

# 자료구조 수업에서 온라인 자동평가용 문제해결 프로그래밍 문제은행 개발 및 적용

김성식<sup>†</sup> · 오소희<sup>††</sup> · 정상수<sup>†††</sup>

## 요 약

본 연구에서는 교원양성 대학에서 정보과학적 문제 해결력의 기본이 되는 자료구조 강의를 수강하는 학생들에게 학습 동기를 부여하고 몰입도를 증가시킬 수 있는 방안의 하나로써, 문제해결 프로그래밍 자동평가 시스템을 적용하기 위해 자료구조의 주요 토픽별 문제 은행을 개발하였다. 자료구조의 주요 토픽에 적합한 프로그래밍 평가 문제은행 70여 문항을 개발하여, 이를 온라인 자동평가 시스템에 탑재하고, 실제 수업에 적용하였다. 수업 적용 후 결과에 따른 학습동기와 몰입도 등을 분석함으로써, 교원양성 대학에서 문제해결 프로그래밍, 자료구조 수업에서 활용할 수 있는 ‘학습 동기를 부여하며, 몰입하여 해결하는’ 교수-학습 콘텐츠와 활용방법을 제시하였다.

**주제어** : 자료구조, 알고리즘, 문제해결 프로그래밍, 문제은행, 온라인 자동평가 시스템

## Development and Application of Problem Bank of Problem Solving Programming Using Online Judge System in Data Structure Education

Seong-Sik Kim<sup>†</sup> · So-Hee Oh<sup>††</sup> · Sang-Su Jeong<sup>†††</sup>

## ABSTRACT

This study is to propose a problem bank of problem solving programming using Online Judge System as one of the ways to motivate learners and increase for immersion to students who take Data Structure lecture that is the basis of problem solving ability using information science. In order to do this, we developed a question bank for each major topic in the Data Structure, by developing 70 problem solving programming problems suitable for the main topics of the Data Structure. By mounting it on an Online Judge System and applying to actual classes, and by analyzing the motivation for learning and the degree of immersion according to the result after the application of the lesson, we propose a teaching-learning contents and usage for problem solving programming and Data Structure classes at the teacher training university which give motivation for learning and immerse in problem solving programming.

**Keywords** : Data Structure, Algorithm, Problem Solving Programming, Problem Bank, Online Judge System

† 종신회원: 한국교원대학교 컴퓨터교육과 교수(교신저자)  
 †† 정 회 원: 한국교원대학교 컴퓨터교육과 박사과정  
 ††† 정 회 원: 한국교원대학교 컴퓨터교육과 석사과정

논문접수: 2018년 6월 19일, 심사완료: 2018년 7월 28일, 게재확정: 2018년 7월 29일

\* 본 논문은 2017년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임(NRF-2017S1A5A2A01026374).

## 1. 서론

2015년 9월 소프트웨어(SW)교육의 강화를 위해 교육과정이 개정되어 올해는 중학교 1학년과 고등학교 1학년, 내년에는 초등학교와 중등학교에서는 순차적으로 적용 되어질 계획이다. 여전히 학교급별 수준에 적합한 교수·학습 방법과 소프트웨어교육의 목표에 대한 논의가 이루어지고 있다. 중학교에서는 블록 기반 언어를 고등학교에서는 텍스트 기반 언어를 사용하도록 권장하고 있다. 고등학교 과학 계열 전문 교과인 ‘정보 과학’에서는 고등학교 ‘정보’보다 높은 수준의 알고리즘과 프로그래밍에 대한 내용요소가 포함되어 있다 [1][2][3]. 요즘의 많은 대학에서는 전공자가 아닌 비전공자를 대상으로 한 프로그래밍(알고리즘) 교육을 강화하고 있다. 이와 같은 주변 환경의 변화에 따라 초·중등교사를 양성하는 교원양성대학에서 SW교육의 중요성이 높아지고 있다.

소프트웨어(SW)교육의 목표는 ‘컴퓨팅 사고력(Computational Thinking)의 함양’이다. 지난 교육과정과 비교해서 2015 개정 정보 교육과정에서는 기본 역량으로 ‘컴퓨팅 사고력’을 규정하고, 정보문화에 대한 기본 소양과 협력적 태도를 강조하고, ‘문제해결과 프로그래밍’ 영역을 보다 강화하였다. 그리고 실생활과 거리가 멀거나 어려운 내용은 축소하거나 삭제하였다. 그리고 학교급별로 학습자의 수준에 맞는 교수·학습 방법을 제시하였다.

학교급별로 중점을 둔 부분을 살펴보면 초등학교에서는 프로그래밍을 할 때 놀이를 통해서 알고리즘을 이해하고 교육용 도구를 활용하고, 중학교에서는 실생활의 문제 상황을 프로그래밍 도구를 활용하여 해결하고, 고등학교에서는 진로와 연계된 심화된 내용을 학습할 수 있도록 텍스트 기반 프로그래밍 언어의 활용을 제시하고 있다[2]. ‘정보 과학’에서는 복합적이고 다양한 문제들을 해결하기 위해 알고리즘을 설계하고 분석하는 과정을 강조하고 있으며, 자기주도적인 학습과 수준별 학습에 편리하도록 프로그래밍 수업에서 자동평가 시스템의 활용을 제안하고 있다[3].

초·중등학교의 예비 교사를 양성하는 교원양성대학에서도 이러한 2015 개정 교육과정의 변화 내용을 반영하여 예비 교사들이 프로그래밍과 정

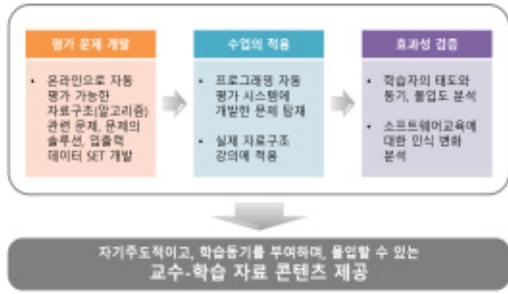
보 과학적 문제 해결 능력을 함양할 수 있도록, 교육과정을 개편하고, 교육내용 및 교수·학습의 변화가 요구된다. 일반적으로 정보·컴퓨터 예비교사를 양성하는 사범대학 컴퓨터교육과의 교육과정을 살펴보면 1학년에는 C, C++등 기본적인 프로그래밍 언어를 편제하고, 2학년에는 자료구조, 3학년에는 알고리즘 과목을 편제함으로써, 정보과학적 문제해결력을 함양할 수 있도록 한다. 그러나 많은 학생들이 프로그래밍 과정에 어려움을 호소하고 있으며, 특히 자료구조와 알고리즘 과목을 기피하는 경향이 있는 것도 사실이다. 오경선, 안성진의 연구에 따르면 프로그래밍을 처음 접하는 대학생들을 대상으로 조사한 결과를 다음과 같이 분석하였다.

<표 1> 단계별 어려움의 정도 (오경선, 2015)

요인	어려움의 정도				
	매우 낮음	낮음	보통	높음	매우 높음
문제 이해 단계	0.65%	21.3%	41.08%	30.88%	6.15%
문제 분석 단계	0.75%	13.95%	42.17%	34.77%	8.43%
알고리즘적 사고 단계	1.47%	12.43%	45.1%	32.93%	8.07%

오경선, 안성진(2015)의 연구 결과에 따르면 프로그래밍을 처음 접하는 학생들의 경우 모든 단계(문제이해, 문제분석, 알고리즘적 사고)에서 보통 이상의 어려움을 느끼고 있었다. 프로그래밍 언어를 사용하여 코드를 작성하는 모든 과정에서 어려움을 느끼고 있음을 알 수 있다. 특히, 문제의 분석 및 알고리즘적 사고의 단계에서는 85% 이상이 어려움을 느끼고 있다는 응답을 보였다[4].

본 연구는 교원양성대학에서 접할 기회가 적은 문제해결 프로그래밍을 전공필수과목인 자료구조 수업을 통해서 다양한 문제해결 프로그래밍을 자기주도적으로 해결하거나 동료학습을 통해 문제해결력이 향상될 것이다. 이를 바탕으로 자료구조 강의를 수강하는 학생들에게 프로그래밍(알고리즘)에 대한 학습 동기를 부여하고 몰입도를 증가시킬 수 있는 방안의 하나로서, 프로그래밍(알고리즘) 자동평가 시스템을 적용하고, 이를 위해 자료구조의 주요 토픽별 문제 은행을 개발하고자 한다.



[그림 1] 연구의 목적

본 연구는 자료구조의 주요 토픽에 적합한 평가 문제를 개발하되, 온라인에서 자동으로 평가할 수 있도록 문제, 문제의 솔루션, 입출력 데이터 세트 등을 개발하고자 한다. 이를 프로그래밍(알고리즘) 자동평가 시스템에 탑재하여 실제 자료구조 수업에 적용하여 학습자의 태도와 인식, 동기와 몰입도 등을 분석함으로써 교원양성 대학에서 프로그래밍, 자료구조, 알고리즘 등의 수업에서 활용할 수 있는 ‘자기주도적이고’, ‘학습 동기를 부여하며’, ‘몰입할 수 있는’ 교수-학습 자료 콘텐츠를 제공함을 목적으로 한다.

## 2. 이론적 배경

### 2.1 프로그래밍(알고리즘) 자동평가 시스템

프로그래밍(알고리즘) 자동평가 시스템(이하, 자동평가 시스템)은 주어진 문제에 대해서 학습자가 문제를 해결하여 소스코드를 제출하게 되면 컴파일 시간과 실행 시간에 대한 오류를 확인한다. 그리고 소스코드가 정확하고 알고리즘이 효율적으로 구현되었는지의 여부를 자동으로 평가한다. 따라서 학습자는 제출한 소스코드에 대한 정답 여부를 즉각적으로 알 수 있는 시스템이다.

#### 2.1.1 자동평가 시스템의 원리

일반적으로 2단계의 과정을 거치는데, 컴파일 과정과 채점 과정으로 이루어져있다. 컴파일 과정에서는 학습자가 주어진 문제에 대한 소스코드를 제출하면 자동평가 시스템은 해당 코드를 서버에 저장하고, 자동으로 컴파일 하여 실행 파일을 만

든다[5]. 이와 같은 컴파일 과정을 통과한 실행 파일은 이미 저장되어 있는 입출력 데이터의 세트에 따라 실행을 반복한다. 입출력 데이터 세트의 데이터가 입력되면 출력 되는 값과 저장되어 있는 출력 값을 비교한다. 이 두 개의 값이 같고, 시간과 메모리 제한을 초과하지 않으면 학습자가 제출한 소스 코드는 올바른 풀이 과정으로 판단한다. 미리 준비된 채점데이터 세트 내에서는 소스코드는 알고리즘이 정확하고 효율적이라는 의미이다[6][7].

### 2.1.2 자동평가 시스템 문제의 구성요소

임형석(2007)에 따르면 알고리즘 및 자료구조 자동평가 시스템에서 사용하는 문제는 기본적으로 <표 2>와 같이 구성요소로 이루어져 있으며, 각 구성요소들은 알고리즘의 기본조건인 명확성, 유용성, 유한성, 효율성을 고려하여 설계 한다[8].

<표 2> 문제 구성요소(임형석, 2007)

구성요소	기본조건		명확성
배경 설명			
핵심요구사항	정확성(유효성)		
제한 시간	시간효율성	유한성	
입력데이터의 범위			
입력과 출력의 예			

### 2.1.3 채점데이터 세트

채점데이터 세트는 입력데이터와 정답데이터의 쌍으로 구성된 20개 이내의 데이터 세트로서 입력데이터와 정답데이터 사이의 종적인 구성으로 알고리즘을 정확성을 평가하고 입력데이터의 범위가 점점 증가하는 횡적인 구성으로 알고리즘의 시간/공간효율성을 평가한다. 그리고 이러한 평가요소는 <표 2>의 문제 구성요소에서 핵심요구사항, 제한사항, 입력데이터의 범위와 밀접한 관련이 있다[8].

## 2.2 자동평가 시스템의 활용

프로그래밍(알고리즘) 자동평가 시스템은 최근에 국·내외에서 사용자층이 많아지고 시스템 내의 자동평가 건수가 지속적으로 증가하고 있다. 우리나라의 경우 현직 교사들이 오픈소스 환경인

hustoj를 활용하여 직접 시스템을 구축하여 교실 현장에서 활용하고 있다[9]. 최근에는 교사 연수 프로그램에서도 자동평가 시스템을 활용하는 사례가 계속 증가하고 있다[10].

### 3. 프로그래밍 자동평가 문제은행 개발

#### 3.1 문제은행 개발 원칙

시스템 개발은 장원영(2017)의 연구에서 진행되었기에 본 연구에서는 모든 사람이 사용할 수 있는 오픈소스 자동채점 시스템에서 사용할 수 있는 문제은행을 개발하여 실험 운영함으로써 많은 교원양성대학에서 사용자를 확대하려는 것이다.

첫째, 우리가 사용하는 오픈소스 자동채점 시스템에서 사용할 수 있도록 개발한다.

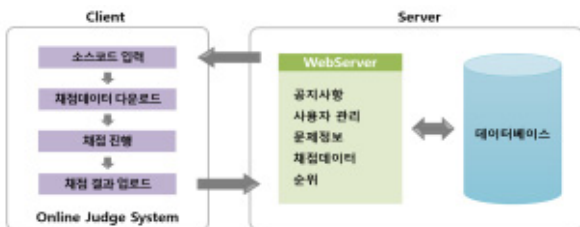
둘째, 문제를 주제별로 골고루 선정하여 주제별 5문항 정도를 출제한다.

셋째, 학습자의 수준에 맞도록 상중하로 수준별로 해결할 수 있는 문제를 골고루 출제한다.

넷째, 데이터 셀은 10개 이내로 하되 범위는 10 단위부터 1억 단위까지 산출하고 입력데이터의 예외적인 처리도 테스트 할 수 있도록 1~2개 정도 엉뚱한 데이터를 산출한다.

다섯째, 실험 운영을 하면서 개발한 문제은행의 난이도를 조절한다.

#### 3.2 자동평가 시스템의 구성



[그림 2] 시스템 구성도

자동채점 시스템은 [그림 2]와 같은 구성도를 가진다. 우선 데이터베이스에는 학습자들의 정보나 채점을 위한 데이터 세트 등이 저장되어있다. 웹서버를 통해 데이터베이스의 내용을 볼 수 있다. 학습자가 소스코드를 입력하면 채점데이터를

다운로드한 후 채점을 진행한다. 그리고 채점이 완료된 후에는 채점 결과를 업로드 한다. 이때 채점 결과는 데이터베이스에 저장되고 웹서버를 통해 확인 할 수 있다.

#### 3.3 문제은행 개발 과정

문제은행 개발은 다음과 같은 과정으로 진행되었다. 연구자 및 연구보조자 등 석·박사에 재학 중인 현직 교사들이 모여 의논하며 진행하였다.

첫째, 문항 수를 정한다. 각 주마다 제공될 문항수를 정하고, 어느 정도의 주기, 기간 등 문제를 어떻게 제공할 것인지 정한다.

둘째, 문항별 주제를 정한다. 자료구조 수업에서 사용될 교재를 바탕으로 기초부터 개념을 쌓아갈 수 있도록 영역을 구성하고 순서를 정한다.

셋째, 난이도를 고려하여 문항 내용을 구성한다. 한 회차에 문제가 제시될 때 쉬운 문제로만 구성되어있거나 너무 어려운 문제로만 구성되지 않도록 골고루 난이도를 정한다. 자료구조의 특성상 후반부로 갈수록 어려워지는 경향이 있는데 이 때는 보다 난이도가 낮은 문항 내용으로 구성한다.

넷째, 문제를 제작한다. [그림 8]과 같이 문제를 제작한다. 문제 상황과 입출력 예시를 제공하여 학습자가 문제를 분석하고 설계한 후 프로그래밍하여 문제를 해결해 나갈 수 있도록 문제를 제작하였다. 문제 제작시 자동채점 시스템에 업로드 할 채점 데이터 세트도 함께 제작한다.

#### 3.4 문제은행 개발 및 자료구조에 활용

본 연구는 배열, 스택, 큐, 그래프, 리스트 등의 자료구조를 적용하여 해결할 수 있는 문제를 제시하거나 동적계획법, 정렬, 탐색, 그래프 탐색(DFS, BFS) 등의 알고리즘을 적용하는 문제를 제시하고, 학생들이 제출한 소스 코드를 자동평가를 하는 것이다.

본 연구에서는 15개 주제에 주제별 5개 문항을 개발하였으며 그 중 일부를 11주차에 걸쳐 난이도를 고려하며 실험 운영하였다. <표 3>은 자료구조 수업에서 제시된 학습 내용을 바탕으로 제시된 문제의 일부이다. 기본적인 개념을 통해 자

료구조의 본질을 이해하고 문제 해결의 응용력을 향상 시킬 수 있도록 문제를 개발하였다.

<표 3> 자료구조와 알고리즘 적용 문제

영역	문제	차시
동적 계획법	n개월 후 토끼의 수구하기	1차시
	하노이 타워	
	팩토리얼 구하기	
배열	1부터 n까지 역순으로 출력하기	4차시
	2차원 배열 채우기	
스택	홀수 마방진	5차시
	지뢰찾기	
	표기식 변환하기	
큐	하나의 배열에 2개의 스택	6차시
	올바른 괄호	
	교원회의 은행 나들이1	
리스트	미로 탐색	7차시
	벌레 X	
	번덕이 심한 해적	
트리	회문	8차시
	삽입 삭제	
	레벨 순회	
그래프	최소 힙 트리	9차시
	이진탐색트리의 삽입 삭제	
	누적 합 트리	
정렬	다익스트라 알고리즘	10차시
	벨만-포드의 최단 경로 알고리즘	
	무향 그래프와 유향 그래프	
탐색	정렬(리스트, 테이블)	11차시
	테이터 재정렬	
	성적순으로 정렬하기	
탐색	보물찾기	11차시
	연속된 구간의 최대 합	
	주어진 구간에서 최댓값	
	달리기 순위	

또한 자동평가 시스템은 알고리즘의 성능을 평가하는 객관적인 자료를 제공한다. 대부분의 전공 서적에서는 ‘시간 복잡도’와 ‘공간 복잡도’에 설명하고 있는데 학생들이 이해하기는 어려운 개념 중 하나이다. 자동평가 시스템은 실제 수행 시간과 메모리 사용량을 보여줌으로써 알고리즘의 성능과 효율성에 대해 깊이 이해할 수 있도록 도와주고 있다([그림 3] 참고).



[그림 3] 채점 상황판

### 3.5 평가의 효율성과 수월성

학생들이 제출한 소스코드에 대해서 객관적이고 정확한 평가는 교수자에게 많은 부담을 안겨준다. 하지만 자동평가 시스템은 학습자가 제출한 소스코드의 크기와 개수에 상관없이 정확하고 객관적으로 평가가 가능하다.

## 4. 프로그래밍 자동평가 시스템 적용

### 4.1 자동평가 시스템의 적용 결과

자동평가 시스템을 개발을 완료한 후 연구 대상자에게 적용하였다. 수업 시작 전 연구 대상자에게 본 시스템의 사용법을 안내하였다. [그림 5]는 학습자에게 제공된 과제의 목록이다. 수업 시간에 공지된 시간에 과제가 오픈되고 다음 수업이 시작되기 하루 전 마감이 된다. 마감이 된 과제는 문제를 풀 수도 없고 문제 열람도 할 수 없다.

ID	Name	Status
1016	부채금리 계산하기	Finished
1015	배열의 곱셈하기	Finished
1014	부채금리 계산하기	Finished
1013	순차적 곱셈하기	Finished
1011	순차적 곱셈하기	Finished
1010	순차적 곱셈하기	Finished
1009	순차적 곱셈하기	Finished
1008	순차적 곱셈하기	Finished
1004	순차적 곱셈	Finished
1003	순차적 곱셈하기	Finished
1002	순차적 곱셈하기	Finished

[그림 5] 전체 과제 목록

[그림 6]의 해당 주차의 과제 목록에서 “Status”를 클릭하면 해당 문제들에 대한 채점 상황을 확인할 수 있고, “Standing”을 클릭하면 [그림 7]과 같은 제출 순위와 제출 여부를 확인할 수 있다. 채점 결과 및 순위를 공개함에 따라 몰입 및 경쟁이 되도록 하였다.

문제 ID	문제명	출제	AC	채점
1019	두 다항식의 곱 (하)	배열	29	128
1020	제한제일 (상)	배열	32	68
1022	다항식의 덧셈(중상)	배열	29	82

[그림 6] 3주차 과제

[그림 8]은 과제 목록에서 문제를 선택하였을 때 학습자들이 보게되는 문제 화면이다. 문제 화면의 하단에 있는 “제출”을 클릭하면 [그림 9]와 같은 화면에 소스코드를 입력하여 제출하면 된다.

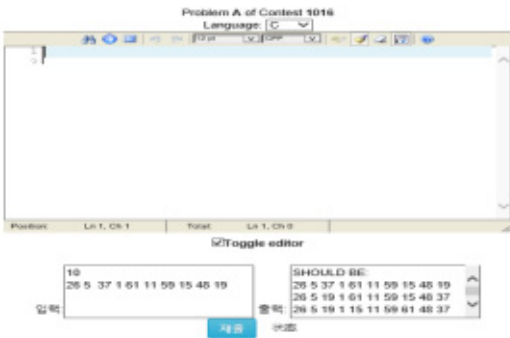
Rank	Priority	소	중	대
1	01111111	00100111	00110011	00000100
2	01142102	00101100	00100100	00111100
3	0243108	01100100	01100100	00010100
4	0200000	01100101	01100100	00010100
5	0248104	01100101	01100100	00010100
6	0411220	01100101	01100100	00010100
7	0412100	01100101	01100100	00010100
8	0412100	01100101	01100100	00010100
9	0412100	01100101	01100100	00010100
10	0412100	01100101	01100100	00010100
11	0412100	01100101	01100100	00010100
12	0412100	01100101	01100100	00010100
13	0412100	01100101	01100100	00010100
14	0412100	01100101	01100100	00010100
15	0412100	01100101	01100100	00010100
16	0412100	01100101	01100100	00010100
17	0412100	01100101	01100100	00010100
18	0412100	01100101	01100100	00010100
19	0412100	01100101	01100100	00010100
20	0412100	01100101	01100100	00010100
21	0412100	01100101	01100100	00010100
22	0412100	01100101	01100100	00010100
23	0412100	01100101	01100100	00010100
24	0412100	01100101	01100100	00010100
25	0412100	01100101	01100100	00010100
26	0412100	01100101	01100100	00010100
27	0412100	01100101	01100100	00010100
28	0412100	01100101	01100100	00010100
29	0412100	01100101	01100100	00010100
30	0412100	01100101	01100100	00010100
31	0412100	01100101	01100100	00010100
32	0412100	01100101	01100100	00010100
33	0412100	01100101	01100100	00010100
34	0412100	01100101	01100100	00010100
35	0412100	01100101	01100100	00010100
36	0412100	01100101	01100100	00010100
37	0412100	01100101	01100100	00010100
38	0412100	01100101	01100100	00010100
39	0412100	01100101	01100100	00010100
40	0412100	01100101	01100100	00010100
41	0412100	01100101	01100100	00010100
42	0412100	01100101	01100100	00010100
43	0412100	01100101	01100100	00010100
44	0412100	01100101	01100100	00010100
45	0412100	01100101	01100100	00010100
46	0412100	01100101	01100100	00010100
47	0412100	01100101	01100100	00010100
48	0412100	01100101	01100100	00010100
49	0412100	01100101	01100100	00010100
50	0412100	01100101	01100100	00010100

[그림 7] 채점 결과

그리고 “채점 상황”을 클릭하면 [그림 3]과 같이 해당 문제에 제출한 사용자들의 순위를 확인 할 수 있는데, 순위는 메모리를 적게 사용한 소스코드일수록 순위가 높다. 메모리 사용량이 같다면 코드 길이가 짧을수록 순위가 높다.



[그림 8] 문제 화면



[그림 9] 소스코드 제출 화면

3.3의 [그림 3]은 데이터베이스에 입력된 문제 번호 1044번의 제출 상황판이다. 원그래프는 채점 결과에 따른 그래프로 AC는 정확한 풀이, PC는 제출 오류, WA는 잘못된 풀이, RE는 런타임 에러, CE는 컴파일에러를 나타낸다.

### 5. 연구 방법

#### 5.1 연구 문제

본 연구는 다음과 같이 연구 문제를 설정하였다. [연구 문제 1] 예비 교원의 정보과학적 사고력 함양을 위해 자료 구조 수업에 적용할 수 있는 온라인 자동평가 방식에 적합한 문제는 무엇인가? [연구 문제 2] 위의 온라인 자동평가 방식의 문제들은 학습자에게 동기를 부여하고 몰입도 증가에 효과가 있는가?

이러한 연구 문제를 해결하기 위해 첫째, 자료 구조의 영역별 주요 토픽을 선정하고 예비 문제를 도출하였다. 둘째, 학습자의 동기를 부여하고 몰입도를 증가시키기 위해 온라인 자동평가 문제 은행을 개발하였다. 셋째, 개발한 문제 은행을 적용하고 동기, 몰입의 효과성을 검증하였다.

#### 5.2 연구 방법

자료구조의 주요 토픽에 적합한 평가 문제를 개발한다. 이때 온라인에서 자동으로 평가할 수 있도록 문제, 문제의 솔루션, 입출력 데이터 세트 등을 체계적으로 개발한다. 이를 프로그래밍(알고리즘) 자동평가 시스템에 탑재하여 실제 자료구조 수업에 적용한다. 소프트웨어교육 태도 검사지를 활용하여 수업 전-후 학습자의 태도와 인식, 동기 와 몰입도 등의 변화를 분석한다.

#### 5.3 연구 대상

본 연구의 대상은 K대학 컴퓨터교육과에 2018 학년도 1학기 전공 교과목으로 개설된 자료구조를 수강하는 학부생 24명이다. 강의실의 여건상



하나의 집단으로 사전검사와 사후검사를 진행하였고, 24명의 학생 중 무작위로 5명을 뽑아 평균 1달에 1회씩 총 4회에 걸친 면담을 진행하였다. 사전검사와 사후검사는 학습 동기와 몰입에 대한 문항으로 구성한다.

### 5.4 검사도구 및 분석

사전검사와 사후검사는 동일한 검사지로 진행되었다. 소프트웨어교육태도 검사 도구는 수학적 학습 태도 검사지를 이진영(2010)이 컴퓨터 교과에 맞게 수정한 검사지를 활용하여 소프트웨어교육 태도로 번안하여 사용하였다[11]. 해당 검사 도구는 수업태도, 코딩흥미, 몰입과 경쟁, SW교육자의 자세의 4개 영역이며, 영역 별로 각 6 문항씩 총 24문항으로 구성되어 있다. 응답 범주로는 ‘전혀 그렇지 않다(1점)’, ‘그렇지 않다(2점)’, ‘보통이다(3점)’, ‘그렇다(4점)’, ‘매우 그렇다(5점)’의 5개 범주로 되어있다. 검사 결과의 통계 처리를 위하여 대응표본 t-검정을 실시하였다.

## 6. 연구결과

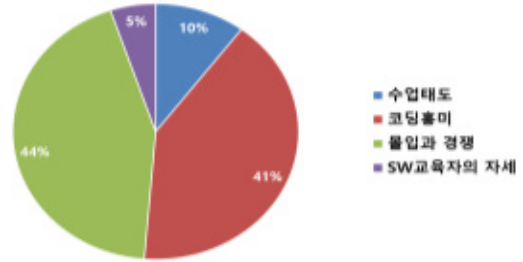
### 6.1 사전-사후 검사결과

프로그래밍(알고리즘) 자동평가 시스템을 적용한 자료구조 수업에서 대학생의 소프트웨어교육 태도 변화에 대해 확인해 보기 위해서 연구대상자의 사전-사후 검사 결과를 비교하였다. 소프트웨어 교육에 대한 사전-사후 검사 결과를 대응표본 t-검정한 결과는 <표 4>와 같다. 사전-사후검사를 비교한 결과, 교육자의 자세를 제외한 그 외의 하위 요인별 유의수준 .05에서 통계적으로 유의미한 차이를 보였다.

<표 4> 소프트웨어교육 태도 사전-사후 결과

하위 요인	사전검사		사후검사		t	p
	평균	표준편차	평균	표준편차		
수업태도	18.917	3.988	20.667	3.409	-3.329	.003
코딩흥미	19.750	3.274	20.667	4.050	-2.381	.026
몰입과 경쟁	17.792	4.634	19.917	4.373	-3.543	.002
교육자의 자세	22.333	2.869	22.500	3.230	-.457	.652
합계	77.958	13.392	83.250	13.854	-3.588	.002

따라서 본 연구는 유의미한 효과가 있는 것으로 확인되었다. 사후검사에서는 자동평가 시스템을 적용한 자료구조 수업에서 도움이 되었던 부분은 무엇인지에 대한 중복응답이 가능하도록 추가 질문을 하였다. 상위 2개 요소는 몰입과 경쟁(44%), 코딩흥미(41%)라고 답하였다([그림 10] 참고).



[그림 10] 자동평가 시스템 활용 시 도움이 된 요소

### 6.2 학습자 면담 내용 분석

24명의 학생 중 무작위로 5명을 뽑아 월 1회씩 4개월에 걸쳐 총 20회에 걸친 학습자 면담을 진행하였다. 면담에 참여한 학습자의 의견을 통합하여 분석한 결과이다.

#### 6.2.1 코딩흥미

자동채점 시스템을 통해 과제를 제출하면서 코딩에 대한 흥미가 생겨 긍정적인 반응을 보인 학습자들이 있었다. 반면 문제가 어려워서 과제를 하는 동안 너무 스트레스를 받은 학생도 있었다.

학생 3 : 과제를 제출하면 바로 정답 유무를 알 수 있고, 문제에 따라 정확히 한 개의 풀이만 있는게 아니고 제가 제 스타일대로 짜도 정답으로 인정되니까 좋았어요.

학생 4 : 중간고사나 기말고사와는 달리 실제로 프로그래밍을 하면서 코딩에 대한 흥미가 생긴 것 같아요. 매일 과제가 있어서 그런지 다른 수업에 비해서 공부하는 비중이 높아지긴 했어요. 여름방학 때는 자료구조 인강을 찾아 들어볼까 생각 중이에요.

학생 5 : 문제가 저한테는 너무 어려워서 과제를 하는 동안 너무 스트레스를 받았어요. 조금 쉬웠으면 더 좋았을 것 같아요.

## 6.2.2 몰입과 경쟁

대부분의 면담 참여 학습자들은 과제를 하는 시간만큼은 과제에만 집중을 할 수 있었다고 말하였다. 순위의 실명제로 인해 스트레스를 받는 학생도 있었다. 반면 순위가 공개됨에 따라 자연스럽게 선의의 경쟁의식이 생긴 학습자도 있었다.

학생 5 : 즉각적으로 맞고 틀린 것을 알 수 있어서 좋았어요.

학생 1 : 과제 할 때 만큼은 집중하고 적절한 알고리즘을 찾아내는 것을 공부하면서 도움이 된 것 같아요. 근데 순위가 이름이랑 함께 공개되니까 그게 좀 스트레스였어요.

학생 3 : 아무래도 순위가 뜨니까 그 순위 때문에 다른 학생들보다 빨리 제출하고 싶은 생각이 들어서 빨리 풀고 싶었어요. 다른 학생들이 풀지 못한 문제를 먼저 풀면 뿌듯했어요.

## 6.2.3 교육자의 자세

면담에 참여한 모든 학습자가 향후 정보교사로 현직에 나아가 자동평가 시스템을 활용하고자 하는 의지가 강했다. 그리고 이 계기를 통해 프로그래밍(알고리즘)에 대해 관심을 가지게 되었고 전공에 중요성을 알고 흥미를 가지게 되었다.

학생 4 : 전공을 제대로 공부해봐야겠다고 생각이 들었어요. 그리고 이런 채점시스템 사이트를 활용해서 프로그래밍 공부를 잘 해봐야겠어요.

학생 1 : 이 수업을 통해서 알고리즘과 코딩이 재미있어졌어요. 그리고 소프트웨어 교육이 중요하다는 것을 알게 되었어요.

학생 2 : 공부를 하면 할수록 소프트웨어교육의 필요성이 더욱 중요하게 느껴졌으나 어떻게 가르쳐야 학생들을 잘 독려하고 이해시킬 수 있을까 하는 고민이 생겼어요.

# 7. 결론 및 제언

## 7.1 결론

본 연구는 자료구조 강의를 수강하는 학생들에게 학습 동기를 부여하고 몰입도를 증가시킬 수

있는 방안의 하나로서, 프로그래밍(알고리즘) 자동 평가 시스템을 적용하기 위해 자료구조의 주요 토픽별 문제 은행을 개발하였다. 실제 수업에 적용하여 코딩흥미와 몰입도 등을 분석함으로써 교원양성 대학에서 프로그래밍, 자료구조 수업에서 활용할 수 있는 ‘학습 동기를 부여하며’, ‘몰입할 수 있는’ 교수-학습 자료 콘텐츠의 제공이 유의미한 변화가 있는지 연구과제로 설정하고 실험, 분석하였다. 그 결과 유의수준 .05에서 통계적으로 유의미한 차이를 보였고, [그림 10]과 학습자 면담 내용을 보면 자동 평가 시스템을 적용한 자료구조 수업에서 학습자들의 소프트웨어교육태도에 변화가 있었음을 알 수 있다.

## 7.2 제언

본 연구를 통해 자료구조(알고리즘) 자동평가 시스템과 관련하여 몇 가지 제언을 하고자 한다.

첫째, 프로그래밍 경시 대회의 준비가 목적이 아닌, 학습 동기를 향상시키고 몰입하고 선의의 경쟁을 하여 성취감 또한 향상시킬 수 있는 교수·학습적인 용도의 시스템들이 개발되어 초·중·고 단위학교에 확산되어야 한다.

둘째, 다수가 동시 접속하여 채점을 수행하더라도 시간 지연이 발생하면 안 된다.

셋째, 학습자가 주어진 문제에 대한 채점 데이터의 입력 특성을 이해하고 있어야 한다. 알고 있어야만 자신의 프로그램(알고리즘)이 잘못된 원인을 스스로 찾아낼 수 있으며, 이러한 과정에서 학습자들의 학습 동기가 촉진된다.

넷째, 자신의 순위는 학습자들에게 프로그래밍에 강한 몰입과 경쟁을 부여하므로 학습자에게 ‘자신의 현재 랭킹’과 ‘상위 랭킹으로의 실현 가능성’을 확인할 수 있도록 다양한 랭킹 시스템을 도입해야 한다.

## 7.3 기대효과

기본적으로 초중고 SW교육에서 교사가 프로그래밍 교수 능력을 갖추는 것은 매우 중요하다. 내용을 깊이 있게 알고 있는 교사는 학습자의 수준에 맞추어 수업을 이끌어 갈 수 있다. 교사가 자



료구조와 알고리즘의 내용을 모르면 학생들은 어렵게 배우게 된다. 따라서 교원양성대학에서 자료구조 수업을 프로그래밍(알고리즘) 자동채점 시스템을 활용하여 교육할 수 있도록 문제해결 은행을 만들어 주는 것은 매우 중요하다.

프로그래밍(알고리즘) 자동 평가 시스템은 사용자가 제출한 소스코드의 정확성과 알고리즘의 시간/공간 효율성 등을 자동으로 평가하고, 주어진 문제를 해결하기 위해 사용자가 제출한 소스 코드에 대해 즉각적인 평가 결과와 교정적 피드백을 제공한다. 또한, 이러한 실시간 평가 결과를 통해 만들어지는 문제별 채점 현황(제출 횟수, 통과 횟수), 랭킹 등의 경쟁적인 요소는 사용자에게 프로그래밍 학습에 대한 흥미와 동기를 제공하는 장점이 있다. 최근 고등학교의 현장 교사들이 오픈 소스로 공개된 Hustoj를 적용하거나, 직접 시스템을 개발하여 학교 현장에서 운영, 활용하고 있으며, 학생들에게 좋은 반응을 얻고 있다. 본 연구는 Hustoj를 기반으로 한 프로그래밍(알고리즘) 자동 평가 시스템을 자료구조 수업에 적용할 수 있는 길을 개척하게 될 것이다.

## 참 고 문 헌

- [1] 교육부(2015). 2015 개정 교육과정 총론. 교육부 고시 제2015-74호 (별책 1)
- [2] 교육부(2015). 실과(기술·가정)/정보과 교육과정. 교육부 고시 제2015-74호 (별책 10)
- [3] 교육부(2015). 과학 계열 전문교과 교육과정. 교육부 고시 제2015-74호 (별책 20)
- [4] 오경선, 안성진 (2015). 프로그래밍이 어려운 이유와 컴퓨팅사고력간의 관계성 연구. 한국컴퓨터교육학회 논문지, 18(5), 55-62.
- [5] 송지희(2011). 자기주도학습을 위한 자동채점기반의 프로그래밍 교육 시스템. 박사학위논문, 송실대학교.
- [6] 정종광(2010). 과학고 학생을 위한 Online Judge 기반 프로그래밍 평가 시스템의 설계 및 구현. 석사학위 논문, 한국교원대학교
- [7] Yingwei L., Xiaolin W., &Zhengyi Z.(2008). Programming grid : a computer-aided education system for programming courses based on online judge. ACM SCE '08 Proceedings, 6.
- [8] 임형석(2007). ACM-ICPC 문제의 출제 및 채점 과정. 한국정보과학지, 25(7).52-55
- [9] GitHub.(2017). zhblue/hustoj. [online] Available at: <https://github.com/zhblue/hustoj> [Accessed 3 Feb. 2017].
- [10] 장원영, 김성식(2017). 정보 교육에서 프로그래밍(알고리즘) 자동 평가 시스템의 활용 가능성에 대한 고찰. 컴퓨터교육학회 논문지, 20(1), 13-26
- [11] 전용주(2017). 새로운 교육과정의 소프트웨어 교육을 위한 컴퓨팅 사고력 기반 창의적 문제해결(CT-CPS) 수업모형의 개발 및 적용. 박사학위 논문, 한국교원대학교.
- [12] 정종광(2017). KoiStudy. [online] Available at: <http://koystudy.net> [Accessed 3 Feb. 2017].
- [13] 장원영(2017). Coding is fun. [online] Available at: <http://codingfun.net> [Accessed 3 Feb. 2017].
- [14] 장원영, 김성식(2014). 알고리즘 자동평가 시스템의 개발 및 적용 : 프로그래밍 학습 효과 분석. 한국컴퓨터교육학회 논문지, 17(4), 26-34.
- [15] 정은미(2009). 프로그래밍 언어에 대한 주관식 자동 채점 시스템의 설계 및 구현. 석사학위 논문, 안동대학교
- [16] 전현석, 정종광, 김성식(2014). C언어 기초 학습을 위한 문제 설계 및 운영. 한국컴퓨터교육학회 학술발표대회 논문집, 18(1), 291-294.



## 김 성 식

1994 고려대학교  
경영학과(경영학사)  
1977~1991 교육부 및  
교육정책자문위원회 근무  
(행정고시 19회)

1988 미국오리곤주립대학교 대학원  
컴퓨터과학과(이학석사)  
1992 고려대학교 컴퓨터학과(이학박사)  
1992~현재 한국교원대학교 컴퓨터교육과 교수  
관심분야: 교육용콘텐츠, 알고리즘, 원격교육  
E-Mail: seongkim@knue.ac.kr



## 오 소 희

2015 한밭대학교  
컴퓨터공학과(공학사)  
2017 한국교원대학교  
컴퓨터교육과(교육학석사)

2018~현재 한국교원대학교 컴퓨터교육과  
박사과정  
관심분야: 교육용콘텐츠, SW교육, 알고리즘  
E-Mail: sohee00815@gmail.com



## 정 상 수

2009 경상대학교  
컴퓨터교육과(이학사)  
2015~현재 한국교원대학교  
컴퓨터교육과 석사과정

2016~현재 세종특별자치시 세종고등학교 교사  
관심분야: 알고리즘, 자료구조, 프로그래밍 교육  
E-Mail: tosang@naver.com