

탐색 알고리즘 교육을 위한 S/W 컴포넌트의 개발

정 인 기*

*춘천교육대학교 컴퓨터교육과

요 약

자료구조 및 알고리즘 분야는 컴퓨터 프로그래밍 교육의 기반이 되는 과목이다. 그러나, 이제는 이 과목에서 활용되는 프로그래밍 방법이 비주얼 프로그래밍과 윈도우 프로그래밍에 적당하지 못하다고 해서 학생들로부터 외면당하고 있다. 따라서, 본 논문에서는 효과적으로 탐색 알고리즘을 교육할 수 있고, 비주얼 프로그래밍에 기반을 둔 소프트웨어 컴포넌트인 SCSA (Software Component for Search Algorithm)를 개발하였다.

Development of S/W Component for Search Algorithm Education

Inkee Jeong*

*Computer Education Department, ChunCheon National University of Education

ABSTRACT

Data structure and algorithm are primary courses for computer programming education. However, now those are not interesting courses for students because the programming methodologies that used these courses are not the proper methodologies for the visual and windows programming. Therefore, we developed the SCSA (Software Component for Search Algorithm) which is a software component for effective search algorithm education, and is based on visual programming in this thesis.

1. 서론

컴퓨터 프로그래밍은 컴퓨터 교육의 기반이 되는 분야이다. 특히, 자료 구조와 알고리즘 과목은 컴퓨터 프로그래밍 교육의 가장 핵심이 되는 과목이라고 할 수 있다. 그러나, 학생들에게는 다른 과목에 비하여 과목에 대한 흥미와 이해가 점점 떨어지고 있는 실정이다. 프로그래밍 패러다임이 바뀌고, 사회가 비주얼 시대로 변화하면서 이러한 현상은 더욱 심화되

어 가고 있다.

사실 기존의 컴퓨터 프로그램처럼 문자로 되어있는 경우에는 이해하기 어려울 때가 있으며, 또한 문자 형태의 표현은 프로그램의 기능을 모호하게 전달할 수 있다. 그에 비하여 프로그램의 애니메이션 형태의 표현은 그들의 의미, 방법 및 목적을 좀 더 잘 전달해 줄 수 있다. 애니메이션 형태는 프로그램의 이해, 기존 프로그램의 평가 및 새로운 프로그램의 개발과 같은 활동에 도움을 줄 수 있다[12].

뿐만 아니라 동적이고 추상적인 프로세스를 포함하는 컴퓨터 프로그램들은 그들의 행위를 그래픽 시뮬레이션을 통해 관찰하는 것이 가장 이해하기 쉬울 수 있다. 실행 시의 프로그램의 그래픽 표현은 이해하거나 주의하기에 어려운 몇몇 속성을 드러내게 해준다. 또한, 이러한 전달 매체는 사람과 시뮬레이션되는 프로그램이 경로 정보의 제어에 의한 애니메이션으로 상호작용하는 경우에는 더욱 큰 힘을 발휘하게 된다. 알고리즘의 애니메이션 환경은 컴퓨터 프로그램에 대해서 이러한 패러다임에 따르는 것이다 [14].

이에 따라 선진국에서는 자료 구조를 효과적으로 가르치기 위한 많은 노력이 있어왔다. 특히, 자료 구조의 시각화에 대한 많은 연구가 있었고 어느 정도의 성과를 이루었지만, 이들 연구의 대부분도 Pascal이나 C와 같은 프로그래밍 언어가 주를 이루던 시대의 성과이기 때문에 다음과 같은 문제점을 가지고 있다.

첫째, 비주얼 프로그래밍에 맞게 개발된 것이 아니다. 기존의 자료 구조 교재에 있는 프로그램들을 Visual C++ 등에서 사용할 수 없는 것은 아니지만 텍스트 기반 프로그램에서의 입출력 코드와 윈도우즈 기반 프로그램에서의 입출력 코드는 많이 다르기 때문에 자료 구조를 배우는 초보 프로그래머들은 입출력 코드 작성에 많은 시간과 노력이 투입하여야 한다. 그러나, 대부분의 자료 구조 과목 시간이 이러한 모든 것을 교육할 수 있을 만큼 여유 있지 않기 때문에 좀 더 효율적인 방법을 찾는 것이 필요하다.

둘째, 대부분의 연구 성과가 자료 구조의 개념을 이해시키기 위하여 동작을 단계적으로 시각화해서 보여주는 데 지나지 않고 있다. 따라서, 학생들이 자신이 작성한 프로그램을 시각화하여 볼 수 있는 것과는 거리가 있는 연구 성과들이다. 물론, 자료 구조의 개념을 애니메이션 등을 통하여 보는 것도 상당한 효과가 있으나 이보다 더 효과적인 것은 자신의 프로그램을 볼 수 있도록 하는 것이다.

자료 구조 과목에서 교육되어지는 분야에는 여러 가지가 있으나 일반적으로 큐와 스택, 트리, 그래프, 정렬, 탐색 등의 분야를 대표적으로 언급할 수 있다. 이 중에서 탐색 알고리즘은 자료 구조 및 알고리즘

과목에서 가장 기본이 되는 알고리즘이다. 탐색 알고리즘은 실제 프로그래밍에서 대단히 많이 쓰이는 알고리즘이지만 기본 개념은 매우 간단하고 직관적이기 때문에 교육에 있어서 매우 수월한 알고리즘으로 알려져 있다. 따라서, 교육을 위한 별도의 방법론을 제시한 경우는 많지 않다.

그러나, 순차 탐색인 경우에는 개념이 매우 간단하여 학생들이 알고리즘을 이해하는데 큰 어려움이 없지만 다른 탐색 알고리즘인 경우에는 학생들이 탐색 알고리즘의 개념은 이해하면서도 이를 올바르게 프로그래밍하는데 어려움을 느끼는 경우가 있다.

이러한 현상은 자신이 작성한 프로그램을 시각적으로 볼 수 없는 데서 생기는 현상이며, 학생들은 시각적으로 자신의 프로그램이 수행되는 것을 볼 수 없기 때문에 자신이 작성한 프로그램에 대해서 확신을 가지지 못하고, 오류에 대하여 정확한 디버깅을 할 수 없게 된다.

특히, 순차 탐색을 제외한 다른 종류의 탐색 알고리즘을 프로그래밍하는 경우에 첨자 연산과 같은 수식을 연산하는 경우가 발생하는데 학생들의 정확한 연산을 위한 프로그래밍 작성 능력 부족으로 인하여 탐색을 제대로 못하는 경우가 발생하게 된다. 이러한 경우에 오류가 발생한 시점에서 프로그램의 수행 상태를 알게 되면 쉽게 디버깅할 수 있을 것이다. 물론, 프로그래밍 개발 시스템에서 제공하는 디버깅 도구를 이용하면 현재의 상태를 간단하게 알 수 있지만, 프로그램을 처음 공부하는 학생들은 이러한 도구를 사용하는데 익숙치 않다. 따라서, 학생들에게 가장 좋은 방법은 탐색이 수행되는 상태를 시각적으로 보여줄 수 있는 도구를 제공하는 것이며, 본 논문에서는 이를 위하여 비주얼 프로그래밍 도구에서 자료 구조에 대한 지식을 효과적으로 습득할 수 있는 소프트웨어 컴포넌트를 개발하였다.

2. 기존 자료 구조 교육용 시스템 분석

지금까지 자료 구조 및 알고리즘 교육을 위하여 많은 시스템들이 개발되었다. 이들의 특성을 분석해보면 다음과 같이 분류해 볼 수 있다.

① 자료 구조 및 알고리즘 연산 수행의 통계적 처리 형태

알고리즘을 수행하고 난 후 특정 연산에 대한 실행 결과를 통계적으로 처리하여 보여주며 학생들은 이것을 분석해 봄으로써 알고리즘의 성능 등을 비교해 볼 수 있다. 이러한 형태는 새로운 알고리즘을 개발하는 경우에는 좋은 도구로서 이용될 수 있으나 이미 개발된 알고리즘을 정확히 프로그래밍하는 것이 중요한 초보 프로그래머 및 학생들에게는 많은 도움을 주지 못하는 경우가 발생한다. 일반적으로 초기의 시스템에서 보여지며, Hebert L. Dershem 등[11]이 만든 시스템에서 일부 사용되어 지고 있다.

② 자료 구조 개념 이해를 위한 그래픽/애니메이션 형태

자료 구조 및 알고리즘의 수행 상태를 그래픽 혹은 애니메이션의 형태로 보여줌으로써 학생들이 개념을 이해하는데 도움을 주는 형태이다. 이와 같은 애니메이션 형태는 개념의 이해에는 도움을 주지만 학생들이 실제 프로그램을 작성하는 데는 큰 도움을 주지 못하게 되는 경우가 많다. 이러한 형태로는 Bruce R. Maxim 등[3], G. Michael Barnes 등[8], G. Scott Dwen[9], Moshe Augenstein 등[15]이 만든 시스템이 있으며, Hebert L. Dershem 등[11]이 만든 시스템은 그래픽/애니메이션 형태와 함께 통계적 처리 방법도 사용하고 있다. 이 중에서 G. Michael Barnes 등[8]이 만든 시스템과 G. Scott[9]이 만든 시스템은 이진 트리를 위한 시스템이다. 또한, Hebert L. Dershem 등[11]이 만든 시스템은 정렬 알고리즘을 위한 것이고, Bruce R. Maxim 등[3]이 만든 시스템과 Moshe Augenstein 등[15]이 만든 시스템은 스택, 정렬, 연결 리스트 등 다양한 자료 구조 및 알고리즘에 적용할 수 있는 시스템들이다.

③ 자료 구조 및 알고리즘 연산의 개념 이해를 위한 고유의 연산자를 가진 시스템 형태

자료 구조 및 알고리즘 교육을 위한 별도의 시스템을 만들어 그 안에서 고유의 연산을 수행하도록 하는 형태인데, 이들은 자료 구조 및 알고리즘의 기본 연산에 대하여 고유의 구문 법칙을 가지고 이를

사용자가 입력 혹은 선택함으로써 연산의 수행 상태를 학생들에게 보여주도록 하고 있다. 이러한 방식은 각 연산에 대한 정확한 흐름을 학생들에게 보여주어 전체적인 개념뿐만 아니라 각 연산에 대한 정확한 이해를 할 수 있게 해 주지만 학생들에게 또 다른 형태의 시스템에 대한 사용법을 익히도록 요구하게 되며, 범용 프로그래밍 언어로 구현하는 경우에는 학생들이 이를 응용 및 전환해야 하는 어려움이 생긴다. 이러한 형태의 시스템들로는 SWAN[4], VISAL[6], Tango[12], BALSА-II[14], webGAIGS[19], JAWAA[21] 등의 시스템과 Gerar K. Rambally[7]이 만든 시스템이 있다. 이들은 대부분 정렬, 트리, 연결 리스트 등 다양한 자료 구조와 알고리즘에 적용할 수 있도록 만들어졌다.

④ 라이브러리/전처리기 형태

라이브러리 및 전처리기 형태는 학생들이 범용 프로그래밍 언어를 이용하여 자료 구조 및 알고리즘 과목에서 배운 프로그램을 구현하고자 하는 경우에 필요한 입출력 모듈이나 수행 상태를 표현하고자 할 때 전문 개발자에 의해 만들어진 라이브러리나 전처리기 모듈 형태를 이용하게 하여 학생들에게 프로그래밍에 대한 부담을 줄여 효과적으로 자료 구조 및 알고리즘 내용을 습득할 수 있게 해 준다. 이러한 형태는 범용 프로그래밍 언어를 사용하게 함으로써 별도로 시스템을 배울 필요가 없는 장점을 가지고 있지만, 학생들이 라이브러리의 사용법을 미리 인지하고 있어야 하며, 프로그램의 형태가 주어진 방식대로가 아니라면 원하는 결과를 나타내지 않을 가능성을 포함하고 있다. 이러한 형태의 시스템들로는 BALSА-II[14], GAIGS 3.0[18] 등의 시스템들과 Moshe Augenstein 등[16]의 연구가 있다. 이들 역시 다양한 자료 구조 및 알고리즘에 적용할 수 있는 시스템들이다.

⑤ 소프트웨어 컴포넌트 형태

소프트웨어 컴포넌트는 최근에 비주얼 프로그래밍과 함께 새롭게 나타나고 있는 프로그래밍의 형태이다. 소프트웨어 컴포넌트는 컴포넌트 자체가 하나의 독립된 모듈을 이루고 있고, 비주얼 프로그래밍 개발

탐색 알고리즘 교육을 위한 S/W 컴포넌트의 개발

시스템에서는 만들어진 컴포넌트의 인터페이스에 대하여 프로그램 작성 시에 안내를 해 줌으로써 학생들이 사용하는데 불편함이 없도록 할 수 있다. 또한, 범용 프로그래밍 언어를 사용함으로써 별도의 교육 없이 이용할 수 있는 장점을 가진다. 이러한 개념의 시스템으로 개발된 예는 아직 많지 않은 실정이다.

지금까지 개발된 자료 구조 교육용 주요 시스템들의 내용을 정리하면 <표 1>과 같다.

<표 2> 자료 구조 및 알고리즘 교육용 프로그램

| 연구자 | 시스템 | 대상 자료구조 및 알고리즘 | 시스템 형태 |
|--|-----------|--|----------------------------|
| Bruce R. Maxim, Bruce S. Elenbogen [3] | | Stack, Sorting, Linked List | 그래픽 혹은 애니메이션 형태 |
| Clifford A. Shaffer, Lenwood S. Health, Jun Yang [4] | SWAN | Sorting, Tree, Graph | 독립 시스템 형태 |
| Elena I. Giannotti [6] | MSAL | Linked List, Tree | 독립 시스템 형태 |
| Gerar K. Rambally [7] | | Linked List | 독립 시스템 형태 |
| G. Michael Barnes, Gary A. Kind [8] | CABTO | Binary Tree | 그래픽 혹은 애니메이션 형태 |
| G. Scott Dwen [9] | | Binary Tree | 그래픽 혹은 애니메이션 형태 |
| Herbert L. Dershem, Peter Brummund [11] | | Sorting | 통계적 처리 형태, 그래픽 혹은 애니메이션 형태 |
| John T. Stasko [12] | Tango | sorting, searching, hashing, computational geometry, string, graph, tree, etc. | 독립 시스템 형태 |
| Marc H. Brown [14] | BALSA-II | Array, Stack, Queue, Linked List, Binary Tree, General Tree, Graph, etc. | 독립 시스템 형태, 라이브러리/전처리 형태 |
| Moshe Augenstein, Yedidyah Langsam [15] | | Array, Stack, Queue, Binary Tree, Linked List | 그래픽 혹은 애니메이션 형태 |
| Moshe Augenstein, Yedidyah Langsam [16] | | Array, Stack, Queue, Tree, Linked List | 라이브러리/전처리 형태 |
| Thomas L. Naps, Brian Swander [18] | GAIGS 3.0 | Array, Stack, Queue, Linked List, Binary Tree, General Tree, Graph | 라이브러리/전처리 형태 |
| Thomas L. Naps, Eric Bressler [19] | webGAIGS | Array, Stack, Queue, Linked List, Binary Tree, General Tree, Graph | 독립 시스템 형태 |
| Willard C. Pierson, Susan H. Rodger [21] | JAWAA | Array, Stack, Queue, Graph | 독립 시스템 형태 |

<표 1>에서와 같이 대부분의 시스템들이 그래픽/애니메이션 형태, 라이브러리/전처리 형태 혹은 독립 시스템 형태를 가지고 있다. 특히, 소프트웨어 컴포넌트 형태는 현재 각광받고 있음에도 불구하고, 아직은 이 분야에서 뚜렷한 연구 결과물이 적다.

또한, 적용되는 자료 구조 및 알고리즘은 배열, 스택, 큐, 트리, 연결 리스트, 그래프 등이 있는데, 탐색 알고리즘을 적용할 수 있는 시스템은 BALSА-II와 같은 것들을 제외하면 거의 없는 것으로 파악되었다.

이는 탐색 알고리즘이 상대적으로 이해하기 쉬운 개념을 가지고 있는 것으로 생각된다.

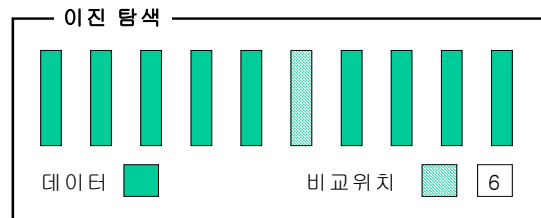
그러나, 선형 탐색을 제외하면 프로그래밍 작업에서 학생들이 느끼는 바는 그렇지 않다. 따라서, 탐색 알고리즘을 교육할 수 있는 시스템의 개발은 반드시 필요하며, 그것이 현재의 사용되고 있는 프로그래밍 기법인 소프트웨어 컴포넌트에 해당하는 것이라면 더욱 소기의 목적을 달성할 수 있을 것이다.

3. 탐색 알고리즘 교육용 소프트웨어 컴포넌트 (SCSA: Software Component for Search Algorithm)

탐색이란 자료 집단 내의 특정 자료를 찾아내는 작업을 말한다. 탐색은 자료 구조의 종류는 아니지만 자료 구조 과목에 필수적으로 나오는 기본 알고리즘이며, 외부 탐색과 내부 탐색으로 크게 나누어지며, 내부 탐색의 대표적인 알고리즘에는 선형 탐색, 이진 탐색, 피보나치 탐색 및 보간 탐색 등이 있다[2]. 본 논문에서는 내부 탐색만을 대상으로 하고자 한다.

탐색은 자료가 배열의 형태로 있을 때를 대상으로 프로그래밍하는 것이 일반적이며, 프로그램의 진행 상태는 배열 상에서 비교되어지는 위치를 보여주는 것으로 표현할 수 있다. 따라서, 본 논문에서는 프로그램의 진행 상태를 보여주기 위하여 자료가 비교되는 위치를 표현하여 탐색 알고리즘을 시각화하고자 하였다.

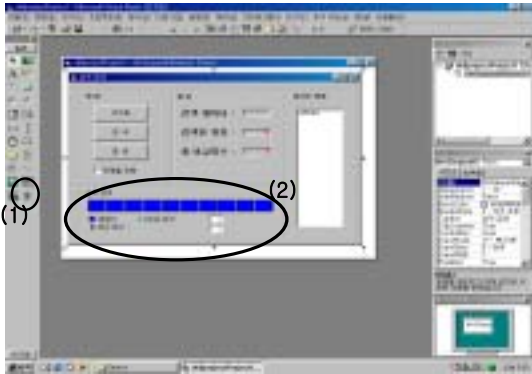
탐색에 있어서 리스트의 상태 변화를 보여주는 가장 주요한 부분은 자료와 검색 키의 비교이기 때문에 본 논문에서는 <그림 1>에서와 같이 자료가 검색 키와 비교되는 경우의 상태를 보여줄 수 있는 형태로 개발하였다.



<그림 1> 탐색 알고리즘 시각화 컴포넌트의 구성

탐색 알고리즘 교육을 위한 S/W 컴포넌트의 개발

본 논문에서는 <그림 1>과 같은 형태의 소프트웨어 컴포넌트로서 SCSA (Software Component for Search Algorithm)를 개발하였으며, Visual Basic 혹은 Visual C++에서 이용할 수 있도록 ActiveX를 이용하여 개발하였다. SCSA를 실제 프로그래밍에서 사용하는 예는 <그림 2>와 같다.



<그림 2> SCSA를 사용한 Visual Basic 프로그래밍 작성 예

<그림 2>와 같이 Visual Basic 프로그래밍 도구에서 프로그래머가 SCSA 컨트롤을 선택하게 되면 컨트롤 도구 모음에 (1)과 같은 아이콘이 나타나며, 이를 일반 컨트롤처럼 프로그래밍하고 있는 폼 위에 끌어다 놓으면 (2)와 같은 컨트롤이 형성된다. 이후에는 다른 비주얼 프로그래밍의 컨트롤과 같이 프로그래밍하면 된다.

비주얼 프로그래밍의 경우에 기본적으로 제공하거나 제3자가 만들어서 제공하는 컨트롤을 사용하면 매우 빠르고 쉽게 프로그램을 작성할 수 있으나, 적당한 컨트롤을 구하지 못하는 경우에는 직접 화면 및 연산을 구성하여야 한다. 이러한 경우에 전체적인 프로그램 작성에 있어서 필요한 객체를 생성하고, 연산을 정의하는데 많은 시간과 노력이 들어가게 된다. 물론, 그 작업이 목표가 되는 경우라면 직접 작성하는 것이 좋을 것이지만 제한된 시간 안에 여러 분야를 교육해야 하는 경우라면 원래의 목적에서 벗어나는 시간과 노력은 줄여주는 것이 교육의 목적을 달성하는데 좀 더 효과적이 될 것이다. 본 논문에서 개발된 SCSA 컨트롤을 사용하면 <표 2> 혹은 <표 3>에서처럼 기존의 탐색 알고리즘에 몇 줄의 프로그

램을 삽입함으로써 소기의 목적을 달성할 수 있다. <표 2>와 <표 3>은 각각 Visual Basic 언어로 작성된 순차 및 이진 탐색 알고리즘에 SCSA 컨트롤 사용을 위한 코드를 삽입한 예이다.

<표 3> SCSA 컨트롤 활용 코드를 삽입한 Visual Basic 프로그램의 예 - 순차 탐색

```
Private Sub cmdSearch_Click()
    Dim i, s, SearchData As Integer
    Dim Prompt, Title As String

    Prompt = "탐색할 자료를 입력하십시오."
    Title = "순차 탐색"
    SearchData = Val(InputBox$(Prompt, Title))
    lblSearchData.Caption = SearchData
    vshSearch.Init
    vshSearch.StepMode = chkStep
    i = 0
    Do While SearchData <> x(i) And i < DataNumber
        i = i + 1
        vshSearch.Compare i
    Loop
    If i < DataNumber Then
        lblSearchPosition.Caption = i
        lstData.LstIndex = i
        lblSearchNumber.Caption = i + 1
    Else
        lblSearchPosition.Caption = "없음"
        lblSearchNumber.Caption = i
    End If
End Sub
```

<표 2>와 <표 3>의 vshSearch 컨트롤은 <그림 2>와 같이 진행되는 프로그래밍 작업에서 부여된 컨트롤 이름이다. 여기에서 vshSearch.Init는 컨트롤을 초기화하는 명령으로 SCSA 컨트롤의 속성 값을 초기화하는 명령이며, vshSearch.StepMode는 알고리즘의 수행을 단계적으로 볼 수 있는 모드로 설정하는 속성으로 이를 true로 설정하면 프로그램은 한 단계씩 진행한 후 사용자의 입력을 기다리며, false로 설정되어 있으면 프로그램을 멈추지 않고 끝까지 수행한다. vshSearch.Compare는 자료를 비교하는 위치를 표시해주는 명령으로 숫자를 대입하면 그에 해당하는 위치를 표현해 준다. 이와 같이 자료 구조의 시각화를 위하여 프로그램 내에 삽입되는 코드를 최소화함으로써 기존 교재에 있는 알고리즘들을 거의 변경 없이 사용할 수 있다.

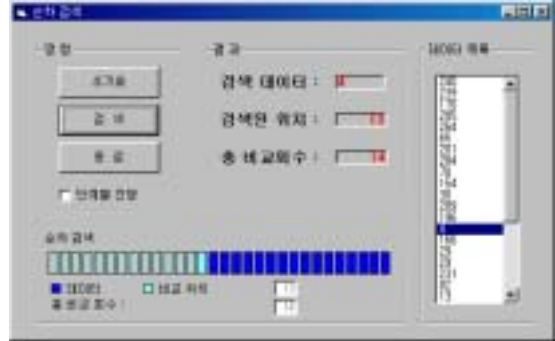
탐색 알고리즘 교육을 위한 S/W 컴포넌트의 개발

SCSA 컨트롤을 사용하여 순차 탐색 프로그램과 이진 탐색 프로그램을 작성하여 실행시켰을 때의 화면은 각각 <그림 3> 및 <그림 4>와 같다.

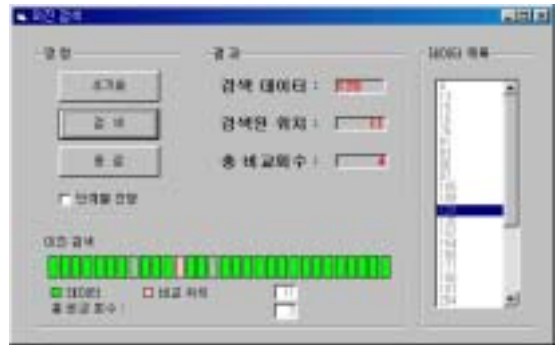
<표 4> SCSA 컨트롤 활용 코드를 삽입한 Visual Basic 프로그램의 예 - 이진 탐색

```
Private Sub cmdSearch_Click()
    Dim i, Low, Middle, High As Integer
    Dim SearchData As Integer
    Dim Prompt, Title As String

    Prompt = "탐색할 자료를 입력하십시오."
    Title = "이진 탐색"
    SearchData = Val(InputBox$(Prompt, Title))
    lblSearchData.Caption = SearchData
    vshSearch.Init
    vshSearch.StepMode = chkStep
    i = 0
    Low = 0
    High = DataNumber - 1
    Do While Low < High
        Middle = (Low + High) / 2
        i = i + 1
        vshSearch.Compare Middle
        if SearchData = x(Middle) Then
            Exit Do
        ElseIf SearchData > x(Middle) Then
            Low = Middle + 1
        Else
            High = Middle - 1
        End If
    Loop
    If SearchData = x(Middle) Then
        lblSearchPosition.Caption = Middle
        lstData.ListIndex = Middle
        lblSearchNumber.Caption = i + 1
    Else
        lblSearchPosition.Caption = "없음"
        lblSearchNumber.Caption = i
        lstData.ListIndex = -1
    End If
End Sub
```



<그림 3> 순차 탐색 프로그램 실행 예



<그림 4> 이진 탐색 프로그램 실행 예

본 논문에서 개발된 SCSA 컨트롤의 주요 속성은 <표 4>와 같다.

<표 5> SCSA 컨트롤의 주요 속성

| 속성명 | 속성값 | 의미 |
|-----------|---------------|--------------------------------|
| BarColor | 정수 | 전체 탐색 대상이 되는 bar의 색깔 |
| BarNumber | 정수 | 전체 탐색 대상 데이터의 수 |
| CompColor | 정수 | 키와 비교하고 있는 bar의 색깔 |
| StepMode | true 혹은 false | false - 전체 실행 true - 단계적 실행 |

또한, SCSA 컨트롤에서 사용할 수 있는 주요 메소드는 <표 5>와 같다.

탐색 알고리즘 교육을 위한 S/W 컴포넌트의 개발

<표 6> SCSA 컨트롤의 주요 메소드

| 메소드명 | 인수값 | 의미 |
|---------|-----|--------------------|
| Compare | 정수 | 키와 비교되고 있는 데이터의 위치 |
| Init | 없음 | SCSA 컨트롤 속성값의 초기화 |

4. SCSA의 활용 및 평가

SCSA를 활용하여 수업을 한 후 학생들에게 설문지를 통하여 효과를 조사한 결과는 <표 6>¹⁾과 같다.

<표 7> SCSA 활용 효과 분석 결과

| 번호 | 항 목 | 효과 적임 | 효과 없음 | 혼란 가중 | 무 응답 | 합계 |
|----|----------------------------|-------|-------|-------|------|--------|
| ① | 탐색 알고리즘 개념의 이해에 도움이 되었다. | 19 | 14 | 0 | 1 | 34 |
| | | (56%) | (41%) | (0%) | (3%) | (100%) |
| ② | 탐색 알고리즘 구현 시 디버깅에 도움이 되었다. | 32 | 2 | 0 | 0 | 34 |
| | | (94%) | (6%) | (0%) | (0%) | (100%) |
| ③ | 탐색 알고리즘 비교 작업에 도움이 되었다. | 26 | 4 | 1 | 3 | 34 |
| | | (76%) | (12%) | (3%) | (9%) | (100%) |
| ④ | 탐색 알고리즘 비교 횟수 분석에 도움이 되었다. | 24 | 6 | 2 | 2 | 34 |
| | | (70%) | (18%) | (6%) | (6%) | (100%) |

<표 6>과 같이 알고리즘 교육에 활용하는 경우에 좋은 결과를 얻을 수 있었는데 특히, 알고리즘을 프로그램으로 구현하는 과정에서 발생하는 오류 등을 발견하는데 좋은 효과가 있는 것으로 나타났다. 그러나, 항목 ①에서는 큰 효과를 보지 못한 것으로 나타났는데, 이는 탐색 알고리즘 이해에 대한 교육을 SCSA 컨트롤을 활용하기 전에 실시하였으며, 탐색 알고리즘이 일반 강의로도 이해하기 쉬웠던 분야이기 때문인 것으로 분석되었다. 항목 ③과 ④에서의 긍정적이지 못한 소수의 응답에 대해서는 알고리즘의 비교 작업과 비교 횟수 계산 자체를 거의 이해하지 못하는 학생들이 효과를 보지 못한 것으로 분석되었다. 따라서, SCSA 컨트롤은 알고리즘 자체를 이해하는 단계보다는 프로그램을 구현하는 단계에서 훨씬 효과적인 것으로 파악할 수 있다.

SCSA 컨트롤의 주 사용자는 자료 구조를 공부하

1) 2001년에 본 교의 컴퓨터교육과 3학년 자료 구조 시간에 활용 후 조사한 결과임

는 학습자들이며, 자료 구조를 가르치는 교수자들 역시 학습자들에 대한 교육 방법의 개발과 과제 부여 등에 이용할 수 있는데 SCSA 컨트롤을 이용하면 탐색 알고리즘을 교육할 때 다음과 같은 장점이 있다.

첫째, 탐색 알고리즘의 시각화를 통하여 탐색 알고리즘의 개념에 대한 학습자의 직관적 이해를 증진시키고, 프로그램의 디버깅에 도움을 준다.

둘째, 탐색 알고리즘 상태를 변화시키는 연산 후의 모습을 학습자로 하여금 볼 수 있게 함으로써 자신의 프로그램에 대한 수행 결과를 확인할 수 있다.

셋째, 일반적인 프로그램 라이브러리가 아닌 소프트웨어 컴포넌트로 구성함으로써 사용자들이 특별한 조치 없이 컨트롤들을 활용할 수 있다.

넷째, 비주얼 프로그래밍 도구에 맞는 소프트웨어 컴포넌트인 ActiveX 컨트롤로 구성함으로써 비주얼 프로그래밍에 익숙한 학습자들은 새로운 언어나 프로그래밍 도구 체계를 새로 습득할 필요 없이 탐색 알고리즘을 프로그래밍할 수 있다.

다섯째, 소프트웨어 컴포넌트로 제공하면서 최소한의 명령으로 자료 구조의 시각화를 구현하여, 교수자들과 학습자들이 현재의 자료 구조 교체에 나타나 있는 탐색 알고리즘들을 거의 변경 없이 사용할 수 있도록 해 준다.

여섯째, 소프트웨어 컴포넌트로 제공하기 때문에 탐색 알고리즘의 시각화를 위한 별도 프로그램 작성이 필요 없으므로 학습자들은 탐색 알고리즘에만 신경 쓰면서 공부할 수 있다.

일곱째, 탐색 알고리즘 시각화를 위한 프로그램의 작성에 많은 시간을 낭비하지 않아도 되므로 교수자들은 주어진 교과 시간에 좀 더 많은 알고리즘을 효과적으로 강의할 수 있다.

5. 결론

자료 구조 및 알고리즘 과목은 컴퓨터 관련 학과의 교육과정 중에서 가장 기본적인 과목으로 인식되어 왔고, 학생들의 프로그램에 대한 개념 이해 및 방법에 대한 교육 분야에서 많은 역할을 해왔다. 그러나, 프로그래밍에 대한 패러다임이 변화하는 현 시점에서 자료 구조 및 알고리즘 과목에 대한 교육 방법

도 변화하여야 한다. 특히 비주얼 프로그래밍으로 대변되는 큰 흐름 속에서 자료 구조의 과목도 이러한 패러다임에 맞게 변화할 필요가 있다.

따라서, 본 논문에서는 비주얼 프로그래밍을 활용하면서 자료구조 및 알고리즘 과목을 교육할 수 있는 방법으로 소프트웨어 컴포넌트를 이용하는 방법을 제안하였다. 또한, 자료 구조 및 알고리즘의 여러 분야 중에서 탐색 알고리즘 교육을 위한 소프트웨어 컴포넌트로써 SCSA를 개발하였고, 이를 탐색 알고리즘 교육에 적용하여 좋은 효과를 얻을 수 있었다.

앞으로 자료 구조 및 알고리즘의 다른 분야에서의 교육용 소프트웨어 컴포넌트들도 개발하여 다양한 분야의 자료 구조들을 학생들이 효과적으로 익힐 수 있는 환경을 만들 계획이다.

참고문헌

- [1] 이이표, 김병세, Microsoft Visual Basic Bible 6.0, 삼양출판사, 2001.
- [2] 황종선, 손진곤, 알고리즘, 정익사, 1994.
- [3] Bruce R. Maxim and Bruce S. Elenbogen, "Teaching Programming Algorithm Aided by Computer Graphics", SIGCSE '87 Papers of the 18th SIGCSE technical symposium on Computer Science Education, pp. 297-301, ACM SIGCSE, 1987.
- [4] Clifford A. Shaffer and Lenwood S. Heath, Jun Yang, "Using the SWAN Data Structure Visualization System for Computer Science Education", SIGCSE '96 Proceedings of the twenty-seventh SIGCSE technical symposium on Computer Science Education, pp. 140-144, ACM SIGCSE, 1996.
- [5] David Scanlan, "Data-Structures Students May Prefer to Learn Algorithms Using Graphical Methods", SIGCSE '87 Papers of the 18th SIGCSE technical symposium on Computer Science Education, pp. 302-307, ACM SIGCSE, 1987.
- [6] Elena I. Giannotti, "Algorithm Animator : A Tool for Programming Learning", SIGCSE '87 Papers of the 18th SIGCSE technical symposium on Computer Science Education, pp. 308-314, ACM SIGCSE, 1987.
- [7] Gerard K. Rambally, "Real-Time Graphical Representations of Linked Data Structures", CSE '85 Proceedings of the sixteenth SIGCSE technical symposium on Computer Science Education, pp. 41-48, ACM SIGCSE, 1985.
- [8] G. Michael Barnes, Gary A. kind, "Visual Simulations of Data Structures During Lecture", SIGCSE '87 Papers of the 18th SIGCSE technical symposium on Computer Science Education, pp. 267-276, ACM SIGCSE, 1987.
- [9] G. Scott Dwen, "Teaching of Tree Data Structures using Microcomputer Graphics", SIGCSE '86 Papers of the seventeenth SIGCSE technical symposium on Computer Science Education, pp. 67-72, ACM SIGCSE, 1986.
- [10] Guido Rößling, Markus Schüler and Bernd Freisleben, "The ANIMAL Algorithm Animation Tool", SIGCSE '00 Proceedings of the thirty-first SIGCSE technical symposium on Computer Science Education, pp. 37-40, ACM SIGCSE, 2000.
- [11] Herbert L. Dershem, Peter Brummund, "Tools for Web-Based Sorting Animation", SIGCSE '98 Proceedings of the twenty-ninth SIGCSE technical symposium on Computer Science Education, pp. 222-226, ACM SIGCSE, 1998.
- [12] John T. Stasko, "Tango: A Framework and System for Algorithm Animation", IEEE Computer, Vol. 23, No. 9, pp. 39-44, 1990.
- [13] Judith Wilson and Robert Aiken, "Review of Animation Systems for Algorithm Understanding", SIGCSE '96 Proceedings of the twenty-seventh SIGCSE technical symposium on Computer Science Education, pp. 75-77, ACM SIGCSE, 1996.

- [14] Marc H. Brown, "Exploring Algorithms Using Balsa-II", IEEE Computer, Vol. 21, No. 5, pp. 14-36, 1988.
- [15] Moshe Augenstein and Yedidyah Langsam, "Graphic Displays of Data Structures on the IBM PC", SIGCSE '86 Papers of the seventeenth SIGCSE technical symposium on Computer Science Education, pp. 73-81, ACM SIGCSE, 1986.
- [16] Moshe Augenstein and Yedidyah Langsam, "Automatic Generation of Graphic Displays of Data Structures Through a Preprocessor", SIGCSE '88 Proceedings of the nineteenth SIGCSE technical symposium on Computer Science Education, pp. 148-152, ACM SIGCSE, 1988.
- [17] Robert L. Kruse, Data Structures and Program Design, Prentice-Hall International Editions, 1987.
- [18] Thomas L. Naps, Brian Swander, "An Object-Oriented Approach to Algorithm Visualization - Easy, Extensible, and Dynamic", SIGCSE '94 Selected Papers of the twenty-fifth annual SIGCSE symposium on Computer Science Education, pp. 46-50, ACM SIGCSE, 1994.
- [19] Thomas L. Naps and Eric Bressler, "A Multi-windowed Environment for Simultaneous Visualization of Related Algorithms on the World Wide Web", SIGCSE '98 Proceedings of the twenty-ninth SIGCSE technical symposium on Computer Science Education, pp. 277-281, ACM SIGCSE, 1998.
- [20] Thomas L. Naps, James R. Eagan, Laura L. Norton, "JHAVÉ - An Environment to Actively Engage Students in Web-based Algorithm Visualizations", SIGCSE '00 Proceedings of the thirty-first SIGCSE technical symposium on Computer Science Education, pp. 109-113, ACM SIGCSE, 2000.
- [21] Willard C. Pierson and Susan H. Rodger, "Web-based Animation of Data Structures Using JAWAA", SIGCSE '98 Proceedings of the twenty-ninth SIGCSE technical symposium on Computer Science Education, pp. 267-271, ACM SIGCSE, 1998.

저자소개

정 인 기



1988 고려대학교 전산과학과
(이학사)

1990 고려대학교 대학원
수학과 전산학전공
(이학석사)

1996 고려대학교 대학원
전산과학과 전산학전공
(이학박사)

1997~ 현재 춘천교육대학교
컴퓨터교육과 교수

관심분야 : 컴퓨터과학교육, 데이터베이스, 원격교육

E-Mail : inkee@cnue.ac.kr