

교육용 한글 프로그래밍 언어 'HanScript'의 적용

정영식†

요 약

본 연구에서는 프로그램을 보다 쉽고 편리하게 작성할 수 있도록 교육용 한글 프로그래밍 언어인 HanScript를 개발하여 학교 현장에 적용하였다. 또한 HanScript를 평가하기 위해서 교육용 프로그래밍 언어의 조건을 가독성, 가작성, 교정성으로 구분하고, 이를 Visual Basic과 비교하였다. 그 결과 HanScript는 우리말식으로 표현되어 있어 Visual Basic에 비해 가독성과 교정성이 높게 나타났다. 그러나 Visual Basic에 비해 라이브러리가 부족하여 복잡한 기능을 구현하기 어려웠고, 소스 코드를 쉽게 작성할 수 있는 편집기가 제공되지 않아 실습하는 데 어려움이 많았다. 따라서 HanScript가 풍부한 라이브러리와 자세한 디버그 정보를 제공하고, 소스를 편리하게 작성할 수 있는 편집기를 제공한다면 보다 유용한 교육용 프로그래밍 언어가 될 수 있을 것이다.

키워드 : 한스크립트, 프로그래밍언어

The Development and Evaluation of Educational Hanguk Programming Language 'HanScript'

Young-Sik Jeong[†]

ABSTRACT

This study developed HanScript, which is an educational Hanguk programming language, in order to make programs easily and conveniently and applied it to the school fields. To evaluate HanScript, additionally, the researcher divided the properties of educational programming languages into easiness to read, easiness to write and easiness to debug and, based on them, compared HanScript with Visual Basic. According to the result, HanScript was expressed in Korean language style, so was easier to read and debug than Visual Basic. But it was not different from Visual Basic in debugging. Due to fewer libraries compared to Visual Basic, however, it was difficult to implement complex functions using HanScript. In addition, HanScript did not include an editor to edit source codes. Thus, if HanScript is equipped with rich libraries, detailed debugging information and an editor for source code edition, it will be a useful educational programming language.

Keywords : HanScript, Programming Language

1. 연구의 필요성 및 목적

오늘날은 과거와 같이 많이 아는 것만으로는 부족하며, 문제 해결에 필요한 정보를 획득하고 이를 이용하여 실제 생활에 적용할 수 있는 정보

† 종신희원: 한국교육개발원 연구원
논문접수: 2003년 12월 29일, 심사완료: 2004년 4월 27일

처리 능력과 새로운 아이디어를 창출하는 정보 생산 능력이 강조되고 있다.

컴퓨터 교육 중에서 가장 중요한 분야 중 하나가 프로그래밍 분야로 컴퓨터 교육은 프로그래밍을 통하여 컴퓨터를 이해하고, 문제 해결력을 기르며, 논리적 사고와 창의력을 개발하고, 프로그래밍을 통해서 프로그래밍 언어의 개념과 프로그래밍 방법론 등의 기초 개념을 습득하기 때문이다[1].

현재의 7차 교육과정의 프로그래밍 교육은 실업계 고등학교의 '컴퓨터 일반'에서 프로그래밍 언어에 대한 기초 이론과 Visual Basic을 중심으로 한 실습 교육이 이뤄지고 있다. 이러한 프로그래밍 교육은 다음과 같은 문제점을 지니고 있다.

첫째, Visual Basic을 중심으로 한 언어 교육은 논리적 사고력 향상을 위한 알고리즘 교육보다는 마우스 클릭을 통한 빠르고 시각적인 구현을 중심으로 하고 있다. 따라서 문제 해결력과 논리적 사고의 향상에 크게 도움이 되지 못하고 있다.

둘째, 기존의 프로그래밍 언어는 모든 명령문이 영문으로 되어 있을 뿐 아니라 프로그램 작성 시에도 빈번히 사용되는 변수나 함수의 이름도 영문으로만 사용이 가능하기 때문에 현재 영어를 모르거나 또는 안다고 할지라도 프로그램을 작성하는 데 많은 어려움이 있다[2].

셋째, Visual Basic 중심의 실습 환경은 다양한 웹 환경을 지원하지 못하고 있다. 웹은 언제 어디서나 프로그래밍 교육을 받을 수 있고 다양한 전문가에게 조언을 얻을 수 있으며, 교육자와 학습자 간의 상호작용이 활발하게 이뤄져 폭넓은 프로그래밍 기회를 제공할 수 있기 때문에 웹을 지원하는 언어가 필요하다.

넷째, 이러한 문제점에도 불구하고 이를 대체할만한 교육용 프로그래밍 언어가 없다.

따라서 본 연구에서는 교육용 프로그래밍 언어의 조건을 살펴보고, 이를 토대로 누구든지 쉽게 이해하고, 자기의 생각을 우리말로 옮기듯이 쉽게 프로그램을 작성할 수 있도록 우리말식 교육용 프로그래밍 언어인 'HanScript'를 개발하고자 한다. 그리고 이를 실업계 고등학교에 적용한 후 Visual Basic과 비교·검증하고자 한다.

2. 교육용 프로그래밍 언어의 조건

프로그래밍 교육이 효율적으로 운영되려면 좋은 프로그래밍 언어를 선택해야 한다. 좋은 프로그래밍 언어의 조건을 오세만은 다음과 같이 요약하였다[3].

첫째, 언어의 개념이 명료해야 한다. 문법적인 구조와 그에 따른 의미가 일관성이 있으며 단순해야 한다.

둘째, 프로그래머의 생각을 자연스럽게 표현할 수 있어야 한다.

셋째, 언어의 확장성이 우수해야 하고 좋은 프로그래밍 환경을 갖고 있어야 한다. 또한 프로그램의 호환성, 신뢰성, 모듈화, 효율성 등이 좋아야 한다.

이현호는 교육용 프로그래밍 언어가 갖추어야 할 조건을 다음과 같이 제시하였다[4].

첫째, 교육용 언어는 단순, 명료해야 한다. 프로그램을 코딩하였을 때 단순하고 개념이 명료하여야만 어떤 문제를 표현하고 해결하는 과정에서 혼란이 없다.

둘째, 이해하기 쉬워야 한다. 프로그램도 하나의 표현된 문장이기 때문에 프로그램만 보고도 쉽게 이해가 되도록 읽기 쉽게 코딩할 수 있어야 한다.

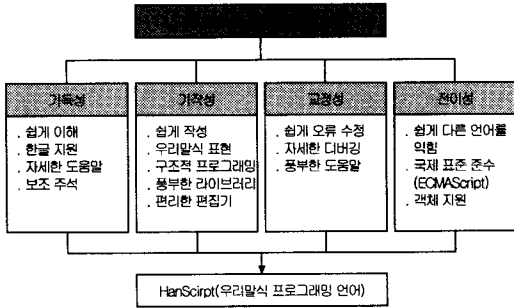
셋째, 새로운 언어로서의 전이가 쉬어야 한다. 다른 언어가 갖추고 있는 언어의 구조를 갖춤으로서 다른 언어를 배우고 익히는 데 쉽게 전이가 이루어져야 한다.

넷째, 복잡한 문제를 쉽게 해결할 수 있는 강력한 언어이어야 한다. 이를 위해서는 풍부한 라이브러리와 객체 지향 기능을 지원해야 한다. 또한 사용자가 쉽게 오류를 수정할 수 있도록 강력한 디버그 기능과 풍부한 도움말이 필요하다.

다섯째, 한글용 프로그래밍 언어이어야 한다. 프로그래밍 언어를 배우는 것이 또 하나의 외국어를 배우는 것과 같아서 안 된다. 또한 우리말과 비슷한 구조로 컴퓨터에게 지시를 내릴 수 있어야 한다.

여섯째, 웹 환경을 지원해야 한다. 웹은 언제 어디서나 프로그래밍 교육을 받을 수 있고 다양

한 전문가에게 조인을 얻을 수 있으며, 교육자와 학습자, 학습자와 학습자, 학습자와 전문가의 상호작용이 활발하여 폭넓은 프로그래밍의 기회를 제공한다. 이러한 교육용 프로그래밍 언어의 조건을 정리하면 (그림 1)과 같이 가독성, 가작성, 교정성, 전이성으로 구분할 수 있다.



(그림 1) 교육용 프로그래밍 언어의 조건

첫째, 가독성은 다른 사람이 만든 소스 코드를 쉽게 이해할 수 있음을 뜻한다. 가독성을 높이기 위해서는 변수나 함수의 이름을 한글로 표현할 수 있어야 하며 읽기 쉽도록 주석이나 보조 주석을 제공할 수 있어야 한다. 또한 자세한 도움말을 제공해야 한다.

둘째, 가작성은 프로그래머가 생각한 것은 쉽게 코드로 작성할 수 있음을 뜻한다. 가작성을 높이기 위해서 우리말을 옮기 듯이 표현할 수 있어야 하고, 풍부한 라이브러리와 편리한 편집기를 제공해야 한다.

셋째, 교정성은 프로그램의 오류를 알려주고 이를 쉽게 수정할 수 있음을 뜻한다. 교정성을 높이기 위해서는 오류의 유형을 경고, 에러, 예외 등으로 분류하여 오류 위치와 해결 방법 등을 자세히 제공해야 한다.

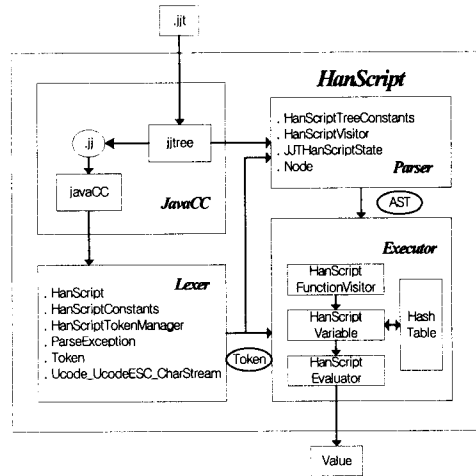
넷째, 전이성은 하나의 언어를 완벽하게 배우면 다른 언어도 쉽게 익힐 수 있음을 뜻한다. 전이성을 높이려면 ECMAScript와 같은 국제적 표준에 따라 언어를 구성하고, 객체를 직접적으로 지원하고 다양한 객체를 사용할 수 있어야 한다.

따라서 교육용 프로그래밍 언어가 가져야 할 조건인 가독성, 가작성, 교정성, 전이성을 만족할 수 있는 'HanScript' 언어를 개발하고자 한다. 또한 대부분의 실업계 고등학교에서 가르치고 있는 Visual Basic과 비교·검증하고자 한다.

3. HanScript의 개발

3.1 HanScript의 전체 구조

HanScript는 LL 파서(recursive descent)의 일종인 JavaCC를 이용하여 구현하였다. JavaCC는 2개의 실행 파일로 구성된다. jtree.exe는 토큰의 유형과 문법을 정의한 .jtt 파일을 입력받아 구문 분석기와 .jj 파일을 생성한다. javacc.exe는 .jj 파일을 이용하여 어휘 분석기를 생성한다[5].



(그림 2) HanScript의 전체 구조도

3.1.1 어휘 분석기

어휘 분석기는 사용자가 입력한 소스 코드에서 토큰을 추출하여 구문 분석기와 실행기에 전달한다. 특히 HanScript가 갖는 주요 특징 중에서 예약어와 주석에 대해 살펴보면 다음과 같다[6].

<표 1> 예약어

사용 중인 예약어	사용될 예약어
발출, 계속, 지우기, 아니면, 반복, 함수, 만약, 이면, 중에, 새로운, 반환, 자기, 자료형, 변수, 무반환, 동안, 와, 참, 거짓, 널	경우, 검사, 잠기, 묶음, 상수, 별레잡기, 기본, 해라, 나열, 내보내기, 확장, 마침내, 들어오기, 상위, 선택, 던지기

예약어는 현재 언어에서 사용 중인 예약어와

앞으로 사용될 예약어로 구분하여 제시하였다. 즉 HanScript에서 제공하는 예약어는 <표 1>과 같이 36가지이다.

주석은 단일행 주석과(//)과 다중행 주석(**/)을 제공하며 문장의 해석을 보다 쉽게 하기 위해서 우리말의 목적격 조사에 해당되는 ‘을’과 ‘를’을 보조 주석으로 도입하였다. 다음은 보조 주석을 사용한 예로서 출력 결과는 100이다. 이때 ‘를’이 보조 주석이므로 생략이 가능하다.

```
가=100.
(가)를 쓰기.
```

3.1.2 구문 분석기

구문 분석기는 소스 코드와 토큰을 입력받아 에러를 검출하고 문법에 맞는지 확인한다. 문법에 적합하면 구문 트리(AST)를 생성하여 실행기에 전달한다. 문장의 종료자로 온점(.)을 사용하며, 주요 구문 중에서 조건문과 반복문을 살펴보면 다음과 같다.

<표 2> 조건문

일반형	[<식>] 이면 <문1> 아니면 <문2>
문제	점수가 70점 이상일 때 "합격" 판정 받기
예	점수=80. [점수>=70]이면 결과="합격" 아니면 평점="불합격" (결과)를 쓰기.
결과	합격

조건문은 <표 2>와 같이 <식>의 값이 참이면 <문1>을 처리하고 거짓이면 <문2>를 처리한다.

<표 3> 반복문

일반형	[<식>] 반복 <문> [<초기값>]; [<종료조건>]; [<증감식>] 반복 <문> [변수 <선언문>; [<식 1>; [<식 2>]] 반복 <문> [<식 1> 중에 <식 2>] 반복 <문>
문제	1부터 100까지의 합 구하기
예	합계=0. [가=1;가<=100;가++] 반복 합계+=가. (합계)를 쓰기.
결과	5050

반복문은 <표 3>과 같이 4가지 형태를 갖는다. 첫 번째 반복문은 <식>의 값이 참인 한 <문>을 반복한다. 두 번째와 세 번째 반복문은 <초기값>을 한 번 지정한 후 <종료조건>을 판정해서 참이면 <문>을 실행한 후 <증감식>을 실행한다. 다시 <종료조건>을 검사해서 거짓이 되면 반복문을 종료한다. 네 번째 반복문은 <식 1>의 객체의 모든 특성을 <식 2>에 할당한 후 그 특성의 수만큼 <문>을 반복한다.

함수 표현에 있어서는 기존의 언어와는 달리 매개 개변수를 먼저 사용하고 함수 이름을 나중에 쓰도록 하였다. 즉, 다음과 같이 우리말식 표현처럼 ‘목적어+동사’ 형태로 표기하였다.

```
쓰기(가) → (가)를 쓰기.
동사+목적어   목적어+동사
```

3.1.3 실행기

실행기는 구문 트리를 탐색하면서 해쉬 테이블을 생성하고, 각 노드의 값을 평가한 후 그 결과를 출력한다. 실행기를 구성하는 주요 클래스는 <표 4>와 같다.

<표 4> 실행기의 주요 클래스

클래스	기능
HSHashTable	변수와 명명자를 관리한다.
HanScriptFunctionVisitor	함수 선언문을 처리한다.
HanScriptVariableVisitor	변수 선언문을 처리한다.
HanScriptEvaluatorVisitor	노드에서 생성된 객체를 평가한다.

첫째, HSHashTable 클래스는 해쉬 테이블에 변수를 삽입하거나 추출한다.

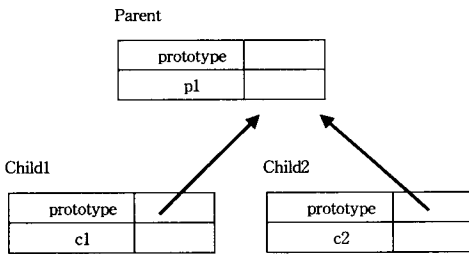
둘째, HanScriptFunctionVisitor 클래스는 함수 선언문을 처리하며, 함수의 매개 변수를 몸체에 맞게 변수로 처리하여 벡터에 저장한다.

셋째, HanScriptVariableVisitor 클래스는 변수 리스트에서 변수명을 얻어 벡터에 저장한 후 그 벡터를 반환한다.

넷째, HanScriptEvaluatorVisitor 클래스는 구문 트리를 탐색하면서 객체의 값을 평가하거나 객체 간의 연산을 수행하여 그 결과를 반환한다.

3.2 객체 모형

HanScript는 클래스 기반의 객체 지향 언어가 아닌 프로토타입 기반의 객체 지향 언어이다. 프로토타입 기반의 객체 지향 언어는 클래스를 정의하지 않지만 실행 코드에서 생성자 함수를 이용하여 객체를 생성시키고 그 객체가 갖고 있는 프로퍼티들의 초기값을 할당시킬 수 있다. 또한 모든 생성자는 프로토타입이라는 프로퍼티를 갖는다. 즉, (그림 3)과 같이 프로토타입에 상속받고자 하는 부모 객체를 연결하면 부모의 모든 프로퍼티를 상속받을 수 있게 된다.



(그림 3) 프로토타입 기반의 유전

(그림 3)과 같이 Child1, Child2의 prototype의 값을 Parent로 지정하면 Parent의 p1은 Child1, Child2에 유전된다. 또한 Parent에 p2를 동적으로 추가하면 Child1, Child2 모두 p2를 상속받을 수 있다.

따라서 HanScript는 프로토타입 기반의 객체 지향 언어기 때문에 분류, 상속, 캡슐화, 정보 은닉과 같은 클래스 기반의 객체 지향 언어가 갖는 기능을 제공하지 않는다. 그러나 미리 정의된 객체 모형을 제공하여 사용자가 쉽게 프로그래밍을 할 수 있다.

객체 모형은 객체에 들어 있는 값을 액세스할 수 있는 프로퍼티와 객체에 어떤 동작을 표현할 때 사용하는 메소드를 제공한다. 미리정의된 객체 모형에는 '전역객체', '객체', '숫자', '논리', '배열', '문자열', '함수', '수학', '날짜' 등이 있다.

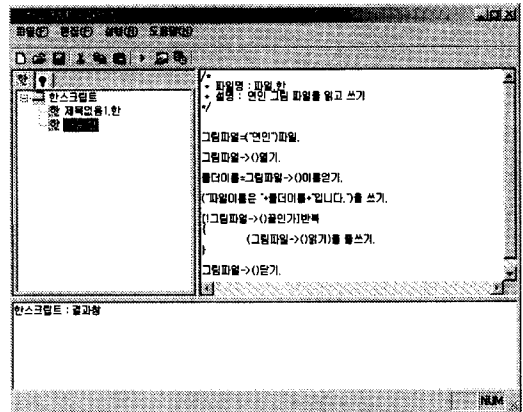
다음은 검사하고자 하는 문자열을 이용하여 '정규표현' 개체를 생성하고, '검사하기' 메소드를 이용하여 특정 문자열이 정규 표현에 올바른지를 검사하는 예제이다.

(예) 정규식=새로운 ("a[1-9]*.[0-9]*")정규표현.
가="A1.2".

```
나=정규식->(가)검사하기. //정규 표현식에 맞는가?
(나)를 줄쓰기.
<실행 결과> 참
```

3.3 편집기

구현된 HanScript를 이용하여 프로그램을 작성하기 위해서는 (그림 4)와 같이 각종 도움말과 디버그 정보를 쉽게 보여 줄 수 있는 편집기가 필요하다.



(그림 4) 소스 편집기

편집기의 주요 기능은 프로그램 작성하기, 실행 결과 보여주기, 디버그 정보 보여주기, 프로젝트 파일 관리하기, 도움말 보기 등으로 나뉜다. 특히 디버그 정보 보여주기 기능에서는 소스 코드 내에서의 오류 위치와 간단한 설명을 결과 창에 보여 준다.

4. 적용 및 평가

4.1 적용 방법

대부분의 실업계 고등학교의 '컴퓨터 일반' 교과서에 포함되어 있는 Visual Basic과 본 연구자가 개발한 HanScript를 비교하기 위해 경기도에 있는 K 정보산업고등학교에서 프로그래밍 교육을 별도로 받지 않은 반 중에서 컴퓨터 관련 교과 성적이 유사한 2개 반 학생 전체를 대상으로 <표 5>와 같이 방과 후에 교육을 실시하였다.

연구 대상자는 A반 43명, B반 45명으로 해당

반 학생 전부를 대상으로 선정하였다.

<표 5> 적용 방법

구분	내용	
연구대상	A 반	43명(HanScript 학습)
	B 반	45명(Visual Basic 학습)
수업시간	각 반별 10시간(방과 후 실시)	
수업회수	5회(1회 2시간씩)	
수업방법	강의 30분, 실습 1시간	

본 연구에서 사용된 교육 과정은 <표 6>과 같이 Visual Basic과 HanScript를 비교하는 데 필요한 최소한의 내용으로 제한하였다.

<표 6> 적용 방법

차시	단원명	재부 내용	
		Visual Basic	HanScript
1	변수와 상수	. 변수, 상수, 자료형	
2	연산자	. 산술, 할당, 관계, 연산자 우선 순위	
3	제어문	. IF문, For~Next문	. 이면, 반복, 탈출, 계속
4	함수	. Function	. 함수
5	입출력	. Print	. 쓰기, 줄쓰기

4.2 평가 방법

본 연구에서는 교육용 프로그래밍 언어의 조건 중에서 전이성을 배제한 가독성, 가작성, 교정성만을 평가하였다. 그 이유는 전이성을 평가하기 위해서는 또 다른 언어들을 배워야 하는 현실적인 어려움이 있기 때문이다.

본 연구에서는 다음과 같이 연구 가설을 설정하였다.

첫째, HanScript는 Visual Basic보다 가독성이 높을 것이다.

둘째, HanScript는 Visual Basic보다 가작성이 높을 것이다.

셋째, HanScript는 Visual Basic보다 교정성이 높을 것이다.

이러한 연구 가설을 검증하기 위해서 <표 7>과 같이 평가 도구를 작성하였다.

주어진 문항은 모두 주관식으로서 각 조건별 난이도를 3단계로 구분하였으며, 하위 단계는 간단한 구문만을 사용하고, 중위 단계는 구문과 함수를 사용하였으며, 상위 단계는 구문, 함수, 입

출력을 사용하였다. 또한 구문 문제는 20점, 함수 문제는 10점, 입출력 문제는 20점으로 하여 각 조건별 난이도에 따라 각각 20점, 30점, 50점으로 계산하였다. 그리고 프로그래밍 언어의 특징이 아닌 편집기의 도움말이나 소스 편집을 위한 편리한 기능에 의해 연구의 결과가 달라지는 것을 배제하기 위해서 실기 평가가 아닌 지필 평가를 하였다.

<표 7> 평가 도구

조건	평가 방법	문항수
가독성	소스 코드를 보고 출력 결과를 쓰기	3
가작성	출력 결과가 나타나도록 소스 작성하기	3
교정성	소스 코드 중에서 오류 부분을 수정하기	3

4.3 평가 결과

4.3.1 가독성

가독성을 평가하기 위해서 3개의 문항을 각각의 프로그래밍 언어에 따라 표현 방식만 다르게 하여 A반과 B반에 제시하였다. A반과 B반의 가독성에 대한 검증 결과는 <표 8>과 같다.

<표 8> 가독성의 검증 결과

	n	평균	표준 편차	t	자유도	유의확률 (양측)	평균차
가독성	A 43	68.95	18.405	2.280	86	.025	8.18
	B 45	60.78	15.149				

*p<.05, n=88

A: HanScript를 학습한 반

B: Visual Basic을 학습한 반

가독성을 평가한 문항에서 A반의 평균은 68.95이고, B반은 60.78로 나타났다. 이것을 독립 표본 t 검정을 통해 분석한 결과 t 통계 값은 2.280으로써 유의 확률이 .025이었다. 이 통계 값은 유의 수준 $\alpha=.05$ 보다 작으므로 A반의 평균과 B반의 평균은 유의한 차이가 있다.

따라서 HanScript는 Visual Basic보다 가독성이 높다고 할 수 있다. 그 이유는 <표 9>와 같은 함수 표현 방법에 있어서 HanScript는 보조 주석인 목적격 조사 ‘을’, ‘를’을 사용함으로써 그

뜻을 보다 쉽게 이해할 수 있다.

<표 9> 함수 비교

Visual Basic	HanScript
r=10 area=3.14*r^2 msgbox(area)	반지름=10. 넓이=3.14*반지름^2 (넓이)를 줄쓰기.

4.3.2 가작성

가작성을 평가하기 위해서 코드를 직접 작성하는 형태의 문제를 3문항을 제작하여 A반과 B반 모두에게 똑같이 제시하였다. A반과 B반의 가작성에 대한 검증 결과는 <표 10>과 같다.

<표 10> 가작성의 검증 결과

	n	평균	표준 편차	t	자유도	유의확률 (양측)	평균차
가작성	A 43	60.35	17.196	1.566	86	.121	5.13
	B 45	55.22	13.355				

*p<.05, n=88

A: HanScript를 학습한 반

B: Visual Basic을 학습한 반

가작성을 평가한 문항에서 A반의 평균은 60.35 이고, B반은 55.33으로 나타났다. 이것을 독립 표본 t 검정을 통해 분석한 결과 t 통계 값은 1.566으로써 유의 확률이 .121이었다. 이 통계 값은 유의 수준 $\alpha=.05$ 보다 크므로 A반의 평균과 B반의 평균은 유의한 차이가 없다.

따라서 HanScript는 Visual Basic보다 가작성이 높다고 할 수 없다. 그 이유는 가작성 평가 문제는 어떤 언어를 선택하느냐보다는 문제를 풀기 위한 알고리즘을 충분히 이해하고 이것을 프로그램으로 옮겨야 하기 때문에 우리말식 표현이라는 HanScript의 장점이 크게 도움이 되지 않음을 알 수 있다.

4.3.3 교정성

교정성을 평가하기 위해서 오류가 있는 소스 코드를 각각의 프로그래밍 언어에 따라 표현 방식만 다르게 하여 A반과 B반에 제시하였다. 그리고 제시된 소스 코드의 오류를 찾아 수정하게

하였다. A반과 B반의 교정성에 대한 검증 결과는 <표 11>과 같다.

교정성을 평가한 문항에서 A반의 평균은 63.84 이고, B반은 56.89로 나타났다. 이것을 독립 표본 t 검정을 통해 분석한 결과 t 통계 값은 2.123으로써 유의 확률이 .037이었다. 이 통계 값은 유의 수준 $\alpha=.05$ 보다 작으므로 A반의 평균과 B반의 평균은 유의한 차이가 있다.

<표 11> 교정성의 검증 결과

	n	평균	표준 편차	t	자유도	유의확률 (양측)	평균차
교정성	A 43	63.84	16.827	2.123	86	.037	6.95
	B 45	56.89	13.788				

*p<.05, n=88

A: HanScript를 학습한 반

B: Visual Basic을 학습한 반

따라서 HanScript는 Visual Basic보다 교정성이 높다고 할 수 있다. 그 이유는 <표 12>와 같은 조건문에 있어서 Visual Basic은 영어식 표현 방법을 사용하지만, HanScript는 일상적으로 사용하는 우리말식 표현 방법을 사용하므로 그 뜻을 보다 명확히 이해할 수 있고, 표기상의 오류를 보다 쉽게 찾아낼 수 있다. 즉, Visual Basic에서 'endif'를 붙여 쓸 경우 이것을 찾지 못하는 학생이 15.6%이었지만, HanScript에서 '아니면결과'와 같이 붙여 쓸 경우 이것을 찾지 못하는 학생이 4.7%로써 Visual Basic보다는 HanScript의 교정성이 높게 나타났다.

<표 12> 조건문 비교

Visual Basic	HanScript
if kor>=80 then result="잘함" else result="못함" end if	[국어 >=80] 이면 결과="잘함" 아니면 결과="못함".

5. 결 론

학생들이 작성된 프로그램을 쉽게 이해하고, 생각한 알고리즘을 쉽게 작성할 수 있도록 교육용 한글 프로그래밍 언어인 HanScript를 학교 현장에 적용하였다.

HanScript의 가독성, 가작성, 교정성을 평가하기 위해서 실업계 고등학생을 대상으로 방과 후 수업을 실시한 후 평가하여 Visual Basic과 비교·검증하였다. 그 결과 HanScript는 가독성과 교정성이 Visual Basic보다 우수하였다. 즉, 학생들이 프로그램을 쉽게 이해할 수 있었으며, 소스 코드 내의 버그를 쉽게 찾아내어 수정할 수 있었다. 그러나 가작성에 있어서는 두 언어 간의 차이는 없었다. 이것은 소스 코드의 한글 표현 방식보다는 알고리즘에 대한 논리적 사고에 따른 영향이 보다 크게 미치는 때문이다. 따라서 HanScript로 작성된 코드가 Visual Basic으로 작성된 코드보다 이해하기 쉽고 잘못된 코드를 보다 쉽게 수정할 수 있는 것으로 검증되었다.

앞으로 HanScript에 대한 라이브러리를 다양하게 만들고 자세한 디버깅 정보를 제공해 줄 수 있는 통합 개발 환경을 제공한다면 효과적인 교육용 프로그래밍 언어가 될 수 있을 것이다. 더욱이 HanScript는 플랫폼 독립적인 Java로 작성되었기 때문에 다양한 플랫폼과 인터넷 환경을 지원할 수 있으며 원격지간의 프로그래밍 교육에도 많은 도움이 될 것이다.

참 고 문 헌

[1] 김명렬(1989). 컴퓨터 교육용 프로그래밍 언어 CELL의 설계에 관한 연구, 홍익대학교 대학원 박사학위 논문.

[2] 강덕순(1998). 한글 프로그래밍 언어 '샘'의 교재 개발에 관한 연구. 한국교원대학교 대학원. 석사학위 논문.

[3] 오세만(1994), 컴파일러 입문, 정익사.

[4] 이현호(1996). 프로그래밍 교육에 교육용 프로그래밍 언어 'SAM'의 적용 및 평가, 한국교원대학교 대학원 석사학위 논문.

[5] Metamata(1997). JavaCC Document. [Online] Available: <http://www.metamata.com/JavaCC/javaccdocs.html>

[6] 정영식, 김명렬(2003). 자바 기반의 컴퓨터교육용 프로그래밍 언어의 설계 및 구현, 컴퓨터산업교육학회.

정 영 식



1996 춘천교육대학교
수학교육학과(교육학 학사)
2001 한국교원대학교
컴퓨터교육과(교육학 석사)
2004 한국교원대학교
컴퓨터교육과(교육학 박사)

1996~2000 경기도 하남초등학교 교사
2004~현재 한국교육개발원 연구원
관심분야 : 컴퓨터교육, 원격교육, 평생교육
E-Mail : nurunso@hotmail.com