

# 학부 문제해결기법 교과목의 운영 사례

부산외국어대학교 | 김수환\*  
가톨릭대학교 | 박정흠\*  
한국과학기술원 | 좌경룡\*

## 1. 서론

컴퓨터 과학/공학 전공의 학부 학생들 대부분은 1학년 과정에 개설된 기초 프로그래밍 관련 교과목을 통해서 컴퓨터 프로그래밍의 세계에 입문한다. 기초 프로그래밍 과목의 교육 목표는 C 또는 Java와 같은 특정 프로그래밍 언어를 선택하여 문법 중심의 학습을 통한 프로그래밍 원리를 익히는 것이 보통이다. 이후 학생들은 자료구조와 소프트웨어공학적인 프로그래밍 기법을 익혀 응용 프로그래밍의 세계로 뛰어 들게 된다. 이 과정 중에 상당수의 학생들이 프로그래밍에 관한 흥미를 잃어 비프로그래밍 분야로 진로를 바꾸어 버리기도 한다. 그렇지 않은 학생들의 상당수도 한국 정보올림피아드(KOI)의 중·고등부 심지어 초등부 문제조차도 제대로 풀지 못하는 것을 발견할 수 있다. 이는 정형화된 틀 안에서 프로그래밍 작업을 수행하는 것은 가능하지만, 새로운 문제를 창의적으로 해결하는 능력이 부족하기 때문이다.

최근에 주어진 문제에 대해 깊이 생각하여 스스로 문제를 해결하는 방식을 찾아내고 적절한 자료구조를 선택하여 좋은 프로그램을 설계할 수 있는 능력의 중요성이 증대되고 있다. 이에 따라 학생들의 문제 해결 능력을 평가하는 ACM-ICPC 경시대회에 대해 대학과 학생들의 관심이 점점 커져가고 있다. 이는 해마다 이 대회에 참여하는 대학과 학생의 수가 늘어가고 있다는 사실을 통해서도 알 수 있다[1].

대학의 교육과정에서도 학생들의 문제해결 능력 향상을 위한 교육의 필요성이 요구되고 있다. 프로그래밍 언어의 문법과 기초 자료구조 지식을 바탕으로 문제 해결 능력을 배양하는 교육을 담당하는 교과목은 보통 「문제해결기법」이라는 명칭으로 불린다. 문제해결기법 교과목은 프로그래밍 활용 능력을 배양하고

스스로 문제를 풀 수 있다는 자신감과 성취감을 불러일으킬 수 있어, 우수한 일부 학생뿐만 아니라 대다수 학생들에 대해서도 매우 효과적이며 프로그래밍에 대한 그들의 관심과 흥미를 다시 유발시키는 매개체가 된다.

본고에서는 문제해결기법 교과목을 운영하고 있는 국내외 대학들의 운영 사례 연구를 통하여 이 교과목의 성격을 정리해보고자 한다. 교과목 운영에 따라 발생할 수 있는 여러 가지 문제점을 예견하도록 함으로써, 교과목 도입을 검토하고 있는 학과와 담당교수에게 도움이 되기를 기대한다.

본고의 구성은 다음과 같다. 2절에서는 문제해결기법 교과목의 수업 모델을 설계할 수 있는 기초를 마련할 수 있도록, 국내외 여러 대학에서 개설된 강좌의 강의 방법 및 강의 내용을 고찰하여 교과목의 공통적인 특성을 추출하고 정리한다. 3절에서는 한국과학기술원 전자전산학부 전산학전공과 가톨릭대학교 컴퓨터정보공학부, 부산외국어대학교 컴퓨터공학부의 운영 사례를 각각 구체적으로 살펴본다. 4절에서는 결론을 기술한다.

## 2. 문제해결기법 교과목 수업 모델

프로그래밍 입문 성격의 과목과 차별되는 문제해결기법 강좌는 대체로 자료구조론을 수강한 학생을 대상으로 개설된다. 국내의 한국과학기술원 전자전산학부와 가톨릭대학교 컴퓨터정보공학부, 부산외국어대학교 컴퓨터공학부의 경우는 2학년 봄학기에 자료구조론을 개설하고, 2학년 가을학기에 문제해결기법을 개설한다[2-4]. 미국의 Cornell 대학과 캐나다의 British Columbia 대학에서는 각각 3학년 가을학기 와 4학년 가을학기에 「Creative Problem Solving in Computer Science」와 「Problem Solving in Computer Science」명칭의 강좌를 개설한다[5,6].

\* 종신회원

미국의 Duke 대학은 좀 특이하게 반복 수강이 가능한 「Problem Solving Seminar」 과목을 봄/가을 학기 모두 개설한다[7]. Duke 대학은 ACM-ICPC 대회에 참여할 팀을 육성할 목표로 이 과목을 운영하고 있으며, 지금까지 총 14회 개최된 세계대회에 13번이나 팀을 참여시키는 성과를 보이고 있다.

이들 대학에서 개설하고 있는 강좌의 공통적인 특징은 다음과 같다.

#### 가. 교재와 참고문헌

특정한 교재나 참고서를 필요로 하지 않는다. 다만, 다루어지는 문제의 분야와 난이도에 따라 학생들이 자료구조 또는 알고리즘 관련 참고문헌을 필요로 할 경우가 있다.

#### 나. 선수과목

공식적으로 선수과목을 지정하지 않지만, 기초 자료구조 및 알고리즘 관련 교과목이 먼저 개설되기 때문에, 학생들이 기본적인 자료구조에 관한 지식을 가지고 있다고 가정한다.

#### 다. 강좌 운영

대학마다 약간의 차이가 있지만, 문제 풀이 및 토론식 운영이 주가 된다. 대학에 따라서는 강의 시간의 일부분을 프로그래밍 기법과 알고리즘에 관한 이론적인 내용을 강의하는데 사용한다.

학생들은 이번 주에 출제된 프로그래밍 문제를 과제로 하여 다음 주 강의 시간 이전까지 결과물을 제출해야 한다. 결과물로서 프로그램 소스뿐만 아니라 문제를 해결하는데 사용한 자신의 방법을 정확하게 기술한 보고서를 제출해야 한다. 이 과제는 성적 평가의 큰 잣대가 된다.

강의 시간에는 과제로 제출된 문제에 대해 학생들의 토론 수업이 진행된다. 담당 교수는 문제의 정확한 해를 직접 설명하지 않는다. 다만 학생들이 토의를 통해 문제에 대한 여러 가지 해법을 도출하고 학생 스스로 결론을 짓도록 유도한다. 교수의 역할은 토론 진행자이며 필요시 약간의 힌트를 제공하는 것으로 한정된다. 문제에 대한 토론이 완료되면 교수는 전체적인 총평과 약간의 강의로서 수업을 마무리한다.

#### 라. 강의 내용

이론적 강의를 병행하는 대학의 경우, 기초적인 자료구조 및 기본적인 알고리즘 설계에 관한 내용을 따로 시간을 할애하여 소개한다. 또한 순환, 라이브러리 함수 활용 등 프로그래밍 기법에 관한 내용과 프로그램 개발 과정에 필요한 설계 및 구현, 테스트, 문서 작성 등에 관한 내용도 소개한다.

수업 시간에 다루는 문제는 수강생의 수준에 따라 난이도에 차이가 나겠지만 문제의 주제는 대체로 다음과 같다.

- 문자열 처리
- 정렬과 탐색
- 대수와 정수론
- 그래프 이론
- 계산기하학

문제들은 담당 교수가 직접 만들기도 하고, ACM-ICPC 및 KOI 대회를 비롯한 각종 프로그래밍 경진대회의 기출 문제를 활용하기도 한다. 다루어지는 문제들 중에는 동적계획법이나 퇴각검색법과 같은 알고리즘 설계 기법의 적용이나 복잡한 그래프 알고리즘이나 기하 알고리즘을 필요로 하는 문제들도 있다. Duke 대학은 세계 각지의 ACM-ICPC 지역대회 문제들만을 대상으로 하며, 올해 봄학기에는 2001년도 대전에서 개최된 아시아 지역대회 문제를 사용하기도 하였다.

#### 마. 성적 평가

성적 평가는 매우 다양하다. 출석 및 제출된 과제에 대한 평가와 더불어 토론 수업에 대한 참여도, 지난 수업의 해법이 틀림을 증명하는 것, 흥미있는 새로운 문제를 제시하는 것 등도 점수에 가산되기도 한다. 한국과학기술원, Cornell 대학, Duke 대학은 중간고사와 기말고사를 실시하지 않는다. 가톨릭대학은 중간고사와 기말고사를 실시 시험으로 실시하고, 부산외대와 British Columbia 대학은 필기시험을 실시한다.

### 3. 운영사례

본 절에서는 국내의 3개 대학에서 운영하는 문제 해결기법 교과목의 운영 사례를 좀 더 상세하게 살펴본다.

#### 3.1 한국과학기술원 전산학과

한국과학기술원 전산학과에서는 1992년부터 문제해결기법 과목을 개설해 왔다. 현재는 학부 2학년 가을 학기에 교과목을 개설하고, 수강생은 최근 3년간 30~40명 정도이다.

강좌는 1주일에 2시간 강의와 3시간 실습으로 운영된다. 강의는 1주일에 한 번, 2시간에 걸쳐 토론식으로 진행되고, 수업시간에 다룰 모든 문제들은 미리 과제로 주어진다. 과제로 주어진 토론 주제는 수업이 시작되기 전에 채점을 마쳐서 학생들이 토론이 진행되는 중에 참고할 수 있도록 한다.

중간고사 이전까지는 시간복잡도 개념을 익힐 수

있는 고전적인 알고리즘과 ACM-ICPC 인터넷 예선 문제 등 다양한 문제를 다룬다. 중간고사 이후에는 그래프 이론, 계산기하학 관련 문제들과 ACM-ICPC 서울 대회 기출문제 등 다양한 문제를 다루고, 학기말 프로젝트를 수행한다.

이 과목에서는 정확한 답을 내는 것이 아닌, 어떠한 접근방식을 얼마나 논리적으로 표현하는지를 중요하게 본다. 주어진 문제에 대한 매우 효율적인 방법을 서술했다고 하더라도, 정확히 표현하지 못하면 좋은 점수를 받을 수 없다. 반면, 약간 비효율적인 답이더라도 다른 학생들이 생각하지 못한 독특한 방법을 논리적으로 설명했다면 높은 점수를 받을 수도 있다.

주어진 문제들은 문헌상에 발표된 논문들의 내용을 단순화시킨 것이다. 학생들은 나름대로 생각한 방법을 사용하여 문제를 풀고, 수업시간에 이에 대해 토론하게 된다. 학생들은 이 과정에서 단순한 문제를 어떻게 해결하고, 나아가서는 이를 어떻게 좀 더 일반화된 문제로 발전시킬 수 있는지, 이 과정에서 또 어떤 어려움이 있을지를 생각하게 된다. 담당교수는 토론 진행자 역할을 하며, 토론이 끝나면 담당교수의 총평으로 한 문제에 대한 강의가 끝난다.

이 강의의 목적은 정형화된 방법의 나열이 아닌, 주어진 문제를 해결하고 그 해결 방법을 토론하는 과정에서 논리적 사고력을 배양하고, 단순한 문제를 일반화시키는 연구 과정을 체험하는 과정에서 주어진 문제를 스스로 해결하는 창의적 사고능력을 키우는 것이다. 또한, 학생들이 문제를 해결하기 위해 사용한 방법들이 실제로 그 문제를 연구한 연구자들이 사용했던 문제 해결 방법, 실제 연구 결과와 어떤 관계를 가지며, 문제들이 실생활의 어떤 문제들과 연계될 수 있는지 체험하게 된다.

이 강좌에는 모두 5명의 조교를 활용한다. 1명은 총괄 조교이고 4명은 채점 조교이다. 학부때 이 강좌를 수강한 경험이 있는 석사과정 학생들을 우선적으로 배정한다.

### 3.2 가톨릭대학교 컴퓨터정보공학부

가톨릭대학교 컴퓨터정보공학부에서는 1999년부터 학부 2학년 가을학기에 문제해결기법 과목을 개설하고 있다.

매주 3시간의 수업에서 1시간은 기초적인 문제 해결 기법과 프로그래밍에 관하여 강의식 수업을 실시하고, 2시간은 담당교수가 출제한 프로그래밍 문제에 대한 토론을 중심으로 토론식 수업을 진행한다. 프로그래밍 문제는 중간고사와 기말고사 기간을 제외하고 매주 출

제되어 한 학기에 13~14 문제를 다룬다. 프로그래밍 과제는 매주 토론 시간에 학생들에게 주어지며, 그 다음 주 토론이 시작되는 시각에 과제 점수를 마감하며 마감시간을 넘긴 과제는 점수하지 않는다.

매주 1시간 동안 진행되는 강의식 수업에서는 순환, 기초적인 자료구조의 응용, 기초적인 알고리즘 설계 기법, 프로그래밍 기법 등을 포괄적으로 다룬다. 필요에 따라 가장 최근에 다룬 토론 문제의 변형에 대한 해결 방법을 다루기도 한다. 또한 출제될 예정인 문제에 대한 기초 지식을 다루기도 한다. 예를 들면, 기하 문제를 출제하려고 준비하는 경우, 기초적인 기하 연산을 미리 강의하는 것이다.

지난주에 출제된 프로그래밍 문제에 대하여, 학생들이 이 문제에 대한 과제를 이미 제출한 상태에서 토론 수업이 시작된다. 대략 2~5명 정도의 학생을 지명하여 본인이 사용한 알고리즘을 소개하도록 한다. 지명된 학생들의 발표가 끝나면, 자원하는 학생들의 발표시간을 갖는다. 이때에는 이미 발표된 것보다 독특한 해결방식이나 구현 방식을 발표한다. 토론 수업의 마지막 부분에서는 담당교수가 약 20분 정도 시간을 할애하여 이미 발표된 여러 해결방식을 정리하고, 각 해결방식에 대한 코멘트, 그리고 학생들의 것과 차이가 있을 경우 담당 교수의 해결방식을 소개한다.

중간고사와 기말고사도 프로그래밍 과제와 같은 형식으로 문제를 출제하여 프로그램을 작성하여 제출하도록 하고 있다. 학생들이 주어진 문제에 대하여 차분하게 생각하고 정리한 다음 프로그램을 작성할 수 있도록, 시험 일시는 보통 다른 교과목의 시험이 끝난 후로 정하며 시험 시간도 여유있게 약 6~7시간 정도로 정한다. 시험에서는 다른 학생들과의 토론을 금지하고 있다. 입력의 크기와 난이도가 다양하도록 10개의 테스트 데이터를 작성하여 채점한다. 시험에서는 10개의 데이터 모두를 통과하였을 때 100을 기준으로 하나씩 틀릴 때마다 남은 점수의 1/3씩 감점한다. 예를 들면 10개중 하나가 틀리면 67, 둘이 틀리면 44점이 된다. 학생들이 제출하면 즉시 그 자리에서 채점하여 성적을 확인한다. 물론 시험 문제와 테스트 데이터는 교과목 홈페이지에 공개한다.

이 교과목에서 시간복잡도에 대한 개념은 사용하지 않으나 다양한 크기의 입력에 대한 실행시간을 측정하여 봄으로써, 학기말에는 다른 학생이 발표한 해결 방식을 듣고 대략 어느 정도의 시간이 소요될 것이라는 것을 예측하기도 한다. 최소한 시간복잡도라는 개념이 왜 필요한지를 스스로 인식하게 되어 이후 시간

복잡도를 강의할 때 주도적으로 수업에 참여하는 모습을 보여주고 있다. 일부 수강생들은 프로그래밍이 서툴러서 학기초에는 거의 과제를 제출하지 못하지만, 4-5주 정도 지나면서 과제를 제출하기 시작하여 학기 말까지 지속되는 학생들도 상당수 있음을 발견할 수 있다.

### 3.3 부산외국어대학교 컴퓨터공학부

부산외국어대학교 컴퓨터공학부에서는 2001년부터 학부 2학년 가을학기에 문제해결기법 과목을 개설하고 있다.

강좌 운영은 교수 중심의 강의와 학생들의 발표 및 토의로 구성된다. 처음 3주 동안은 C 프로그래밍 기법과 기초적인 자료구조(연결리스트, 스택, 큐 등)를 복습하고, 간단한 예제를 사용하여 문제 해결 절차를 학습하게 한다. 4주째부터는 문제 풀이 중심으로 학생들의 발표와 토의를 통한 수업과 문제 풀이 과정 중에 필요한 기초 알고리즘 교육과 트리와 그래프에 관한 내용을 강의하는 것을 병행한다.

강좌의 목표는 프로그래밍의 이해와 기본적인자료 구조와 순환 등 프로그래밍 기법을 문제 해결에 활용할 수 있는 능력을 배양하는 것으로 둔다. 따라서 강좌에서 다루어지는 문제들은 수강생의 수준과 학습 진도에 맞추어 선택된다.

강의 시간에 다루는 문제는 1주 전에 학생들에게 제공한다. 항상 ACM-ICPC 대회나 KOI 대회의 형식을 따르도록 하여 학생들이 과제로서 주어지는 문제의 형식에 익숙하도록 한다. 출제하는 문제를 모두 과제로 지정하지 않고 자율적으로 풀어오도록 하기도 한다. 그런 문제들은 보통 시간복잡도의 개념을 잘 파악할 수 있는 다양한 해법이 가능한 문제이거나 아주 전형적인 해법을 지닌 문제들로서 기초 알고리즘이나 자료구조 학습에 유용한 문제들이다. 과제로 지정되는 문제의 개수는 학기마다 다르지만 보통 4~5문제 정도이다. 과제는 학생이 혼자 해결하도록 주어지기도 하고, 2명이 한 팀을 이루어 해결하도록 하기도 한다. 과제의 결과물로서 간단한 설명을 곁들인 소스 프로그램을 교과목 게시판에 등록하도록 한다.

문제 토의 시간에는 학생들의 발표를 유도하기 위해 발표시마다 점수를 부여한다. 학생들이 해결하지 못하는 문제에 대해서는 교수가 점진적으로 해답에 가까운 힌트를 제공하여 학생들이 문제 해결에 기여하도록 한다.

중간고사와 기말고사는 필기시험으로 실시한다. 중간고사는 프로그래밍에 관한 내용으로 주어진 프로그램의 출력값을 쓰는 문제가 주종을 이룬다. 기말고사는 실기 시험을 치르거나 필기시험인 경우 트리와 그래프와 같은 자료구조에 관한 내용과 주어진 문제의 해법을 서술하는 형식의 문제를 출제한다.

문제해결기법 교과목을 수강한 후에도 문제 해결에 대한 흥미 유도와 프로그래밍 활용 능력의 향상을 위해 학부 홈페이지(<http://computer.pufs.ac.kr>)에 문제 풀이를 위한 게시판을 운영한다. 이 게시판을 전 학생을 대상으로 하며 프로그래밍 문제를 한 달에 세 번 정도 출제하고 올바른 프로그램을 제출한 학생에 대해 선착순으로 간단한 시상을 한다.

## 4. 결론

본고에서 학부 학생들의 창의적인 문제 해결 능력과 프로그래밍 활용 능력을 향상시키기 위한 문제해결기법 교과목의 국내외 대학 운영 사례를 중심으로 살펴보았다.

문제해결기법 교과목은 개설 학년, 수강생의 수준이나 조교의 활용 등 대학마다 교육 환경이 다르기 때문에, 각 대학에 적합한 수업 모델을 설정해야 할 것이다. 그렇지만 학생들이 프로그래밍에 흥미를 갖게 하고 새로운 문제를 스스로 해결하려는 도전 정신을 불러일으키려는 교과목 개설 취지에 맞게 운영하는 것이 필요하다.

학생들의 수준에 맞는 문제들을 발굴하고, 매주 과제를 평가해야 하는 것은 매우 힘든 작업이다. 문제은행의 운영과 자동 채점 시스템의 보급 등 교과목을 원활하게 운영할 수 있는 여러 가지 방안을 범 대학 차원이나 학회에서 연구할 수 있기를 희망한다.

## 참고문헌

- [1] 김재훈, 배상원, “아시아 지역대회 동향 분석”, 한국정보과학회지 25권 7호, 2007년 7월.
- [2] <http://cs.kaist.ac.kr/class/cs202>
- [3] <http://tcs.cuk.ac.kr/~jhpark/courses.html>
- [4] <http://galab.pufs.ac.kr/~shkim/technote/prob.cgi>
- [5] <http://www.cs.cornell.edu/courses/cs305/2006fa>,
- [6] <http://www.ugrad.cs.ubc.ca/~cs490/>
- [7] <http://www.cs.duke.edu/courses/cps149s/spring07/>



**김수환**

1983~1987 서울대학교 계산통계학과(학사)  
1987~1989 한국과학기술원 전산학과(석사)  
1989~1995 한국과학기술원 전산학과(박사)  
1992~현재: 부산외국어대학교 컴퓨터공학부 교수  
관심분야: 알고리즘 설계, 계산기하학  
E-mail : shkim@pufs.ac.kr



**박정흠**

1981~1985 서울대학교 계산통계학과(학사)  
1985~1987 한국과학기술원 전산학과(석사)  
1987~1992 한국과학기술원 전산학과(박사)  
1992~1993 한국과학기술원 정보전자연구소 연  
수연구원  
1993~1996 한국전자통신연구소 부호기술연구  
부 선임연구원

1996~현재 가톨릭대학교 컴퓨터정보공학부 교수  
관심분야: 알고리즘 설계, 그래프 이론  
E-mail : j.h.park@catholic.ac.kr



**좌경룡**

1971 서울대학교(학사)  
1977 Northwestern University(석사)  
1980 Northwestern University(박사)  
1985~1986 Visiting Scientist IBM T.J. Waston  
Research Center  
1997~1998 한국정보과학회 회장

2003~2004 한국과학기술원 공학장  
2004~현재 한국과학기술원 학원 정회원  
2004~현재 한국공학한림원 정회원  
1981~현재 한국과학기술원 전자전산학부 교수  
E-mail: kychwa@cs.kaist.ac.kr

**제34회 정기총회 및 주제학술발표회**

- 일 자 : 2007년 10월 26~27일
- 장 소 : 부산대학교
- 내 용 : 논문발표 등
- 주 최 : 한국정보과학회
- 상세안내 : 사무국 한영진 과장(02-588-9246/7)