

# 초등정보과학영재용 프로그래밍 교육 (비주얼 베이식을 이용한 접근)

강성원, 이해정\*, 이재호\*\*  
인천산곡초등학교, 한양여자대학\*, 경인교육대학교\*\*

## 요 약

컴퓨터 기반 산업이 활성화되고 이에 따라 학생들을 위한 IT 교육이 급성장 하였다. 국가적으로 정보화에 부합되는 인간을 양성하기 위해 다양한 IT 관련 교육이 실시되고 있다. 특히 정보과학영재에 대한 관심이 높아지면서 정보 과학 영재를 위한 다양한 프로그래밍 언어 교육이 활성화되어 일선 교육청과 각 대학에서 영재를 위한 프로그래밍 교육이 실시되고 있다. 하지만 대부분의 학생들이 딱딱한 알고리즘 교육과 일부 특정 언어에 치중한 교육에 싫증을 느끼고 있으며 프로그래밍 교육이 시험위주로 이루어지고 있다. 본 논문에서는 다른 프로그래밍 언어보다 배우기 쉽고 다루기 쉬우며 초보자도 쉽게 사용할 수 있는 비주얼 베이식을 이용한 프로그래밍 교육과정을 개발 적용하였다. 이를 통해 학생들은 실제로 다양한 방식으로 자신이 배운 프로그래밍 지식을 홈페이지를 통해 적용하게 하면서도 프로그래밍 언어에 대한 기초를 쌓을 수 있다.

## Programming Education for the Gifted of Elementary School Students in Information Science (Approach Using Visual Basic)

Sungwon Kang, Ae-Jung Lee\*, Jaeho Lee\*\*  
Inchon Sangok Elementary School,  
Dept. of Computer Info. System, Hanyang Woman's College\*  
Dept. of Computer Education, Gyeongin Nat'l Univ. of Education\*\*

## ABSTRACT

Computer based industry is activated and IT education for student has developed quickly. Variety of related IT education has performed. Especially interest of Gifted of Information Science is increased , and programming education for Gifted of Information Science is carrying out in Office of Education and college. But most student are tired of learning tedious algorithm, language centered programming, and test focused learning. In this paper we offer opportunity to learn programming for students through implementing programming curriculum using Visual Basic more easily learning than other programming language, actually apply it in their homepage. and prepare ground of programming.

## 1. 서론

현대 사회에서 개인의 자아실현이라는 관점 뿐 만 아니라 국가·사회적 관점에서도 인재를 양성하여 발전의 토대가 되게 한다는 면에서 중요한 의미를 가지고 있는 영재교육은 수학, 물리학 등과 같은 순수과학 분야와 더불어 정보과학분야와 같은 다양한 분야에서 교육이 이루어지고 있다.

현재 정보과학영재를 위한 교육내용은 프로그래밍 언어, 응용 프로그램 사용, 웹 설계 및 디자인, 창의력 개발(문제해결능력) 등으로 구성되어 있다. 이 가운데 프로그래밍 언어 교육은 정규시간에 학습할 기회가 거의 없는 분야고 수학, 물리 등과 같이 초등학교에서부터 그 기초를 이루는 공부를 꾸준히 해온 분야가 아니기 때문에 정보영재교육에 참여한 학생들은 극소수 학생을 제외하고는 우수한 프로그래밍 능력은 그만두고라도 심지어 특정 언어의 컴파일 방법 등 기초적인 프로그래밍 학습 환경에도 생소한 학생들이 대부분이다. 이에 컴퓨터를 이용하여 창의적으로 문제를 해결하기 위해서 우선적으로 기초 프로그래밍 능력의 함양이 절대적으로 필요한 것이다[1]. 프로그래밍 교육은 정보과학영재 교육의 핵심적인 분야로 다양한 언어를 기반으로 영재 교육이 실시되고 있다. 그러나, 기존 프로그래밍 학습에 대한 접근은 프로그래밍 학습에서 '무엇을 배워야 할지?(what to know?)'에 대한 연구가 부족했던 것이 현실이다. 또한 정보영재교육에 대한 이해부족과 대학입시 위주의 교육으로 인해서 체계화되지 못한 상태로 진행되고 있어 표준화된 교육과정 요소 없이 특정언어에 치우친 상태에서 시험을 대비한 프로그래밍 교육으로 진행되는 경우가 많다.

프로그래밍과 알고리즘에 대한 이해는 특정한 언어 중심으로 설명될 수도 있지만 프로그래밍 언어란 것은 학습하게 될 로직이나 알고리즘을 설명하기 위한 도구로서 사용된 것이 그 자체가 교육의 목적이 될 수는 없다. 따라서, 프로그래밍 필수 학습요소를 선정하고

이를 통해 교육과정을 설계한다면 보다 정보과학영재에게 적합한 효율적인 프로그래밍 교육이 실시될 것이다.

본 논문에서는 프로그래밍 교육의 필요성을 알아보고 교육언어 선정과정을 통하여 비주열 베이식을 초등정보과학영재를 위한 프로그래밍 언어로 제안하였다.

## 2. 정보과학영재의 정의

영재의 정의에 대해서는 많은 선행 연구가 있었으나 아직까지 명확하고 모두가 합의할 만한 정의는 내려지지 않고 있다. 한종하(1987), 조석희(1989), 이군현(1992) 등의 일반적인 과학영재의 정의와 나동섭 외(2001)의 정보과학영재에 대한 정의를 살펴보면 <표 1>과 같다.

<표 1> 과학 및 정보과학영재의 정의에 대한 제 의견

정의주체	정의 내용
한종하	일반적으로 높은 지능을 가지고 있으며 과학 및 수학분야에 뛰어난 학업성취를 보이며 과학학습에 대한 강한 학습 의욕과 높은 탐구동기 등의 심리적 특성을 보이고 사람 [2]
조석희	동일 연령의 다른 사람들과 비교하여 고학력 능력, 창의력, 과제에 대한 집착력의 모든 영역에서 85% 이상에 속하고, 그 중 한 영역에서는 적어도 98% 이상에 속하며 과학분야의 탐구활동에 강한 흥미와 긍정적인 태도를 소유한 사람 [3]
이군현	일반적으로 지능수준이 높을 뿐만 아니라 수학적 재능이 뛰어나며, 추상적 언어력이 뛰어나고, 일반적으로 학업성취의 속도가 빠르며, 새롭고 창의적인 과업에 몰두하기를 즐기고, 사고의 개방성과 융통성이 높은 사람 [4]
나동섭 이재호	발생된 문제 또는 과제에 대하여 흥미와 관심을 갖고, 이의 해결을 위해 정보에 대한 지식과 우수한 지적 능력을 동원하여 문제를 정확히 이해하여 수학적 모델을 구성할 수 있고, 컴퓨터 또는 인터넷 등의 새로운 기술이나 지식을 보다 빠르고 유연하게 습득할 수 있는 능력과 정보 기술 활용 능력을 바탕으로 수렴적 또는 발산적 사고과정을 거쳐 과제해결에 필요한 정보를 수집하며, 또한 수집된 정보를 분석, 종합, 일반화, 특수화의 과정을 통하여 가공함으로써 문제를 해결하고, 새로운 정보를 창출해 낼 수 있는 능력을 지닌 아동 [5]

이상과 같은 제 견해를 종합하여, 본 논문에서는 정보과학영재를 “주어진 문제를 파악, 이해, 분석하고 정보통신기술 활용능력을 바탕으로 새로운 정보를 수집·가공·재창출 할 수 있는 아동”이라고 정의한다.

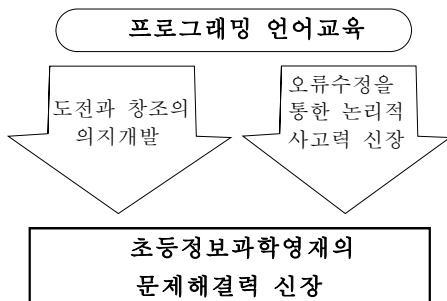
### 3. 프로그래밍 언어교육

#### 3.1 프로그래밍 언어의 교육적 의미

컴퓨터 프로그래밍은 다음 두 가지 측면에서 교육적 의미를 가진다고 할 수 있다[6].

먼저 오류 수정 활동을 통해 논리적 사고력을 향상시킬 수 있다. 오류 수정은 여러 가지 어려움을 하나 하나 처리해 감으로써 상대적으로 개선된 프로그램을 만들어 감을 의미한다. 앞에서 언급한 것과 같이 이러한 과정을 통하여 오류를 스스로 진단하고 하나 하나의 알고리즘을 분석하여 수정함으로써 논리적 사고력 향상에 도움을 주게 된다는 것이다. 이때 중요한 점은 프로그램의 질을 판단하고 평가하는 주체가 아동이 되어야 한다. 즉 교사의 일방적인 지적이 아닌 자신의 입장에서 프로그래밍 오류를 수정해 감으로써 논리적 사고력 향상에 도움을 준다는 것이다.

둘째 프로그래밍 언어 학습을 통해서 컴퓨터에 대한 이해를 돕고 스스로 문제를 해결하도록 한다는 측면에서 새로운 도전을 제시할 수 있다는 것이다. 단순한 응용 프로그램에서는 기대할 수 없는 새로운 도전과 창조의 의지를 가질 수 있도록 도움을 준다는 것이다.



<그림 1> 프로그래밍 언어 교육의 효과

#### 3.2 프로그램 언어의 학습을 위한 접근 방법

전통적 프로그래밍 학습은 구조론(syntax)을 중심으로 이루어졌기에 프로그래밍 과정이 컴파일과의 씨름을 통해서 이루어졌다. 프로그래밍 언어 교육이 문법에 대한 기계적 암기나 사용법을 익히는데 치중한다면 이는 학습자의 인지 부담이 크며 논리적 사고력을 기르는데 적합하지 못하다고 한다. 따라서 단순한 알고리즘의 암기나 프로그래밍 언어를 익히는데 국한하지 않고 프로그래밍 언어에 쉽게 접근, 이용할 수 있도록 하여 흥미를 유발하고 지속적인 학습 능력을 유지해 가도록 하는 것이 무엇보다 중요하다고 할 수 있다.

최근의 프로그램 언어의 접근법에는 구조 및 절차 학습에서 자유로운 Syntax Free Approach, 프로그램 언어를 읽고 쓸 수 있는 문법을 강조한 접근법 Literacy Approach, 문제 해결 과정을 중심으로 하는 Problem-solving Approach, 컴퓨터와의 상호작용을 중시하는 Computation as Interaction 등 여러 가지 접근법이 대두되고 있다[10].

#### 3.3 프로그래밍 교육을 위한 비주얼 베이식

프로그래밍 언어의 선택에 있어서 고급 언어의 선택이 어셈블리 언어나 기계어보다 배우기 쉽기 때문에 고급 언어를 선택하는 것이 좋다. 또한, 쉽게 다룰 수 있고 접근하기 쉬워야 한다[11]. 그렇기 때문에 많은 사용자 층을 가지고 있어 자신의 창의적인 아이디어와 결과물을 만들고 이를 웹을 사용하는 전 세계인들과 함께 공유할 수 있는 비주얼 베이식은 아동의 창의성과 학습 동기를 자극하는데 탁월하며 초등학생에게도 적합한 프로그래밍 언어이다.

### 4. 프로그래밍 언어 교육 내용

#### 4.1 프로그래밍 언어교육 요건

김홍환은 프로그래밍 언어의 효과적인 교육을 위한 요구조건으로 다음과 같이 지적하였다[7].

첫째, 프로그래밍 언어는 실습을 통해서 익혀야 한다. 따라서 실습이 강화되어야 하고, 이를 수업, 숙제, 평가 등으로 분리하여 지속적으로 실습을 하도록 유도할 수 있어야 한다.

둘째, 언어는 문법에 대한 명료한 설명과 예제를 사용한 실습 설명을 제공해야 한다.

셋째, 자연언어와 마찬가지로 프로그래밍 언어에도 용어 및 숙어의 개념이 존재한다. 예를 들면, 정규적인 반복이 필요한 경우에는 for 문장을 사용하는 것이 좋든지, 배열이나 리스트의 내용을 차례로 접근할 때는 전형적인 for 문장을 사용하는 것이 있다든지, 누적해서 더한다든지, 곱하는 경우를 들 수 있다. 이에 대한 풍부한 자료와 찾기 기능은 매우 유용하다.

넷째, 프로그래밍 언어에서 많이 사용하는 함수나 헤더 파일에 대한 내용을 알기 쉽게 전형적인 용법을 사용하여 설명해 주는 도움말 기능이 있어야 한다.

이를 볼 때 프로그래밍 교육교재의 개발을 위해서는 다양한 실습과 과제, 명료한 설명이 될 수 있는 다양한 언어예제의 개발, 프로그래밍의 기본 원리를 익힐 수 있는 기법을 전형적인 사용법 중심으로 개발해야 된다고 할 수 있다.

#### 4.2 프로그래밍 교육 요소

일반적인 프로그래밍 학습 요소를 정리하면 아래와 같다.

<표 2> 프로그래밍 교육요소

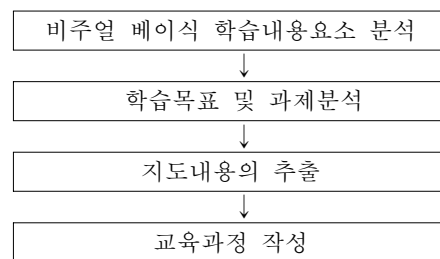
구분	◆ 언어 공통적인 규칙
연산자	◆ 연산자 - 단항, 산술, 쉬프트, 관계, 비트논리, 일반논리, 3항, 대입 등 ◆ 연산자 우선 순위
변수	◆ 변수의 정의, 선언 및 사용 - 변수명, 선언, 참조(reference), 값(value)
상수	◆ 상수 - 키코드, 수학 상수
수식	◆ 산술식, 논리식, 문자열식
제어문	◆ 조건문 - if문, else문, else if문, switch case ◆ 반복문 - while 문, for 문, do ~ while 문

자료형	◆ 형선언 ◆ 열거 자료형 ◆ 기본 자료형 - 수(number), 논리형(Boolean), 문자(Character) ◆ 구조적 자료형- 배열 ◆ 형 변환
함수	◆ 함수의 정의 ◆ 함수의 호출 ◆ 사용자 정의 함수

### 5. 비주얼 베이식 프로그래밍 교육과정 개발

#### 5.1 교육과정 내용 추출과 단계 구분의 원칙

비주얼 베이식을 이용해서 프로그래밍 교육과정을 개발하기 위해서는 우선 프로그래밍의 개념을 학습하기 위한 비주얼 베이식의 학습내용을 분석해야 한다. 추출된 내용은 난이도에 따라 학습 순서에 따라 계열성을 구분하여야 하며 선정된 내용이 프로그래밍 교육과정 내용으로 타당한 것인가에 대한 타당성을 검토하여 내용 요소 학습을 위한 학습목표를 설정하고 지도내용을 추출한 후 교수 요목을 작성하여 교육과정을 완성하는 단계를 거치게 된다. 교육과정 내용을 교육과정 내용을 추출한 교육과정 편성의 절차는 <그림 2>와 같다.



<그림 2> 비주얼 베이식 교육과정 개발 절차

#### 5.2 비주얼 베이식 기본 학습 내용 요소

비주얼 베이식을 활용한 프로그래밍 교육과정을 개발하기 위하여 기본적으로 학습할 필요가 있는 학습 내용요소들을 정리하였다. 비주얼 베이식 프로그래밍 교육과정을 설계하기 위한 기본적인 학습 요소들을 정리하면 <표 3>과 같다[8][9].

<표 3> 비주얼 베이식 기본 학습 내용 요소

구분	◆ 내용
프로젝트 Project	같은 일을 하는 관련된 파일들의 모임.
폼 Form	화면에 표시될 창(Window)
컨트롤 Control	폼 위에서 여러 가지 일을 하는 것.
개체 Object	모든 작업 단위나 대상. Form + Control
속성 Property	폼이나 컨트롤들의 작동과 특성을 결정하는 것.
이벤트 Event	프로그램 사용자나 시스템에 의해 어떤 동작이 취해지는 것.
모듈 Module	Visual Basic 코드로 구성된 단위.
프로시저 Procedure	일정한 기능을 수행하는 코드의 집합
함수 Function	매개변수를 전달하여 값을 얻어낼 수 있는 재사용할 수 있는 코드의 집합 (block)
메서드 Method	특정 개체가 특정한 작동을 하게 하는 일종의 함수
연산자 Operator	하나 또는 그 이상의 값을 계산할 때 사용되는 용어
디버그 Debug	프로그램을 점검하거나 수정하는 일련의 작업
컴파일 Compile	프로그램 단독으로 실행할 수 있도록 실행형 파일을 만드는 작업
변수 Variables	일정 크기의 데이터를 저장하는 메모리의 주소값
상수 Constant	변화되지 않는 값(수나 문자, 문자열)
데이터 형 Data types	자체적으로 연산이 가능한 값(value)의 집합. 문자열(String), 정수형(Integer), 참, 거짓(Boolean)등
매개변수 Argument	함수에 값을 전달하기 위한 공간 또는 장소
배열 Array	같은 형의 많은 변수를 만들 때 사용. 정적 배열과 동적 배열
비교문	변수나 값, 상수의 연산을 통하여 진리의 값이 되면 어떠한 일을 수행하는 문장. If~Then~Else, Select Case
반복비교	비교문이 참일 경우까지 어떤 코드를 반복적으로 수행하는 문장. Do~Loop, For(Each)~Next
클래스 Class	새로운 형태의 오브젝트를 정의해서 만들어 내기 위한 데이터 형
생성자 Constructor	선언된 클래스의 형태를 가지고 있는 개체가 생성될 때 실행되는 메서드
소멸자	생성된 개체가 소멸될 때 실행되는 메서드
DAO	Database Access Object. 데이터 베이스 엔진에 의해 정의되어 있는 개체형

위에서 제시된 비주얼 베이식 학습 기본 요소들은 일반적인 비주얼 방식의 프로그램 개발툴을 통해 프로그래밍 언어를 학습하기 위한 해당 프로그램 개발도구의 사용법을 익히는 과정과 유사하다. 어떤 개발 언어 선택하여 지도한다고 할지라도 언어적인 요소를 학습하는 것과 더불어 개발 도구의 기본적인 프로그램 개발 방식과 규칙, 툴의 사용법을 학습할 필요가 있는 것이다.

### 5.3 지도내용 개발의 원칙

비주얼 베이식을 통해 학습할 수 있는 내용 요소들은 많을 수가 있으므로 적절한 지도내용 개발의 원칙이 없이는 내용만이 많은 교육 과정이 될 수 있다. 보다 효과적이면서도 프로그래밍에 대한 기초를 학습할 수 있는 프로그래밍 교육과정은 몇 가지 원칙을 통해 개발되어 프로그래밍 학습내용 요소를 모두 포함하면서도 전체적인 교육과정의 일관성이 있는 교육과정이 개발될 필요가 있다. 비주얼 베이식 프로그래밍 교육과정 개발은 아동의 흥미를 지속시키고 효과적인 프로그램 언어의 지도에 초점을 맞추어 세 가지 개발의 원칙을 가지고 개발하도록 한다.

첫째, 초급, 중급, 고급 3단계로 나누어 교육과정을 개발한다.

정보영재라고 할지라도 아동들이 가지고 있는 컴퓨터 프로그래밍 능력은 서로 다를 수가 있다. 서로 다른 프로그래밍 능력을 배려하기 위해서는 교육과정을 3단계로 나누어 자신의 단계에 맞는 학습을 진행할 수 있도록 하고 이수 시간에 있어서도 학습자의 성취와 이해에 따라 다르게 진행할 수 있도록 한다. 초급, 중급, 고급의 3단계는 학습하는 과정을 건너뛰는 단계가 아니라 세 가지 모두를 포함하는 과정을 말하며 상위 단계에 있는 학습자는 하위과정의 학습내용을 학습하는데 보다 적은 시간과 노력으로 학습될 수 있음을 나타낸다. 예를 들어 고급과정을 학습하고 있다는 것은 초급과 중급과정 내용을 모두 알고 있다는 것

을 말하며 프로그래밍에 대해서 기본적인 지식을 가지고 있는 초등정보과학영재는 초급과정을 학습하되 적은 시간이 소요됨을 말한다.

둘째, 34개 주제의 교육과정을 제작한다. 34개 주제는 비주얼 베이식 프로그래밍 교육과정을 이해하기 위한 주제의 개수를 말하는 것으로 34시간의 내용을 의미하지 않는다. 34개의 주제는 학습이해가 빠른 초등 정보과학영재에게는 짧은 시간에 마스터될 수도 있고 반대로 더 많은 시간이 소요될 수도 있다.

셋째, 품의 디자인적인 요소와 컨트롤의 속성, 일반적인 프로그래밍 개념 요소가 한 개의 주제에서 함께 학습되도록 이루어져야 한다. 비주얼 베이식 프로그래밍 교육과정은 1개의 주제를 학습하면서도 일반적인 프로그래밍 요소와 품의 디자인과 속성을 다루는 내용이 함께 이루어지도록 교육과정을 개발하도록 하며 실무 예제코드와 실습 가능한 예제를 개발하도록 한다.

#### 5.4 비주얼 베이식 프로그래밍 교육과정 내용

이를 통해 다음과 같은 비주얼 베이식 프로그래밍 교육과정 요소를 추출할 수 있다. 언어의 지도에 초점을 맞추어 초급, 중급, 고급의 세 단계로 세분화하였으며 이 과정을 통해 프로그래밍에 대한 기초와 이해를 학습할 수 있도록 하였다. 단계에서 나누어진 초,중,고는 프로그래밍 교육 내용의 난이도를 나타낸 것이 아니라 단계적인 수준을 말한다[13].

<표 4> 비주얼 베이식 프로그래밍 지도내용 추출

순	주제	학습 내용	단계
1	규칙	• 프로그래밍 규칙 • 프로그래밍 용어 설명	초
2	개체	• 개체의 종류 • 개체의 속성 • 개체의 사용법	초
3	이벤트	• 이벤트 프로시저의 종류 • 이벤트 프로시저 사용법 • 버튼 이벤트	초
4	디버깅과 컴파일	• 디버깅 • 컴파일과 배포본 만들기	초
5	변수	• 변수의 선언과 대입 • 변수의 값 유형	초
6	상수	• 상수의 종류 • 예약어	초

7	데이터형	• 데이터(data)의 종류 • 데이터 형 변환과 다루기	초
8	원시 데이터형	• 숫자형 • 문자열형과 문자열의 조작	초
9	연산자	• 연산자 우선 순위와 • 연산자 결합법칙 • 대입연산자(=) • 단항연산자	초
10		• 산술연산자 • 관계연산자 • 기타연산자	초
11	구문	• 구문의 종류와 기본 문법 • 구문의 사용법	중
12		• With~End With 문	중
131		• 조건문의 종류 • If~Then 문 • If~Then~Else문 • If~Then~ElseIf문	중
14		• Select Case문 • 다중 조건문	중
15		• 반복문의 종류 • For~Next 문	중
16		• Do While~Loop 문 • Do~Loop While 문	중
17		• Do Until~Loop 문 • Do~Loop Until 문 • 순환문 빠져나가기	중
18		• 함수의 개념 • 함수 만들기	중
19	함수	• 함수에 값 전달하기 • 함수 종료 및 값 리턴	중
20		• 함수사용 범위 • 함수 영역 • 함수 매개 변수	중
21		• 내부함수 • 사용자 정의 함수	중
22	메뉴 편집기	• 메뉴 편집기의 사용법 • 메뉴 편집기의 활용	중
23	배열	• 배열의 정의 • 배열의 구조 • 배열의 원소	고
24		• 배열 만들기 • 원소의 추가 • 원소의 삭제	고
25		• 다차원 배열 • 배열의 응용	고
26	파일 다루기	• 파일의 처리 • 순차 파일 • 랜덤 파일	고
27	DB 다루기	• 데이터베이스의 정의 • 데이터 다루기	고
28		• DAO • 데이터베이스 조작	고
29	모듈	• 모듈의 정의 • 모듈 사용법	고
30		• 모듈 활용하기	고
31	클래스	• 클래스 모듈의 정의 • 클래스 모듈 맞보기	고
32	API 함수	• API 함수의 정의 • API 함수 맞보기	고
33	통합	• 통합실습	고
34	실습	• 통합실습	고

위에서 제시된 비주얼 베이식 프로그래밍 교육과정은 총 34개의 주제로 구성되어 있다. 아래 <표 5> 비주얼 베이식 프로그래밍 교육과정 개발의 예시 자료를 보여준다.

<표 5> 비주얼 베이식 프로그래밍 교육과정 예시

주제	13 조건문	난이도	중급
목표	<ul style="list-style-type: none"> <li>조건문의 가장 기초인 If 구문의 선언 방식을 이해할 수 있다.</li> <li>If 구문의 개념적인 흐름도를 이해하고 순서도의 형식으로 그릴 수 있다.</li> </ul>		
내용	<p>① if 선언문 방식 If 조건식 Then 조건식이 만족할 때의 명령문 EndIf</p> <p>② 설명 조건식 true 또는 false인지 여부를 평가하는 표현식 구문 조건이 true인 경우 실행할 명령문 설명 조건이 참이면 명령문을 실행한다. 조건이 거짓이면 명령문을 건너뛰고 다음 명령문을 실행한다.</p>		
코드	<pre>If A=0 Then Label1.Caption="A는 0입니다." EndIf</pre>		

## 5.5 비주얼 베이식 프로그래밍 교육과정의 적용

### 5.5.1 적용의 방향

초등정보영재를 위한 프로그래밍 교육과정 개발은 정보 영재아에게 프로그래밍에 대한 기초적인 지식과 개념을 학습하게 하여 향후 학습하게 될 다양한 프로그래밍 과정의 교두보를 마련하게 하는데 그 의의가 있다. 따라서 교육과정 내용을 단순하게 전달하는데 그쳐서는 안되며 학습자가 확실하게 이해하고 다음 단계로 넘어갈 수 있도록 지도하는 것이 필요하다.

### 5.5.2 적용의 예

이러한 적용은 크게 이론부분과 실습 부분으로 나뉘게 되는데 이론 부분은 프로그래밍 이론을 익히는데 중점을 두고 있으며 실습 부분은 학습한 프로그래밍 기초를 이용하여 실습예제를 통해 직접 확인하는 과정을 말한다.

#### 1)프로그래밍 이론 과정의 예

프로그래밍 교육과정을 지도하면서 이론부분에서 학습될 수 있는 교육내용의 세부적인

예는 다음 표와 같다.

<표 6> 프로그래밍 이론 과정 예

타이머와 조건문
<p>◆ 폼이 실행되면서 변수가 선언되고 그림이 표시됨.</p> <pre>Private Sub Form_Load() i = 0 Image2.Picture = Image1(i).Picture End Sub</pre>
<p>◆ 타이머에 의해 변수값이 변하며 조건문을 통해 변수값으로 그림을 바꿔게 함.</p> <pre>Private Sub Timer1_Timer() i = i + 1 If i = 8 Then i = 0 End If Image2.Picture = Image1(i).Picture End Sub</pre>

#### 2) 실습 과정의 예

실습과정의 예는 프로그래밍 개념 학습 후 이를 실제적으로 적용하여 학습할 수 있도록 구성되어야 한다. 단순히 비주얼 베이식의 기능적인 요소를 활용하는 것보다는 프로그래밍적인 요소를 많이 포함하고 있으면서도 학생의 창의성을 촉진하면서 개념을 확실히 익힐 수 있도록 예제를 구성한다.



<그림 3> 실습과정 예제 화면

위 예제에 사용된 코드는 다음과 같다.

<표 7> 실습예제 코드

코드	학습요소
Private Sub Cmd1_Click() Shp1.Shape = 0 End Sub	▶이벤트 프로시저
Private Sub Cmd2_Click() Shp1.Shape = 2 End Sub	▶컨트롤의 속성 설정
Private Sub Cmd3_Click() Shp1.Width = 370 Shp1.Height = 370 HSc1.Value = 0 VSc1.Value = 0 End Sub	▶스크롤 바 의 값(value) 조절
Private Sub HSc1_Change() Shp1.Width = 370+HSc1.Value*370 Lbl2.Caption = HSc1.Value + 1 End Sub	
Private Sub VSc1_Change() Shp1.Height = 370+VSc1.Value*370 Lbl2.Caption = VSc1.Value + 1 End Sub	

## 6. 결론 및 제언

본 연구에서는 초등 정보과학영재를 위한 프로그래밍 교육의 필요성을 탐색하여 프로그래밍 교육내용 요소를 추출하였으며, 효과적인 프로그래밍 교육을 위해서 프로그래밍 언어 선정기준을 정하고 이를 통해 비주얼 베이식을 교육용 프로그래밍 언어로 선정한 후 프로그래밍 교육과정을 개발하였다.

초등 정보 과학영재용 프로그래밍 교육과정 개발을 통해 다음과 같은 효과가 기대된다.

첫째, 각급 학교 및 기관에서 교육할 수 있는 프로그래밍 교육내용의 체계가 수립되고 지향점을 가진 초등 정보과학영재 교육이 이루어질 수 있을 것이다.

둘째, 프로그래밍에 대해서 전혀 알지 못하는 초등 정보과학영재일지라도 쉽게 프로그래밍의 기초를 확립할 수 있을 것이며 더불어 중등 진학 후 지속적으로 실시하게 될 상위단계의 프로그래밍 교육의 기초를 형성할 수 있을 것이다.

셋째, 비주얼 베이식의 쉬운 사용법과 다양한 기능은 아동의 창의성을 자극하고 발산하

는데 효과적인 프로그래밍 교육과정이 될 것이다.

향후 계속해서 비주얼 베이식을 활용한 프로그래밍 교육에 대한 연구가 필요하며 다양한 로직과 예제의 개발, 학습한 프로그래밍적인 기능을 통합적으로 익히고 구현할 수 있는 다양한 프로젝트의 개발이 필요하다. 그 외 무엇을 가르칠 것인가(What to teach?)에 이은 어떻게 가르칠 것인가(How to teach?)에 대한 교수법 연구가 필요하다.

## 참고문헌

- [1] 유철중, “초, 중등학교에서의 정보 영재를 위한 컴퓨터 프로그래밍 교육”, 전북교육 통권 제27호, pp.8-15, 2002
- [2] 한중하, “과학영재교육론”, pp.34-40, 1989.
- [3] 조석희 외 1인, “국민학교 고학년 과학영재 판별 도구의 타당화 연구”, 한국교육개발원 연구보고서 RR89-6, 1989.
- [4] 이군현, “영재교육학”, pp.45-67, 1989.
- [5] 나동섭, 이재호, “정보과학영재를 위한 교육분야 정의”, 한국정보교육학회, pp.378-379, 2001.
- [6] 이경화, “초등학생을 위한 로고 프로그래밍 지도방안”, 한국정보교육학회 2002년 하계학술논문집, 제 7권, 제 2호, pp.303-304, 2002.
- [7] 김홍환, “프로그래밍언어의 효과적인 교육을 위한 CAI 기법 연구”, 응용과학연구 제 10권 제 1호, pp.9-22, 2001.
- [8] 이형배, “이형배의 비주얼 베이식 6”, 사이버 출판사, 1999
- [9] 주경민 외 2인, “Visual Basic Programming Bible Ver 5”, 영진출판사, 1998
- [10] Sally Fincher, What are we doing When we teach programming? In Frontiers in Education '99, pages 12a41 - 5. IEEE, November 1999.
- [11] <http://www.seoul-e.ac.kr/guide/hakbo/last/327/7-1.html>



## 저자 소개

### 강 성 원



1992 경인교육대학교  
(교육학 학사)

2001~현재 경인교육대학교  
초등컴퓨터교육과 (석사 과정)

현재 인천산곡초등학교 근무

관심분야 : ICT 활용 교육, 프로그래밍 교육

E-Mail : [ksungwon@dreamwiz.com](mailto:ksungwon@dreamwiz.com)

### 이 애 정

이재호 교수님께서 알려 주실 것임

### 이 재 호

이재호 교수님께서 알려 주실 것임