

프로그램의 주관식 문제 자동 채점 시스템 설계 및 구현

정은미[†], 최미순^{**}, 심재창^{***}

요 약

본 논문에서는 자연어 처리의 어려움을 해결하기 위해 키워드와 불리언 연산을 이용한 프로그래밍 언어 자동 채점 시스템을 설계하고 구현하여 실험한다. 프로그래밍 언어는 정확한 문법 체제와 키워드가 존재한다. 이러한 특징을 이용하여 본 논문은 키워드와 불리언 연산을 이용한 프로그래밍 언어 자동 채점 시스템을 설계하고 구현하여 실험한다. 프로그래밍 언어를 정답 유형별로 7가지로 나누고 키워드 입력 시 접속사를 특수문자를 이용해 입력하게 하여 논리식을 바로 만들어 내어 쉽게 연산이 가능하게 한다. 제안된 자동 채점시스템의 주관식 채점이 객관적으로 잘 처리됨을 보이기 위해 컴퓨터공학 전공 학생 10명을 대상으로 같은 시험을 시험지와 웹에 동일하게 응시하게 한 후, 3명의 담당 교수에게 채점을 의뢰하고 개발한 시스템으로 채점하여 결과를 비교 분석한다. 주관식 채점에서 활용되는 방대한 자연어 처리를 배제할 수 있도록 키워드와 불리언 연산을 이용한 방식을 제안하므로 구현의 효율성을 높이고 채점을 웹으로 이관함으로써 채점자의 주관적 개입을 배제하고 결과를 빠르게 피드백 해주어 객관성과 신속성을 높이는데 목적이 있다.

Design and Implementation of Automatic Marking System for a Subjectivity Problem of the Program

Eunmi Jung[†], Misun Choi^{**}, Jaechang Shim^{***}

ABSTRACT

The purpose of this paper is to design, implement and test the automatic marking system for programming languages using key-words and boolean operations to solve the processing problems of natural languages. There are accurate grammar systems and key-words in programming languages. Using these characteristics, We have designed, programmed, and tested automatic marking system for programming languages through key-words and boolean operations in this paper. We have categorized programming languages into 7 types as the type of answer and when a professor input any key-words, the system make him put conjunction with the special character. It can be logical expressions instantly so that the system easily operates. We asked 10 students who are majoring in computer engineering to take a test on the paper and web to show how well automatic marking system that we have programmed works. Then We requested 3 professors if the subject problems marked objectively. As a result, automatic marking system proved to be appropriate. We have proposed the way of using key-words and boolean operation for prohibiting huge natural language processing in marking of subjective question. It promotes efficiency rate of programming, objectivity and speed through the transferal to the web for marking since the system prohibits marker to include personal opinion on marking and gives feedback quickly

Key words: Subjectivity test(주관식 시험), Automatic Marking(자동 채점), Programming Language(프로그래밍언어)

* 교신저자(Corresponding Author) : 심재창, 주소 : 경북 안동시 용상동 송천동 388번지(760-749), 전화 : 054)820-5645, FAX : 054)2820-6164, E-mail : jcshim@andong.ac.kr
접수일 : 2008년 12월 15일, 완료일 : 2009년 4월 4일
[†] 준회원, 안동대학교 컴퓨터공학전공, 외래교수

(E-mail : jeilc@naver.com)

^{**} 안동대학교 컴퓨터공학전공, 박사과정
(E-mail : iamcms@freechal.com)

^{***} 종신회원, 안동대학교 컴퓨터공학전공, 교수

1. 서 론

최근 정보 통신 기술의 발달로 인터넷을 통한 원격 강의나 이러닝(E-Learning) 형태의 교육이 확산되었다. 이로 인해 교육을 위한 다양한 형태의 웹 클라이언트 기반 소프트웨어들이 개발되었고 사용되고 있다. 원격 강의나 이러닝과 같은 온라인 강의뿐만 아니라 대학에서 이루어지는 오프라인 강의에서도 웹을 통한 강의 또는 평가 시스템은 많이 사용되고 있다[1]. 또한 사이버 대학 등 온라인 강의를 확산되면서 온라인 시험이 요구되고 있다.

이러한 인터넷을 통한 교육은 학습 자료의 검색, 활용뿐만 아니라 실시간으로 성취도를 평가하는 영역으로까지 확대되고 있다. 평가는 객관식 평가와 주관식 평가로 분류하는데 객관식 평가는 채점의 객관성과 신뢰성은 높지만, 단순한 기억력 내지 정보지식의 측정에 치우칠 수 있고 표현력을 측정하기에 부적당하며 창의성을 발휘할 기회가 부족하다는 단점이 있다. 주관식 평가는 자신의 생각을 직접 서술하기 때문에 객관식보다는 분석력과 종합력의 높은 인지능력을 측정할 수 있고, 또 학생들은 다른 어떤 형태의 시험보다도 더 많은 노력을 해야 하기 때문에 교사들이 교육적인 가치가 뛰어난 평가 방법으로 생각하고 있다. 하지만, 주관식 평가는 채점방법에 있어 기준이 모호하고 신뢰도 확보가 어려우며 채점에 많은 시간과 노력이 필요하다[2,3]. 특히 어문학이나 상공계열의 교과목과는 달리 컴퓨터 프로그래밍 관련 교과목에서는 이론 강의 뿐 만아니라 소프트웨어를 작성하는 실습 위주로 강의를 진행한다[1]. 학생들을 평가하는 방법으로는 지필식 시험과 코드를 작성 후 제출하는 평가 방식으로 진행된다. 이와 같이 주관식 평가에 의존도가 높은 과목일수록 문서를 통한 평가보다 과제형 평가에 더 많은 비중을 둔다. 그러나 현재까지 개발된 시스템의 대다수가 객관식 평가로 주로 운용되고 있으며 프로그래밍 언어의 경우 실기형 채점에 그치고 있다.

이에 본 논문에서는 자동 채점에 대해 기존에 사용되고 있거나 논문으로 발표된 시스템을 분석하고 대학 시험에 바로 적용되었을 때의 문제점과 개선해야 될 사항을 살펴보고 있다. 그리고 컴퓨터 프로그래밍 언어의 시험유형을 분석하여 분류하고 프로그래밍 언어 평가에 적합한 시스템을 설계하고 구현하여 실

험하였다. 제안된 자동 채점시스템의 주관식 채점이 객관적으로 잘 처리됨을 보이기 위해 컴퓨터공학 전공 학생 10명을 대상으로 같은 시험을 시험지와 웹에 동일하게 응시하게 한 후, 3명의 교과 담당 교수에게 채점을 의뢰하고 개발한 시스템으로 채점하여 결과를 비교 분석한다.

컴퓨터 프로그래밍 언어 교과목은 주관식 평가에 의존도가 높으나, 객관식에 비해 채점의 신뢰도가 낮으며, 많은 채점 시간이 소요되고 있다. 이러한 문제점을 해결하기 위하여 본 연구에서는 컴퓨터 프로그래밍 언어 교과목의 시험유형을 분석하여 분류하고 자동 채점 시스템을 설계하고 구현하기 위하여 다음과 같은 내용을 다룬다.

첫째, 주관식 평가 채점 알고리즘을 설계하기 위해 프로그래밍 채점 시스템을 분석하고 장단점을 조사한다.

둘째, 컴퓨터 프로그래밍 언어의 시험유형과 특징을 조사한다. 주관식 평가의 종류를 분석하여 본 논문에 적용할 채점 방법에 의하여 유형을 분류한다.

셋째, 주관식 평가 분류를 토대로 자동으로 채점할 수 있는 시스템을 구현한다.

본 논문에서는 자연어 처리 어려움을 해결하기 위해 프로그램 언어 교과목을 정답 유형별로 분류하고 키워드와 불리언 연산을 이용해 자동 채점을 하여 출제자의 주관적 개입을 배제하고 학생들에게 빠르게 채점 결과를 피드백하여 객관성과 신속성을 높이는 데 목적이 있다.

2. 기존의 프로그램 자동 채점 시스템

현재 국내 프로그래밍 경시대회는 그 종류가 다양하고 많다. 그 중 공신력이 있는 ACM-ICPC[4-7]의 PC²시스템이다. ACM-ICPC대회 채점 프로그램은 출제자가 정확하고도 효율적인 프로그램만이 통과할 수 있도록 테스트 데이터를 설계한다. 제한적인 경우나 비효율적인 방법으로 작성된 프로그램들이 제한시간 내에 통과하지 못하도록 다양한 데이터와 크기를 충분히 크게 만든다. 가능한 여러 가지 방법으로 정확하고 효율적인 프로그램만이 통과하도록 설계한다. 그리고 출제자가 직접 자신이 고안한 방법으로 프로그램을 작성하여 다양한 형태의 데이터로 실험해보면서 변수의 범위, 테스트 데이터의 크기 및

제한시간을 설정한다. 경시대회 당일에는 대회 중 각 팀에서 온라인으로 제출된 답을 정답 프로그램의 수행결과와 비교하여 그 결과를 곧바로 통보한다. 각 문제의 출제자가 직접 채점하며 제한 시간 이내에 모든 테스트 데이터에 대해 정답을 출력하는 프로그램만을 통과시킨다.

Neo Test5[8]는 경북권내 대학들의 사이버 강좌에서 사용하고 있는 시험 시스템이다. 문제출제는 문제은행을 풀더 단위로 관리하며, 문항의 종류로는 선택형, 단답형, 서술형으로 나누어지며, 선택형은 다시 단일선택과 복수선택으로 나누어진다. 단답형은 답을 여러 개 추가 할 수 있으며, 공백무시, 답안순서 무시란 등의 세부 설정사항이 있다. 서술형은 긴 문장을 입력할 수 있고 채점 기준 입력란 등이 있다. 모든 문제는 교수가 필요시점에 형성할 수 있으며 문제은행을 미리 구축해 놓고 필요시 선택하여 시험지를 형성할 수 있다. 문항은 모두 시험과 퀴즈로 문제를 구분할 수 있으며 배점을 임의지정 가능하며, 난이도 조정과 주차를 지정할 수 있다. 선택형과 단답형은 자동으로 채점이 이루어지며, 서술형은 출제자가 직접 채점을 해야 한다.

자동화된 프로그래밍 주관식 문제 채점 보조 시스템[9]은 출제자가 문제를 출제하기 위해서는 문제와 상세조건, 컴파일 방법, 인수, 실행방법, 출력결과를 입력해야 하며, 학습자가 답안을 작성하여 웹서버에게 전달하면 웹 서버에서는 이를 프로그래밍 소스로 만들고 이를 출제자가 입력한 컴파일 방법으로 컴파일 하여 오브젝트 파일을 생성한다. 이 오브젝트 파일에 인수를 넣어 출제자가 입력한 실행 방법대로 실행하게 하여 출제자가 입력한 출력 결과와 비교한다. 또한 학습자의 프로그래밍 소스가 출제자가 입력한 상세 조건과 부합하는지 검사하도록 하여 이에 대한 결과를 데이터베이스 서버에 저장하도록 한다. 즉, 출제자가 담당했던 역할을 모두 웹 서버에서 자동으로 실행하도록 하는 시스템이다.

웹 기반 주관식 답분류 채점시스템[10]은 근거리 통신, 인터넷 등의 통신을 이용하여 해당 출제자의 분야별로 출제한 주관식 문제를 모든 응시자가 주관식 문제의 답을 보내고, 해당 답들을 문제별로 답을 분류한 답, 빈도수, 문제, 해설 등의 채점 정보를 출제자에게 알려주면 출제자는 응시자가 쓴 답들의 종류, 각 답들의 분포, 문제의 해설 등을 참고하여 각 답에

대하여 배점을 결정하고 서버는 이 배점으로 채점한 결과를 출제자에게 보여주는 주관식 답분류 채점시스템에 관한 것이다.

WAP(Wireless Application Protocol)환경에서의 주관식 채점 시스템[11]은 WAP 환경에서 주관식 채점시스템을 접목시킨 시스템으로 근거리 통신, 인터넷 등의 통신을 이용하여 응시자가 주제별 주관식 문제에 응시한 후 해당 답들을 주제별로 답을 분류한 채점 정보를 출제자는 출제자의 무선 단말기를 이용하여 응시자가 작성한 답안의 배점을 결정하고 서버는 이 배점으로 채점한 결과를 응시자에게 신속하게 알려 주는 시스템으로서 응시자는 신속한 채점결과를 확인하여 학습능력을 향상 시킬 수 있는 데에 목적을 둔 시스템이다.

일반화된 벡터 공간 모델을 적용한 주관식 문제 채점 보조 시스템[12]은 자연어 처리의 어려움을 해결하기 위하여 문제 공간을 벡터 공간으로 정의하고 벡터를 구성하는 각 자질간의 상관관계를 고려한 방법을 적용한 시스템으로 학습자가 답안을 작성할 때 동의어 사용을 한다는 가정하에 출제자가 여러 개의 모범 답안을 작성하고 이들 답안을 말뭉치에 첨가하여 구성된 다음 형태소 분석기를 통하여 색인을 추출하고 학습자가 작성한 답안 역시 색인을 추출한 다음, 이들 색인들을 각 자질로 정의한 벡터를 구성한다. 이렇게 구성된 벡터들을 이용하여 답안들 간 유사도 측정을 하고, 유사도 범위에 따라 답안을 자동으로 정답과 오답으로 분류하려는 시스템이다.

위 사례들을 분석해 본 결과 프로그래밍 언어 채점은 실기형으로 제한되어 있다. ACM-ICPC[4-7]의 PC²시스템과 자동화된 프로그래밍 주관식 문제 채점 보조 시스템의 경우 입출력 조건과 범위에 대한 부분은 모두 체크되나, 프로그래밍 교과목의 시험 같은 경우 그 시험 범위 내에 주어진 조건의 명령어 사용이나 함수사용여부 등의 세부 사항 등을 체크할 필요가 있다. 정보올림피아드나 ACM은 학생의 프로그램 기량을 겨루는 대회이다. 이를 대학시험에 적용하면 창의력, 독창력을 테스트하고, 실력을 향상시킬 수 있는 반면, 학생들의 의욕상실, 성적관리에 문제가 생길 수 있다. 자동화된 프로그래밍 주관식 문제 채점 보조 시스템의 경우 출제자가 담당해야 했던 복잡한 채점을 웹으로 이관했던 점에서는 좋은 성과였지만 채점할 소스를 텍스트 박스(Text Box)에 입

력하고 컴파일을 웹에서 시도함으로써 학생들의 사소한 문법오류로 인해 0점 처리되는 문제점이 있다. 이 또한 실기형에 해당하는 경우이다. 웹 기반 주관식 답분류 채점시스템과 WAP(Wireless Application Protocol)환경에서의 주관식 채점 시스템은 답을 주제별로 분류하고 빈도수에 따라 출제자가 채점하여 피드백해주는 방식이며 일반화된 벡터 공간 모델을 적용한 주관식 문제 채점 보조 시스템은 자연어 처리의 어려움을 해결하기 위해 형태소 분석기를 통해 색인어를 추출하여 벡터 공간을 형성하고 이를 유사도 측정에 따라 정답과 오답을 분류하는 시스템이다. 이를 프로그램 교과목에 바로 적용하기에는 유사도만으로 논리력을 테스트하기에는 부족함이 있다. 합을 구하는 문장에서 합을 구하는 방법은 반복문을 이용하거나 가우스 이론을 이용해 공식으로 해결하는 경우가 있다. 이런 경우 유사도만으로 판별하기에는 무리가 있다. 또한 주요 명령어 하나가 빠지더라도 나머지 부분이 같다면 유사도는 정답 분류 안에 들어갈 수 있다.

따라서 본 논문에서는 자연어 처리의 어려움을 해결하기 위해 키워드와 불리언 연산을 이용하여 객관식은 물론 주관식 문제의 자동 채점이 가능하도록 하여 채점 결과를 빠르게 피드백해 주고 출제자의 채점 노고를 줄이고 객관성과 신속성을 높인다.

3. 프로그래밍 언어의 자동 채점 시스템

기존의 프로그램 자동 채점 시스템은 실기형으로 제한되어 있으며, 주관식 평가에 관한 연구는 자연어 중심의 어문학이나 상공계열, 초등교과에 초점이 맞추어져 있다. 이에 본 논문은 프로그래밍 언어에 대한 시험유형과 특징을 분석하여 분류하고 프로그래밍 언어에 적합한 시스템을 설계하고 구현한다. 어문학의 경우 자연어 문법과 밀접한 관계가 있기 때문에 자연어 처리를 배제할 수 없다. 그러나 프로그래밍 언어의 특성상 키워드가 존재하고 정확한 문법이 존재한다. 이러한 특징을 이용하여 자연어 처리를 하지 않고도 키워드와 불리언 연산을 이용하여 채점할 수 있는 방법을 제시한다. 문제 출제 유형은 객관식 중 단답형, 객관식 중 복수형, 주관식 중 단답형, 주관식 중 복수형, 서술형, 실기형으로 나누어 출제한다.

본 논문에서는 자연어 처리를 배제하기 위해 서술

형의 문장 구분은 엔터키(Enter key)로만 제한하며 서술형 중 프로그램 코딩에 자유성을 주기 위한 문제는 실기형으로 제한 한다. 프로그램 문제 유형 중 특징이나 비교, 분석을 할 경우 긴 문장이 요구된다. 본 논문에서는 긴 문장 입력 시 한 문장씩 분리하여 엔터키로 쳐서 구분한 문장만을 처리한다. 표 1은 처리할 수 있는 문장 예와 그렇지 않는 경우다.

표 2은 서술형과 실기형을 출제자가 어떻게 구분하여 입력해야 하는지를 보여주는 예이다. 사용합수를 제한하거나 특정부분만을 테스트하고 싶은 경우에는 서술형으로 하며, 답안의 자유성을 보장하고 싶은 경우에는 실기형으로 출제해야 한다.

정답입력은 학생에게 답을 제시하기 위한 정답과 채점을 위한 정답으로 구분하여 입력한다. 채점을 위한 정답은 특수기호와 함께 키워드만을 입력한다. 정답 키워드 조합 유형은 '웹기반 주관식 평가문항 채점 알고리즘 설계 및 구현[3]'의 분류방법에 따라 재 분류하고, 명칭을 프로그래밍 언어에 맞게 재 수정하고 실기형을 추가하였다. 키워드 구성을 중심으로 7가지로 나누며, 실기형을 둔다. 표 3은 키워드를 구분하기 위한 접속사로 사용하는 특수 문자표이며, 표 4는 정답 키워드의 조합에 따른 정답유형별 설명이다.

표 5, 6는 정답 유형별 문제와 학생들에게 제시할

표 1. 문장 구분 입력 예

올바른 입력 방법 (처리 가능한 문장)
첫문자로는 알파벳과 밑줄만을 사용한다. [Enter]
특수 문자는 사용할 수 없다.[Enter]
숫자를 시작 문자로 사용할 수 없다.[Enter]
예약어를 사용할 수 없다.[Enter]
잘못된 입력 방법 (처리 불가능 문장)
첫문자로는 알파벳과 밑줄만을 사용하며, 특수문자는 사용할 수 없으며, 숫자를 시작문자로 사용할 수 없고 예약어를 사용할 수 없다.

표 2. 서술형과 실기형 분류 예

서술형
1~100까지의 합을 for문을 이용하여 작성하시오.
코딩에 제한적
실기형
1~100까지의 합을 구하시오.
자유롭게 반복문 선택이 가능

표 3. 제안한 정답 입력 특수문자

정답 유형	설 명	정답 유형	설 명
#order#	답이 순서적 이어야 함	#not#	절대 포함 되어서는 안 됨
#and#	모두 포함 되어야 함	#(#	(
#end#	문장의 끝)#)
#or#	또는	#?#	변수

표 4. 정답유형

정답 유형	설 명
일치	반드시 1개의 답만 있으며 사용자 입력 답안과 모범 답안이 완전히 일치해야 하는 경우
또는	답은 하나이나 여러 유형으로 나올 수 있는 경우
그리고	여러 개의 답이 요구되나 순서와 무관한 경우 또는 명확한 짧은 문장으로 구성되나 여러개의 핵심 키워드를 포함하고 있는 경우
순서	여러 개의 답이 요구되고 반드시 순서가 맞아야 하는 경우
순서 또는	여러 개의 답이 요구되고 한곳의 답이 여러 유 형으로 나올 수 있으며 반드시 순서가 맞아야 하는 경우
복합	문장 기술을 요구하는 경우로 순서형, 그리고, 또는, 순서또는형, 부정형을 적용하여 처리
부정	문장에 요구하는 답이 모두 포함 되어 있으나 절대 포함 되어서는 안 되는 단어가 있을 경우 오답 처리

표 5. 정답 입력 예시

정답 유형	주관식(복수형) 순서 또는 형
문제1	1~10까지의 합을 for문을 이용하여 작성하시오. void main() { int i, sum; (①) for(②) (③) printf("%d",sum); }
정답	① sum=0; ② i=1;i<=10;i++ ③ sum=sum+i;
키워드	① sum=0 ② #order# i=1;#and# #(# i<=10 #or# i<11 #)# #and# #(#i+1 #or# i++ #or# i+=1 #)# ③ sum+=i #or# sum=sum+i
논리식 문자열	② i=1 && (i<=10 i<11) && (i+=1 or i++ or i+=1) 1 && (1 0) && (1 0 0)

표 6. 서술형 입력 예시

정답 유형	서술형 (복합형)
문제2	변수명 규칙에 대해 설명하시오.
정답	첫문자로는 영문자, 밑줄만 사용가능 첫문자로 숫자를 사용할 수 없다. 특수문자를 사용할 수 없다. 키워드(예약어)를 사용할 수 없다.
키워드	#(#첫#or#시작#or#처음#or##)#and##(영#or# 알파벳#or#소#or#대#or#밑줄#)#and##not## ##not#없#and#불#)#end##(#첫#or#시작#or# 처음#or##)# #and#숫자#and##(없##or 불#)#)#end# 특수#and##(없#or#불#)#)#end# #(#키워드#or#예약어#)#and##(없#or#불#)#)#end#
논리식 문자열 벡터	(첫 시작 처음 앞)&&(영 소 대 알파벳 밑줄)&&(!없&&!불) (첫 시작 처음 앞)&&숫자&&(없 불) 특수&&(없 불) (키워드 예약어)&&(없 불)

정답과 채점을 위한 키워드입력과 처리과정을 거친
후에 얻어 질 논리식 문자열 예이다.

4. 구 현

시스템의 구현 환경으로는 WindowsXP에서 JDK
버전은 1.5.0.08이며, 웹서버로는 Apache Tomcat 5.5
를 사용하였고 데이터베이스는 Mysql 5.0.67을 사용
하고 개발 툴로는 Eclipse, Editplus, SQL Gate을 사
용하였다.

4.1 제안된 프로그래밍 언어 자동 채점 시스템

프로그래밍 언어 자동 채점 시스템은 출제자가 문
제를 출제하기 위해서는 문제와 정답, 키워드, 배점,
문항유형을 입력하도록 하고 응시자는 답안을 작성
하여 웹서버에 등록만 하면 웹서버는 이를 가공하여
채점 후 그 결과를 데이터베이스에 기록한다. 출제자
는 정답과 학생들의 결과를 검색할 수 있으며 학생들
답안도 볼 수 있도록 설계하였다. 채점에 관한 모두
사항이 웹서버에서 자동으로 실행된다. 그림 1은 본
시스템을 구현하기 위한 유스 케이스(Use Case)이
며, 그림 2은 시스템 구성도이다.

그림 3은 채점 흐름도이다. 출제자가 문제를 출제
하고 학생이 시험에 응시를 하면 채점시스템은 여러

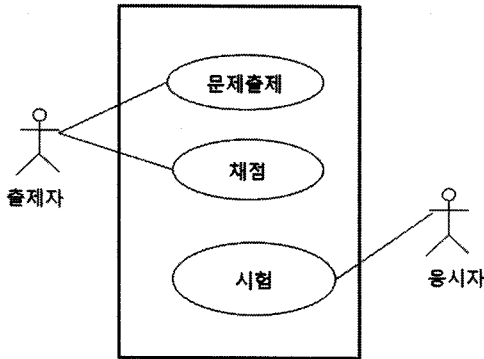


그림 1. 유스 케이스(Use Case)

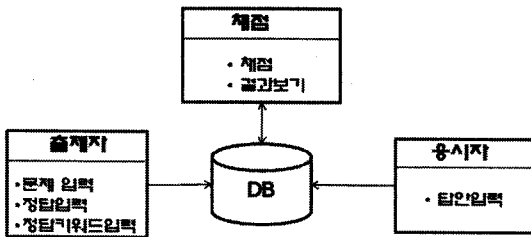


그림 2. 시스템 구성도

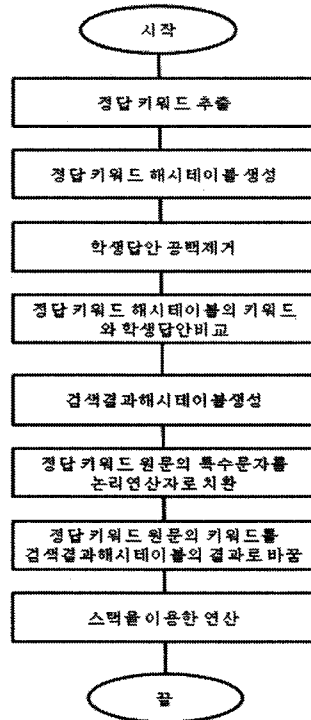


그림 3. 채점 흐름도

단계의 가공을 거치게 된다. 우선 1차 가공에서 키워드를 분리해내고 키 및 키워드로 이루어진 해시테이블을 생성한다. 그 다음은 학생답안에서 공백을 제거해 1차 가공을 한 후 키워드와 비교해 0과 1로 이루어진 검색 결과 해시테이블을 생성한다. 키워드 원문의 특수문자를 논리연산자로 치환 후, 키워드 원문의 키워드를 검색 결과 해시테이블 값으로 바꾸어 논리식 문자열을 얻는다. 이를 연산하여 채점결과를 얻어낸다. 이렇게 얻어낸 결과는 데이터베이스에 저장되고 출제자가 시험결과를 요청하면 정답, 결과, 학생답안을 볼 수 있다.

4.2 채점 알고리즘

출제자가 문제를 출제하고 정답과 키워드를 입력하면 채점 시스템은 아래와 같은 단계를 거쳐 채점을 한다.

단계 1. 키워드에서 키워드만을 빠르게 추출하기 위해서 특수문자를 구분기호로 바꾼다.

단계 2. 구분기호로 키워드를 분리해 해시테이블을 생성한다.

단계 3. 학생답안에서 공백을 제거한다. 공백을 제거하여 하나의 문자열로 구성한다. 공백으로 인해 여

러 단어로 분리되는 것을 막기 위한 것이다. 공백을 제거하더라도 키워드를 이용해 찾기 때문에 새로운 단어의 형성 문제는 고려하지 않아도 된다.

단계 4. 학생답안을 키워드와 비교하여 검색 결과 해시테이블을 생성한다.

단계 5. 키워드 원문의 특수 문자를 논리연산자로 바꾼다.

단계 6. 키워드 원문의 키워드를 검색 결과 해시테이블의 값으로 바꾸어 논리식 문자열을 얻는다.

단계 7. 스택을 이용한 연산으로 채점 결과를 얻는다.

그림 4는 채점 전체과정의 예이다.

정답	: 첫 문자로 알파벳 또는 밑줄만 사용
키워드	: 첫 문자 #and# (# 알파벳 #or#영어 #or# 소 #or# 대 #or# 밑줄 #or# _ #)# and# #not# 없
해석	: 첫 문자를 포함하고 알파벳, 영어, 소대문자, 밑줄 '_' 중 하나를 포함하고 없다가 포함되어서는 안 된다.
학생답안	: 첫 문자로 영어소대문자와 밑줄을 사용할 수 있다.
비교결과	: 1 && (0 1 1 1 1 0)&& !0
채점결과	: 1 로 정답임.

그림 4. 예시

단계1. 정답 키워드를 가공하여 그림 5와 같이 해시 테이블을 생성한다.

단계2. 학생답변을 그림 6와 같이 가공 후 키워드와 비교하여 검색 결과 해시 테이블을 생성한다.

단계3. 그림 7과 같은 과정을 통해 채점결과를 얻는다.

5. 실험 및 고찰

제안된 자동 채점시스템의 주관식 채점이 객관적으로 잘 처리됨을 보이기 위해서 실험을 하였다. 실험은 컴퓨터공학전공 2, 3학년생 10명을 대상으로 24문항의 동일한 시험을 웹과 시험지에 똑같이 답을 입력하고 3명의 강의 담당 교수가 채점한 결과와 제안된 자동 시스템 채점 결과표이다. 한 학생당 24문

```

1차 가공 : 키워드 원문
예) #(#동해#and#물과#or# 백두#)# #and# 산이 #and#
#not# 마르고
#and# ==> && , #or# ==> ||, #not# ==> ! ,
#( # ==>(, #)#==> )
위와 같이 특수문자를 논리연산자로 치환
가공된 키워드 문장 :
(동해&&물과||백두)&&산이&& !마르고
2차 가공 : 키워드 해시와 검색 결과 해시를 해당
글자 대신에 1,0 으로 바꾼다.
가공된 키워드 문장 :
( 1 && 1 || 0) &&1 && !0
3차 가공 : 스택을 이용한 연산으로 채점 결과를 얻
는다.
    
```

그림 7. 채점과정

항에 대해 시스템과 세 명의 담당교수가 채점한 결과이다. 수치는 각 문항에 대한 배점이다. 채점결과 객관식 중 단답형, 객관식 중 복수형, 주관식 중 단답형, 주관식 중 복수형, 서술형은 채점자와 시스템이 거의 일치하는 결과를 보였으며 서술형은 표 7와 같다.

표 8은 24문항 중 서술형 문항의 문제번호이며 채점결과를 시스템과 동일한 결과가 나온 개수와 그렇지 않은 경우를 수로 나타낸 표이다.

문제 9번은 'switch문을 이용하여 A이면 사과, B이면 바나나, 기타는 기타를 출력하는 프로그램을 완성하시오.'라는 문제에서 조건이 'switch문을 사용하라'고 했음에도 불구하고 if문을 사용해 답안을 작성한 경우와 switch문 문법이 틀린 경우에도 채점자들은 부분점수를 주었고 시스템은 0점 처리 한 경우의 오차였다. 21번은 배점이 높고 '1부터 100까지 3만큼 씩 세는 while문을 작성하라.(첨자는 i를 사용)'라는 프로그램 작성문제였는데 시스템은 정확한 코딩에 만 점수를 주었고 채점자는 일부 로직이 맞으면 부분 점수를 주었다. 14번은 '최대공약수를 구하는 함수를 완성하시오. 조건: 함수이름 GCD, 정수두개를 매개 변수로 받는다. 최대공약수를 반환한다. 되부름을 이용한다.' 라는 문제로 재귀호출을 이용해 함수를 완성하라는 문제난이도가 높은 경우였는데 10번 학생을 제외한 모든 학생이 정확한 답을 적지 못했는데도 불구하고 채점자중 한 분이 부분 점수를 주고 있다. 또한 19번은 '배열 data[]의 첫 번째 요소의 주소를 얻는 두 가지 방법을 쓰시오.' 라는 문제로 두 가지 조건을 다 적어야 하는 문제였는데 시스템은 둘 다 있어야 점수를 주었고 채점자는 부분점수를 주었다.

```

현재 키워드 : #(#동해#and#물과#or# 백두#)# #and#
산이 #and# #not# 마르고
1 차가공 : #dvd#동해#dvd#물과#dvd# 백두#dvd#
#dvd#산이 #dvd##dvd# 마르고
2 차가공 : String keywords[]=
keywordTemp[k].split("#dvd#"); //
답변 목록
-> keywords[] = {"동해", "물과", "백두", "", "산이",
"", "마르고"};
3차 가공 : keywordChkHash 테이블 생성

동해, 물과, 백두, 산이, 마르고
1 2 3 4 5
(키, 키워드)
(1, "동해")
(2, "물과")
(3, "백두")
(4, "산이")
(5, "마르고")
    
```

그림 5. 키워드 해시 테이블 생성

```

1차 가공 : 답변에서 공백제거
2차 가공 :
correctChk.put(count,(answer.indexOf(cfKey-
word)>-1?1 : 0)+""");
answer.indexOf(cfKeyword)>-1 ? 1 : 0
키워드가 들어 있으면 1 없으면 0 즉, 해당 키워드가
있으면 아래와 같이 검색 결과 해시 맵 구성
1, "0"
2, "0"
3, "1"
4, "1"
5, "1"
    
```

그림 6. 학생답안 가공과정

표 7. 채점 결과표

시험형 문제번호	1	9	11	12	14	15	17	19	21	24
학생id	배점	3	3	3	3	3	3	3	3	3
system	2	3	3	3	3	3	3	3	3	3
A	2	3	3	3	3	3	3	3	3	3
B	2	3	3	3	3	3	3	3	3	3
C	2	2	3	3	3	3	3	3	3	3
std1	system	3	3	3	3	3	3	3	3	3
A	3	3	3	3	3	3	3	3	3	3
B	3	3	3	3	3	3	3	3	3	3
C	3	3	3	3	3	3	3	3	3	3
std2	system	3	3	3	3	3	3	3	3	3
A	3	3	3	3	3	3	3	3	3	3
B	3	3	3	3	3	3	3	3	3	3
C	3	3	3	3	3	3	3	3	3	3
std3	system	3	3	3	3	3	3	3	3	3
A	3	3	3	3	3	3	3	3	3	3
B	3	3	3	3	3	3	3	3	3	3
C	3	3	3	3	3	3	3	3	3	3
std4	system	3	3	3	3	3	3	3	3	3
A	3	3	3	3	3	3	3	3	3	3
B	3	3	3	3	3	3	3	3	3	3
C	3	3	3	3	3	3	3	3	3	3
std5	system	3	3	3	3	3	3	3	3	3
A	3	3	3	3	3	3	3	3	3	3
B	3	3	3	3	3	3	3	3	3	3
C	3	3	3	3	3	3	3	3	3	3

표 8. 시스템 결과와 불일치 통계표

문제번호 일치여부	1	9	11	12	14	15	17	19	21	24
일치	30	25	24	12	26	29	8	22	19	30
불일치	0	5	6	18	4	1	22	8	11	0

11번, 12번, 17번은 'while문과 do-while문의 차이점을 기술하시오.'와 같은 유형의 문제로 다른 교과목과 같이 설명을 요하는 문제였다. 이 경우 시스템은 예상치 못한 미등록 동의어 사용으로 0점 처리 되는 경우가 발생했다. 24번은 두 수를 매개변수로 받아 합을 리턴 하는 함수구현 부분 이었는데 여러 가지 방법으로 모두 답을 제시했고 분석결과 잘 채점되었다. 위의 결과를 요인별 분석해 본 결과 부분점수로 인한 불일치 수가 29건, 동의어 미등록으로 인한 불일치 건수가 46건으로 나타났다.

표 9는 응시자별 채점 결과표이다. 시스템이 잘못 채점한 경우를 분석해 본 결과, 총 240건을 채점했으

표 9. 응시자별 채점 결과

응시자 수 : 10 명	1번	2번	3번	4번	5번	6번	7번	8번	9번	10번	11번	12번	13번	14번	15번	16번	17번	18번	19번	20번	21번	22번	23번	24번	총점
std1 (+)	2.0	3.0	3.0	3.0	0.0	3.0	3.0	3.0	0.0	3.0	3.0	3.0	0.0	0.0	0.0	3.0	0.0	2.0	5.0	2.0	0.0	5.0	2.0	20.0	68.0/98
std2 (+)	3.0	3.0	2.0	3.0	0.0	3.0	3.0	0.0	0.0	3.0	3.0	3.0	0.0	0.0	3.0	3.0	0.0	2.0	5.0	1.0	0.0	5.0	2.0	20.0	67.0/98
std3 (+)	3.0	3.0	2.0	3.0	0.0	0.0	3.0	3.0	0.0	3.0	0.0	1.0	3.0	0.0	0.0	3.0	3.0	2.0	5.0	3.0	10.0	5.0	2.0	20.0	77.0/98
std4 (+)	3.0	3.0	2.0	3.0	0.0	0.0	3.0	3.0	3.0	3.0	0.0	3.0	3.0	0.0	3.0	3.0	0.0	2.0	5.0	1.0	10.0	5.0	2.0	20.0	80.0/98
std5 (+)	3.0	3.0	1.0	1.0	0.0	3.0	3.0	0.0	3.0	3.0	3.0	1.0	3.0	0.0	3.0	0.0	0.0	2.0	5.0	2.0	0.0	5.0	2.0	20.0	66.0/98
std6 (+)	3.0	3.0	2.0	3.0	0.0	0.0	3.0	3.0	0.0	1.0	3.0	1.0	3.0	0.0	0.0	3.0	0.0	2.0	5.0	0.0	0.0	5.0	0.0	20.0	60.0/98
std7 (+)	3.0	3.0	3.0	1.0	0.0	3.0	0.0	3.0	3.0	3.0	0.0	1.0	3.0	0.0	0.0	3.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0	20.0	51.0/98
std8 (+)	2.0	3.0	2.0	3.0	0.0	0.0	3.0	3.0	3.0	3.0	3.0	1.0	3.0	0.0	0.0	3.0	0.0	2.0	5.0	1.0	0.0	5.0	0.0	20.0	65.0/98
std9 (+)	2.0	3.0	3.0	3.0	0.0	0.0	3.0	3.0	3.0	2.0	0.0	1.0	3.0	0.0	0.0	0.0	0.0	2.0	5.0	2.0	0.0	5.0	2.0	20.0	62.0/98
std10 (+)	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	0.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	2.0	5.0	3.0	10.0	5.0	2.0	20.0	95.0/98

며 그 중 정답인데도 오답 처리한 건수가 20건, 틀린 데 정답처리를 한 경우가 0건으로 91.7% 정확도를 보였다. 오답처리 한 경우는 '무엇을 설명하시오.' '비교하시오' 유형의 문제로 문장기술을 요구하는 문제였고 미등록 동의어 사용과 문장 구분의 문제였다. 이 부분은 자연어 처리가 필요한 부분이며, '문장 구분을 엔터로 하라' 라는 지시문을 넣어 줌으로써 채점의 정확도를 높일 수 있다. 프로그래밍 관련 문제는 잘 채점되었음을 보여 준다.

주관식 채점 시 주관적인 개입을 배제할 수 없으며 채점자별 배점 기준이 달라 시스템이 상대적으로 객관성이 있다고 말할 수 있다. 제안된 방법으로 키워드와 불리언 연산을 이용해 채점이 가능함을 알 수 있다.

제안된 시스템은 대형 강의에서 여러 명의 조교가 채점을 할 경우 매우 유용하게 활용될 수 있다. 여러 채점자들이 채점할 경우 동일한 답안에 대해서도 점수 편차가 심할 수 있으나 제안된 방법은 객관성이 보장된다.

6. 결 론

프로그래밍 언어는 정확한 문법 체제와 키워드가 존재한다. 이러한 특징을 이용하여 본 논문은 키워드와 불리언 연산을 이용한 프로그래밍 언어 자동 채점 시스템을 설계하고 구현하여 실험하였다. 프로그래밍 언어를 정답 유형별로 7가지로 나누고 키워드 입력 시 접속사를 특수문자를 이용해 입력하게 하여 논리식을 바로 만들어 낼 수 있었으며 스택의 구현으

로 쉽게 연산이 가능했다. 그리고 기존의 실기형 채점에 주관식 채점을 추가함으로써 프로그래밍 교과목 채점을 자동화 할 수 있다. 주관식 채점에서 활용되는 방대한 자연어 처리를 배제할 수 있도록 키워드와 불리언 연산을 이용한 방식을 제안하므로 구현의 효율성이 높아졌다. 또한 채점을 웹으로 이관함으로써 학생들에게 결과를 빠르게 피드백해주고 채점자의 주관적 개입을 배제하여 객관성, 공정성, 신속성을 높였다.

서술형은 출제자의 답안 입력 시 선택한 용어에 의해 채점 결과가 많이 좌우된다. 동의어 사전의 구축으로 출제자는 평소에 사용하는 한 단어만을 입력하더라도 사전에 의해 동의어 사용도 채점이 가능해져야 한다.

추후 연구 과제로는 한글의 특성상 여러 가지 동의어가 활용될 수 있으므로 동의어 사전의 구축이 필요하다.

참 고 문 헌

[1] 장정일, "스마트 클라이언트를 적용한 프로그래밍 과제평가 시스템 설계 및 구현," *한국정보처리학회*, 제14권, 제1호, PP. 1380-1382, 2007.

[2] 박희정, "주관식출제·채점모델 설계 및 구현," *안동대학교 컴퓨터교육학 석사논문*, 2003.

[3] 권오영, "웹 기반 주관식 평가문항 채점 알고리즘 설계 및 구현," *한서대학교 교육학과 전자계산교육전공 석사논문*, 2004.

[4] 정현우, "경시대회 참가 규칙 및 진행방식," *한국정보과학회지*, 제25권, 제7호, PP.71-74, 2007

[5] 김재훈, 조신행, "세계대회를 향한 첫 걸음: ACM-ICPC 서울 지역대회 인터넷 예선," *한국정보과학회지*, 제25권, 제7호, PP. 67-70, 2007.

[6] 임형석, "ACM-ICPC 문제의 출제 및 채점 과

정," *한국정보과학회지*, 제25권, 제7호, PP. 52-55, 2007.

[7] ACM-ICPC, <http://acm.kaist.ac.kr/>

[8] NeoTest, <http://neotest.com/>

[9] 오정석 외 3인, "자동화된 프로그래밍 주관식 문제 채점 보조 시스템," *한국정보처리학회 추계학술발표대회 논문집*, 제10권, 제2호, PP. 253-256, 2003.

[10] 방훈 외 4명, "웹 기반 주관식 답분류 채점시스템," *한국정보과학회 가을 학술발표논문집*, Vol. 28, No. 2, PP. 589-591, 2001.

[11] 방훈 외 4명, "WAP 환경에서의 주관식 채점 시스템," *한국정보과학회 봄 학술발표논문집*, Vol. 29, No. 1, PP. 712-714, 2002.

[12] 방훈 외 4명, "WAP 환경에서의 주관식 채점 시스템," *한국정보과학회 봄 학술발표논문집*, Vol. 29, No. 1, PP. 712-714, 2002.

[13] 오정석 외 3명, "일반화된 벡터 공간 모델을 적용한 주관식 문제 채점 보조 시스템," *한국정보처리학회 춘계학술발표대회 논문집*, 제11권, 제1호, 2004.



정은미

2006년 2월 안동대학교 컴퓨터공학과 공학사
 2009년 2월 안동대학교 컴퓨터공학과 공학석사
 2009년 3월~현재 안동대학교 정보통신 공학과 외래강사

관심분야 : 컴퓨터비전, 온라인 자동 채점, 영상처리, 자연어 처리



최 미 순

1991년 2월 안동대학교 전산통계
학과 졸업
1998년 2월 안동대학교 대학원
졸업(공학석사)
2004년 2월 안동대학교 대학원
박사수료
1991년 11월~1994년 12월 (주)
포스코 FA프로그램 개발

1998년~현재 안동대학교 시간강사

관심분야 : Image Processing, Patten Recognition,
Computer Vision, Biometrics



심 재 창

1994년~현재 안동대학교 교수,
인력개발본부장
1998년~현재 (주)과미 감사
2005년~2006년 Princeton Uni-
versity Visiting Fellow
1997년~1999년 IBM T. J.
Watson Research Center,
Researcher

1994년 경북대학교 공학박사

1990년 경북대학교 공학석사

1987년 경북대학교 전자공학과 공학사

관심분야 : 영상처리, 패턴인식, 컴퓨터비전, 임베디드 비
전시스템, 비전 SoC, 카메라 센서네트워크