연, 진로교육 등에서 활성화되고 있다. 이러한 사교육 차원에서의 확산은 매우 긍정적인 현상으로 궁극적으로는 공교육에도 긍정적인 영향을 미칠 수 있다. 현재 2018년부터 의무화 될 예정인 소프트웨어교육에 대한 다양한 교육과정 정비 및 운영방안이 논의되고 있다. 프로그래밍 교육을 비롯한 소프트웨어교육이 활성화되기 위해서는 무엇보다도 학문적인 기초가 튼튼해야 한다. 즉 다른 과목에 비해서 상대적으로 역사가 짧기 때문에 교육방법, 교육내용, 교육평가 등이 아직은 미약한 실정이다. 향후 소프트웨어교육에 대한 다양한 교육과정 정비, 교육내용 현실화, 교수-학습 방법 개발, 평가방법 개발 등이 필요하다고 할 수 있다.

## 참 고 문 헌

- [1] 교육부, “소프트웨어교육 운영지침”, 교육부, 2015.
- [2] 김갑수, “초등학생들의 창의력과 논리력 향상을 위한 프로그래밍 언어 교수전략에 관한 연구”, 정보교육학회논문지, Vol. 14, No. 1, pp. 89-97, 2010.
- [3] 전우천, “정보영재아동의 프로그래밍 능력과 논리적 사고력 상관관계 분석 연구”, 영재교육연구, Vol. 21, No. 3, 761-772, 2011.
- [4] 지디넷코리아, “어떤 프로그래밍 언어를 배워야 할까?”, [http://www.zdnet.co.kr/news/news\\_view.asp?article\\_id=20161230153750](http://www.zdnet.co.kr/news/news_view.asp?article_id=20161230153750), 2017년 1월 1일자

## ● 저 자 소 개 ●

### 전 우 천

1985년 서강대학교 전산학 학사  
 1987년 서강대학교 대학원 전산학 석사  
 1997년 미국 University of Oklahoma 전산학 박사  
 1998년~현재 서울교육대학교 컴퓨터교육학과 교수  
 관심분야 : 정보영재, 장애인정보화교육, 정보통신윤리교육

