



## 순서도를 활용한 알고리즘 교육 시스템 설계

### Design of Algorithm Education System using Flow Chart

오경숙, 류남훈, 이상진, 이혜미, 김응곤  
순천대학교

KyeongSug Oh, NamHoon Ryu, SangJin Lee,  
HyeMi Lee, EungKon Kim  
Dept. of Computer Science,  
Suncheon National University

#### 요약

건축, 의학, 생명공학에서 우주항공에 이르기까지 다양한 분야에서 알고리즘의 개념을 정립해야 하지만 이론만으로는 이해하는데 한계가 있다. 그래서 다양한 멀티미디어 요소를 활용하여 교육하고 있지만 흥미를 유발하기에는 많은 어려움이 있으며, 이로 인해 상위 단계의 교과목 수강 시 많은 어려움을 겪고 있다. 알고리즘 및 프로그래밍은 과목 성격상 구현 원리를 이해할 수 있도록 실제로 프로그램을 실행시켜보는 것이 매우 중요하다. 본 논문에서는 시각화 프로그램으로 순서도를 활용한 알고리즘의 기본 개념과 알고리즘 학습에 있어서 필수 요소라 할 수 있는 프로그래밍 언어의 기본인 C 언어 습득을 위한 알고리즘 교육 시스템을 설계한다.

#### Abstract

The concept for algorithm needs to be established in various fields such as architecture, medicine and life science but understanding only theoretically has its limits. It is therefore educated using many multimedia factors but still is difficult to bring interests and carry out the higher level courses. For algorithm and programming, due to the subject's peculiarity it is very important to actually execute the program to understand the realization theory. In this paper, it aims to design a basic concept of algorithm using a flow chart of visualizing program and an algorithm education system to learn the basic programming C language, which is the fundamental element in algorithm learning.



## I. 서론

알고리즘을 비롯한 프로그래밍 과정은 전자공학 및 컴퓨터 관련학과를 비롯한 많은 이공계 학과에서 매우 중요한 교과목이다. 하지만 데이터구조에 대한 사전 지식을 필요로 하며, 수학적 사고 능력 및 프로그래밍 능력을 필요로 하므로 많은 학생들이 어려워하고 있다<sup>1)2)</sup>. 또한 프로그래밍 과정은 컴퓨터 관련학은 물론 건축, 의학, 생명공학, 우주항공에 이르기까지 다양한 분야에서 사용되고 있다. 이런 프로그래밍 언어를 배우기 위해서는 알고리즘의 개념을 정립해야 하지만, 이론만으로는 알고리즘을 이해하는데 한계가 있다.

지금까지의 알고리즘 및 언어 교육 프로그램은 소스 코드를 입력하면 진행단계를 보여주거나, 변수의 값들

이 변하는 과정을 순서도를 통해서 보여주는 형태를 취함으로 인해 알고리즘의 흐름을 파악하기 어려우며, 프로그래밍이 어렵다는 선입견을 갖기 쉽다.

이러한 문제점을 해결하고자 본 연구에서는 알고리즘의 기본 개념과 알고리즘 학습에 있어서 필수 요소라 할 수 있는 순서도, 그리고 프로그래밍 언어의 기본인 C 언어 습득을 위한 알고리즘 교육 시스템을 설계한다.

본 논문의 구성은 다음과 같다. II장에서는 프로그램 시각화에 대해 알아보고, III장에서는 순서도를 활용한 알고리즘 교육 시스템을 설계하며, IV장에서는 결론을 맺는다.

## II. 프로그램 시각화

알고리즘이란 비정형적으로는 "문제를 해결하거나 함수를 계산하기 위해 좇아야 할 모호함이 없는 간단한 명령들로 구성된 일련의 순서적 단계"로 정의할 수 있다. 알고리즘이란 말의 어원은 아라비아의 수학자 al-Khowarizmi에서 유래한다고 알려져 있다[1].

순서도는 그 알고리즘을 순서대로 처리할 수 있도록 표준적인 기호를 이용하여 그림으로 표현한 것이다. 프로그래밍 언어는 프로그램을 개발하기 위한 하나의 도구이다. 프로그램은 정확하고 효율적인 알고리즘과 그것을 구현하기 위한 데이터 구조의 결합체라 할 수 있다. 알고리즘의 성능과 효율, 처리 속도는 개발되는 프로그램에 많은 영향을 미친다. 같은 목적의 프로그램이라 하더라도 알고리즘을 어떻게 작성하느냐, 어떠한 제어 구조를 사용하느냐, 어떻게 기억장소를 운영하느냐 등 그 효율을 결정하는 데는 많은 요소들이 있다.

프로그램 시각화는 학습자에게 문서로 작성된 프로그램을 보는 것보다 훨씬 더 많은 양의 정보와 더욱 효과적인 이해 체계를 제공해 주며, 프로그램 수행을 연구하는데 있어서 강력한 접근법을 제공해 준다. 프로그램 시각화 시스템은 전자 계산 수업, 알고리즘 디자인, 분석, 효율성 평가, 그리고 프로그램을 문서화하는데 사용된다<sup>3)</sup>. 그리고 프로그래머에게는 프로그램을 시각화함으로써 프로그램의 이해, 개발, 그리고 디버깅 등의 작업을 하는데 도움을 준다<sup>4)</sup>.

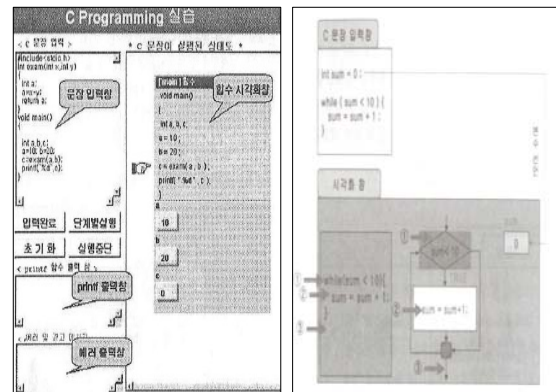
프로그램을 시각화하고자 할 때에 사용하는 방법으로는 Predefinition 방법, 함수 호출(Annotating) 방법, 선언(Declaration) 방법, 조정(Manipulation) 방법 등이 있는데, 이중 함수 호출 방법이 가장 많이 사용된다. 프로그램 시각화의 목적은 개발자에게 가공된 정보를 제공해 프로그램을 쉽게 이해시키는 데, 그 유형은 다양하다. 지금까지 행해진 연구로는 Baecker와 Marcus의 C 언어 프로그램의 Pretty printing을 위한 SEE visual compiler와 Levien의 Lisp을 위한 Visual Sntax Editor가 있다. 또한 Brown University에서 연구한 PECAN은 algebraic 프로그래밍 언어를 위해 설계되고, 추상 구문 트리의 내부 표현과 점진적 파싱을 이용해 다양한 뷰를 제공한다. 다중 뷰를 제공하는 대화적 개발 환경으로 Xerox Palo Alto 연구소에서 개발한 Smalltalk System과 Interlisp environment 등이 대표적이다<sup>5)6)</sup>.

시각화를 통한 정보의 제공은 문서 형태의 정보 제공보다도 많은 양의 정보를 효과적으로 전달해 줄 수 있

다. 프로그램의 시각화는 프로그램의 동작, 제어의 흐름, 그리고 자료의 상태변화 등을 시각적 형태로 표현하는 것을 말한다. 이것은 프로그램의 특정 상태를 정적으로 나타내는 것 뿐 아니라 전체적인 혹은 일정 범위의 수행과정을 동적으로 나타내는 것도 포함한다. 특히 후자의 기능을 제공하는 프로그램 시각화를 알고리즘 애니메이션(Algorithm Animation or Animated Graphical View)이라 한다[1][2].

현재 개발된 프로그램 시각화 시스템으로는 Balsa(Brown Algorithm Simulator and Animator) 알고리즘 애니메이션 시스템<sup>7)8)</sup>, Zeus 알고리즘 애니메이션 시스템<sup>9)</sup>, TANGO(Transition-based animation generation) 프로그램 시각화 시스템<sup>10)11)</sup>, Pavane 시각화 시스템 등이 있으며, 병렬 프로그램에서의 시각화<sup>12)</sup>, heterogeneous 환경에서의 시각화 등 여러 방면으로 연구되어 지고 있다.

국내에서도 알고리즘 교육을 위한 프로그램 시각화 시스템<sup>13)</sup> 및 JFlex<sup>14)</sup>와 Cup<sup>15)</sup>을 사용한 Java 기반의 C 언어 해석기를 선보인바 있다<sup>16)</sup>. 그림 1의 (a)는 Java Applet을 이용한 C 프로그램 함수의 시각화 실행 화면이며, (b)는 순서도 기반의 프로그램 시각화 실행 화면이다.



▶▶ 그림 1. 프로그램 시각화 예

### III. 순서도를 활용한 알고리즘 교육 시스템 설계

지금까지의 알고리즘 및 언어 교육 프로그램은 소스 코드를 입력하면 진행단계를 보여주거나, 변수의 값들

이 변하는 과정을 순서도를 통해서 보여주는 형태가 대부분을 차지하고 있다. 이 방식은 프로그램 초보자가 직접 소스 코드를 입력해야 하므로 프로그램에 대한 흥미를 잃기 쉬우며, 알고리즘에 대한 개념 및 흐름을 이해하기 어렵다는 단점을 가지고 있다. 또한 프로그램 전체의 흐름을 파악하는데 많은 어려움이 따른다.

그래서 본 논문에서는 알고리즘에 맞게 순서도를 배치하고, 순서도에 값을 채우면 자동으로 소스 코드가 생성되며, 단계별로 실행시켜 볼 수 있도록 한다. 또한 실행되는 과정을 보면서 순서도를 수정하고, 실행시켜 봄으로써 알고리즘의 개념을 파악할 수 있으며, 순서도의 사용법을 익히고, 구상한 알고리즘을 순서도로 표현할 수 있으며, C 언어를 통한 프로그래밍 방법을 습득할 수 있도록 한다. 그림 2는 알고리즘 교육 시스템의 구성도를 나타낸다.



▶▶ 그림 2. 알고리즘 교육 시스템 구성도

알고리즘 교육시스템은 사용자 입력 모듈, 사용자 입력 명령어 분석 모듈, 시각화 모듈, 사용자 인터페이스 모듈로 구성된다.

### 1. 사용자 입력 모듈

사용자 입력 모듈은 순서도 입력 모듈과 명령어 입력 모듈로 구성된다.

#### 1.1 순서도 입력 모듈

도구상자의 순서도 목록에서 원하는 순서도를 선택한 후 마우스를 이용하여 순서도 영역에 구상하고 있는 알고리즘에 맞게 배치할 수 있도록 한다.

### 1.2 명령어 입력 모듈

순서도 기호를 선택하여 순서도 영역에 넣은 후 순서도 종류에 따라 조건이나 실행문 등을 입력한다. 표1은 순서도 기호에 따른 입력과 출력 모듈이다.

표 1. 순서도 기호에 따른 입력과 출력 모드

종류	입력모듈	출력모듈
처리		
입력		
조건		
함수		

## 2. 사용자 입력 명령어 분석 모듈

사용자 입력 명령어 분석 모듈은 순서도 분석 모듈, 어휘 분석 모듈, 구문분석 모듈로 구성된다.

### 2.1 순서도 분석 모듈

사용자가 배치한 순서도의 논리적인 부분에 대해 분석할 수 있도록 한다. 예를 들어 연결선이 없는 도형이 있는지 여부를 체크 하는 등 순서도의 흐름도를 보면서 흐름을 파악한다.

## 2.2 어휘 분석 모듈

문법적으로 의미를 갖는 최소 단위인 토큰은 크게 특수 형태와 일반 형태로 분류할 수 있다. 특수 형태는 언어를 정의할 때 언어 설계자가 결정하는 지정어(Keyword), 연산자 기호, 구분자(Delimiter) 등이며, 일반 형태는 프로그래머가 프로그래밍 할 때 사용한 명칭과 상수들이다. 어휘 분석 모듈은 사용자가 입력한 조건문이나 실행문 등을 문법적으로 의미를 갖는 최소 단위인 토큰으로 분류해 내는 것이다.

예를 들어

```
a = b + 10;
```

이라는 문장이 입력된 경우

a, =, b, +, 10 그리고 ;의 여섯 개의 토큰으로 분류해 내는 과정이다.

## 2.3 구문 분석 모듈

구문 분석기(Syntax analyzer)는 파서(Parser)라고도 하는데, 어휘 분석 단계의 출력인 토큰들을 받아 소스 프로그램에 대한 에러를 체크하고 올바른 문장에 대해서는 구문 구조를 만든다.

## 3. 시각화 모듈

시각화 모듈은 순서도 인터페이스, 소스코드 인터페이스, 메모리뷰 모듈로 구성된다.

### 3.1 순서도 인터페이스 모듈

알고리즘의 진행과정을 이해하기 쉽게 시각화하는 모듈로 순차구조, 선택구조, 반복구조에 해당하는 명령들을 순서도로 도식화하여 표시한다.

### 3.2 소스 코드 인터페이스 모듈

사용자가 배치한 순서도의 내용을 분석하여 자동으로 소스 코드를 생성해 내고, 알고리즘의 진행과정에 따라 해당 소스 코드를 시각화하는 모듈이다.

## 3.3 메모리 뷰 모듈

프로그램에 사용된 변수들의 목록을 표시하고, 변수들의 값의 변화를 보여준다.

## 4. 사용자 인터페이스 모듈

사용자에게 효율적인 인터페이스를 제공하기 위해 순서도의 선택이 용이하도록 도구상자로 배치하고, 풍선 도움말 등을 통한 문법 및 예제 코드 등을 사용자에게 제공한다.

## IV. 결론

알고리즘을 비롯한 프로그래밍 과목은 전자공학 및 컴퓨터 관련학과에 있어서 매우 중요하다. 하지만 데이터구조에 대한 사전 지식 및 수학적 사고 능력, 프로그래밍 능력 등을 원하므로 많은 학생들이 어려워하고 있다. 또한, 지금까지의 알고리즘 및 언어 교육 프로그램은 소스코드를 입력하여 진행과정을 보여주는 형태를 취함으로 인해 알고리즘을 이해하는데 많은 어려움이 있다.

사용자가 순서도를 선택하여 배치하고 조건문 및 실행문 등을 입력하여 진행 과정을 단계별로 실행시켜 볼 수 있도록 함으로써 알고리즘의 개념 및 흐름을 쉽게 이해할 수 있으며, 프로그램 전체의 흐름을 파악할 수 있도록 한다. 순서도를 이용한 알고리즘 교육 시스템은 알고리즘 시각화로 인해 교육의 효과를 높일 수 있으며, 알고리즘의 단계별 실행 방법의 구현으로 인해 프로그래밍 언어에 대한 쉬운 접근법의 제공으로 알고리즘의 이해를 높일 수 있다.

향후 C 언어뿐만 아니라 C#, Java 등의 다른 언어로도 변환할 수 있도록 프로그래밍 언어 모듈을 플러그인 형태로 개발하여 순서도를 활용한 알고리즘 교육에 활용될 수 있도록 많은 연구가 필요할 것으로 본다.

## ■ 참고 문헌 ■

- [1] 이지수, 조유근, 홍영식, 알고리즘, 한국방송대학교 출판부, 2000.
- [2] 조유근, 홍영식, 이지수, 알고리즘, 이한출판사, 2002.
- [3] Adam Beguelin, Jack Dongarra, Al Geist, and Vaidy Sunderam, "Visualization and Debugging in a Heterogeneous Environment," IEEE Computer, Vol.26, No.6, pp.88-95, 1993.
- [4] Gruia-Catalin Roman and Kenneth C. Cox, "A Taxonomy of Program Visualization Systems," IEEE Computer, Vol.26, No.12, pp.11-24, 1993.
- [5] David R. Barstow, Howard E. Shrobe, & Erik Sandewall, "Interactive Programming Environments", McGraw-Hill, 1984.
- [6] Shi-Kuo Chang, "Principles of Visual Programming Systems", Prentice-Hall, 1990.
- [7] Marc H. Brown, "Exploring Algorithms Using Balsa-II," IEEE Computer, Vol.21, No.5, pp.14-36, May 1988.
- [8] Marc H. Brown, "MacBALSA Version Aleph.3," Manuscript, 1989.
- [9] Marc H. Brown, "Zeus: A System for Algorithm Animation and Multi-View Editing," DEC, SRC research report 75, Feb. 1992.
- [10] Jone T. Stasko, "The Path-Transition Paradigm: A Practical Methodology for Adding Animation to Program Interfaces," Manuscript, College of Computing, Georgia Institute of Technology, June 1991.
- [11] Jone T. Stasko, "XTANGO Algorithm Animation Designer's Package," Manuscript, College of Computing, Georgia Institute of Technology, Oct. 1992.
- [12] Jone T. Stasko, "A Methodology for Building Application-Specific Visualization of Parallel Programs," Technical Report, Graphics, Visualization, and usability Center, Georgia Institute of Technology, 1992.
- [13] 이학철, 김희철, 이상호, "알고리즘 교육을 위한 프로그램 시각화 시스템의 구현", 한국정보과학회 1995년도 가을 학술발표논문집, 제22권, 제2호, pp.761-764, 1995.
- [14] <http://www.jflex.de/>
- [15] <http://www.cs.princeton.edu/>
- [16] 오세광, 유광호, 하상호, "Java Applet을 이용한 C 프로그램 함수 실행의 시각화", 한국정보과학회, 2003년도 가을 학술발표논문집, 제30권, 제2호, pp.577-579, 2003.