

특집  
10

## S/W공학 글로벌 리더 양성과정 - KAIST-CMU S/W공학 석사 프로그램1)

### 목 차

1. 서 론
2. MSE 프로그램 커리큘럼
3. MSE 프로그램의 교육철학
4. MSE 프로그램에 대한 회고적 성찰
5. 소프트웨어 공학 참조 커리큘럼
6. KAIST-CMU MSE 프로그램의 발전방향
7. 결 론

강성원 · 고인영 · 백종문 · 최호진 · 이단형  
(한국과학기술원)

### 1. 서 론

KAIST-CMU MSE 프로그램<sup>2)</sup>은 한국과학기술원(Korea Advanced Institute of Science and Technology: KAIST)이 미국의 카네기멜론 대학교(Carnegie-Mellon University: CMU)가 공동으로 운영하는 소프트웨어공학 석사(Master of Software Engineering: MSE) 프로그램이다. 이 프로그램을 졸업하는 학생은 KAIST의 석사 학위와 CMU의 석사학위를 같이 받게 된다. MSE 프로그램은 2002년 4월 고품질의 소프트웨어공학의 교육과 연구를 한국의 산업체에 제공하기 위하여, 방송통신위원회의 전신인 정보통신부가 지원하여 KAIST에 설립되었다.

“가장 흔한 소프트웨어 위험은 부적합한 소프트웨어공학 커리큘럼이다”<sup>[2]</sup>라는 말이 있는 것과 같이, 좋은 소프트웨어를 개발하는데 좋은 소프트웨어 공학 기술이 필수적이고, 좋은 소프트웨어 공학교육을 위하여는 좋은 소프트웨어 공학 커리큘럼이 필수적이다. 따라서 이 논문에서는 MSE 프로그램의 커리큘럼을 위한 단기, 장기 개선방향을 찾기 위하여 과거 5년간의 프로

그램 운영에 대하여 고찰하여 보고 미래를 위한 개선 방향을 도출한다. 이러한 목적을 위하여, 이 논문에서는 두 가지 조사를 수행하는데, 하나는 프로그램에 참여하였던 교수진과 졸업생들의 회고적 성찰(reflection)을 통한 것이고 다른 하나는 다양한 참조 커리큘럼을 조사하는 것이다. 그리고 이러한 결과에 기초하여 특수 트랙의 운영을 포함하는 개선된 커리큘럼을 도출한다.

본 논문은 다음과 같이 구성된다: 2절에서는 현재 운용되고 있는 MSE 커리큘럼을 소개한다. 3절에서는 MSE 프로그램의 교육철학을 소개한다. 4절에서는 KAIST-CMU MSE 프로그램의

1) 본 논문은 CSEET 2010에 발표된 논문 [1]을 수정 확장한 논문이다.

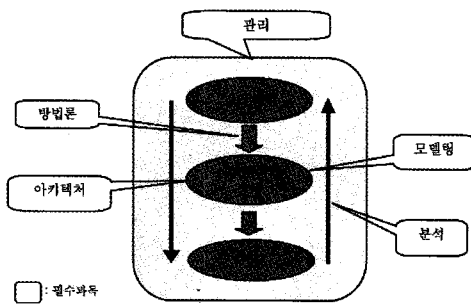
2) KAIST-CMU MSE 프로그램의 원래 명칭은 ICU-CMU MSE 프로그램이었다. ICU는 한국정보통신대학교(Information and Communications University)를 지칭한다. 그러나 2009년 3월 1일, 양 교가 통합되어 프로그램의 명칭은 KAIST-CMU MSE 프로그램으로 대체되었다.

† 저자들은 회고적 성찰에 참여하여 준 카네기멜론 대학교 교수진과 KAIST-CMU MSE/MSIT-SE 졸업생들에게 감사한다. 또한 논문 작성에 도움을 준 카네기멜론 대학교의 MSE 프로그램 매니저인 Jane Miller에게도 감사드린다.

과거 5년에 대한 회고적 성찰의 결과를 요약한다. 5절에서는, 다양한 석사과정의 소프트웨어공학프로그램의 참조모델을 조사한다. 6절에서는 새로운 커리큘럼의 구조를 제시하고 개선된 프로그램을 제시한다. 7절에서는 논문의 결론을 맺는다.

## 2. MSE 프로그램 커리큘럼

KAIST-CMU MSE 프로그램은 2년 또는 그 이상의 산업계 소프트웨어 개발 경험이 있는 학생들을 대상으로 한다. 본 과정은 5개의 필수 과목과 다수의 선택 과목, 그리고 산업체에서 필요로 하는 시스템을 개발하는 스튜디오 프로젝트로 구성되어 있다. 필수 과목에서는 소프트웨어 공학 분야의 기본 원리를 강화하는 것을 목표로 하고 있으며, 이들 과목으로는 “소프트웨어 시스템 모델,” “소프트웨어 개발 방법론,” “소프트웨어 개발 관리,” “소프트웨어 산출물 분석,” 그리고 “소프트웨어 시스템 아키텍처”가 있다. (그림 1)은 MSE 프로그램의 필수과목들 간의 관계를 보여주고 있다.



(그림 1) 소프트웨어 개발과정과 MSE 프로그램의 필수과목들 간의 관계

선택 과목으로는 “요구 공학,” “소프트웨어 프로세스,” 그리고 기타 컴퓨터 과학과 소프트웨어 공학 관련 과목들이 있다. 본 과정의 중요한 특성으로는 수업에서 배운 이론들을 실제 소프트

웨어 개발 프로젝트에 적용한다는 점이다. 소프트웨어 스튜디오에서는 학생들은 실제 산업체 프로젝트를 의뢰인과 함께 수행하게 된다. 학생들은 프로젝트 별로 팀을 구성하여 팀 개발 환경에서 프로젝트 계획과 일정을 수립, 소프트웨어 개발을 수행하고 스튜디오의 멘토들의 지도하에, 진행 상황을 추적하기 위한 방법들을 사용하게 된다.

MSE 과정은 1년에 3학기를 운영하는 집중 과정이다. 학생들은 첫 학기를 KAIST 캠퍼스에서 수업에 참여하고, 두 번째 학기는 피츠버그의 카네기멜론 대학 캠퍼스에서 수업을 받고, 세 번째 학기는 다시 KAIST 캠퍼스에서 스튜디오 프로젝트를 마무리하게 된다. 이러한 1년간의 전공 수업과 스튜디오 프로젝트 개발이 끝나면, 학생들은 KAIST 석사 학위 요건을 만족하기 위한 석사 학위 논문을 작성하게 된다.

MSE 과정의 높은 교육 수준을 유지하기 위해 필수 과목은 카네기멜론 대학의 교수와 카네기멜론 대학 겸임교수직을 가지고 있는 KAIST 교수에 의해 수업이 진행된다. 학생들은 멘토들의 지도를 받으며 소프트웨어 개발 스튜디오 프로젝트를 수행하게 된다. 또한 “성찰적 실천 (Reflective Practice)”을 통하여 학생들은 필수/선택 과목에서 습득한 내용을 실무에 적용하고 그 결과를 분석할 수 있는 능력을 계발하며, 소프트웨어 프로젝트에서 수행되어야 할 일을 판단하는 능력과, 그러한 이유를 평가 하는 방법들을 습득하게 된다.

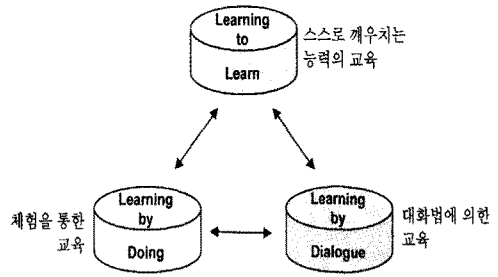
MSIT-SE(Master of Science in Information Technology - Software Engineering) 과정은 MSE 과정의 자매 과정으로 소프트웨어 개발 경험이 없는 학생들을 대상으로 한다. MSIT-SE와 MSE의 차이점은 개별적으로 또는 두, 세 명의 학생들이 그룹을 이루어 스튜디오가 아닌 프랙티컴(Practicum)에 참여하게 된다는 것이다.

### 3. MSE 프로그램의 교육철학

MSE프로그램의 교육철학은 크게 세 개의 교육철학의 원리를 그 바탕으로 한다고 볼 수 있다. (그림 2)는 세 개 교육철학의 원리를 보여주고 그들이 서로 보완적인 관계에 있음을 보여준다.

첫째로, 교육의 목적이 지식을 전달하는데 있는 것이 아니라, 스스로 학습할 수 있는 능력을 개발하는데 있다. 현재의 지식의 양은 무어의 법칙에 맞먹을 정도로 1~2년에 2배로 지식의 양의 증가한다고 한다. 많은 직업에 있어서 필요한 기술이 대학교에서 배우는 지식은 5~7년이 지나면, 재교육을 받아야 할 정도로 빨리 변화한다. 또한 자라나는 세대들은 그들의 생애 동안 평균 5년에 한 번 정도 하는 일을 바꾸게 된다고 한다. 이와 같은 환경에서, 지식의 전수만을 목표로 한다면, 생명력이 짧고 경쟁력이 떨어지는 졸업생을 배출하게 될 것이다. MSE 프로그램은 따라서 학생들이 스스로 학습하는 능력을 갖도록 하여, 졸업 후에 시대와 기술이 바뀌어도 스스로 새로운 기술을 체득하고, 적용하고, 응용할 수 있도록 하는데 초점을 맞추고 있다.

둘째로, MSE의 과목들은 강사의 일방적인 강의보다는 대화와 클래스 안팎에서 상호작용을 통하여, 강사는 문제를 제기 하고, 학생들이 보지 못하던 것을 볼 수 있게 유도하여, 학생 스스로가 문제를 해결할 수 있도록 도와주는 역할을 한다. 이와 같은 교육방법을 “산파법”이라고 하는데, 이방법의 핵심은 산파가 아이를 낳는 것을 도와주는 역할을 하며 아기를 낳는 것은 산모 자신인 것처럼, 강사의 역할도 답을 주는 데 있는 것이 아니라, 학생이 스스로 문제를 이해하고 답을 찾아갈 수 있도록 도와주는데 있다는 것이다. 이와 같은 교육방법은 그 뿌리를 소크라테스의 교육방법에 두고 있는데, 이는 소크라테스가 대화 가운데 통찰력 있는 일련의 질문들을 던짐으



(그림 2) MSE 프로그램의 교육철학

로써 상대방이 그 답을 발견하도록 한데서 그 기원이 있다. 이러한 교육철학은 MSE프로그램의 전 과정에 스며있는 철학이고 방법이다. 이와 같은 방법이 MSE에만 담겨있는 것은 아니며, 서양교육이 기본적으로 이런 교육철학에 바탕을 두고 있다. 그러나 MSE의 특징은 이러한 교육철학을 더욱 두드러지게 교육방법의 기반으로 하고 있고, 특히 스튜디오에서는 멘토들의 역할을 통하여 극명하게 교육 방법의 근간을 이루고 있음을 볼 수 있다. 멘토는 우리말로는 “지도교사” 혹은 “조언자”로 부를 수 있는데, 이들의 역할은 학생들이 스튜디오 프로젝트를 수행하는 가운데, 그들에게 필요한 도움을 주고 비판하는 것이다. 여기서 멘토의 역할은 표준적인 방법을 제시하고 요구하는데 있는 것이 아니라, 학생들이 스스로가 결정하도록 하고 이를 위하여 그들이 고려할 수 있는 다양한 선택을 제시해 주고, 그들이 위험한 방향으로 갈 때에는 그 결과를 미리 조명하여 보도록 유도함으로써, 그 위험을 감수하여 나아갈 것인지, 아니면 피해 갈 것인지를 학생 스스로 결정하도록 하는데 있다.

셋째로, 커리큘럼에 스튜디오와 프랙티컴이 포함되어, 학생들이 필수과목과 선택과목을 통하여 학습한 내용들을 직접 적용해 보고 방법이 동작하는지 동작하지 않는지 확인하고, 동작하면 왜 동작하고, 동작하지 않으면 왜 동작하지 않는지 생각해 볼 수 있는 실험실을 제공한다. 동양격언에 “들은 것은 잊어버리고, 본 것은 기

억하지만, 해본 것은 이해한다”는 말처럼, 직접 적용하여 봄으로써 이해하게 되고 내재화 되게 된다. 이러한 방법은 “경험적 교육(experiential education)”이라고도 알려져 있다.

#### 4. MSE 프로그램에 대한 회고적 성찰

이 절에서는 MSE 프로그램 과거 5년간의 운영에 대하여 회고적 성찰을 수행하고 그 결과를 요약 한다. 회고적 성찰은 KAIST 교수진, CMU 교수진 및 직원, 그리고 MSE 프로그램 졸업생들이 참여하는 설문조사를 통하여 수행되었다. 또한 이 절에서는 이들 교수진과 학생들이 제공한 각종 통계자료 및 의견들을 토대로 MSE 프로그램에서 계속 유지할 점, 제거할 점, 개선할 점들을 정리한다. 이 조사에는 KAIST 교수진 5명, CMU 교수진 4명, CMU 행정직원 1명, 그리고 MSE 졸업생 7명이 참여하였다.

〈표 1〉은 프로그램의 시작된 후 첫 5년 동안의 MSE 프로그램 입학생수와 졸업생수를 보여주고 있다. 처음 2년 동안은 모든 입학생들에게 장학금으로 등록금 면제 혜택이 주어졌고, 3년 차 입학생부터는 등록금을 납부하게 되어, 학생들은 자비로 등록금을 내거나 기업의 위탁교육생들은 기업에서 등록금을 부담하였다. 이렇게 등록금 면제 혜택이 없어짐으로써 지원자 수가 감소하였고, 따라서 입학생 수도 줄어들게 되었다. 한편, 일부 학생들은 졸업을 못하였는데, 이들 대부분은 프로그램기간 동안에 KAIST 졸업논문 요건을 만족하지 못하여 졸업이 미루어진 상태이다.

조사 결과 MSE 프로그램은 강도 있고 엄격한 커리큘럼, 고수준의 품질표준, 필수과목과 선택과목의 효과적인 배합, 스튜디오 프로젝트의 중요성, 이론을 실제에 적용하는 면 등에서 전반적으로 긍정적인 평가를 받았다. KAIST와 CMU 교수진 대부분은 현재의 다섯 개의 필수과목은 중요하고 잘 구성되어 있다고 보고 있으나, 학생

〈표 1〉 KAIST-CMU MSE/MSIT-SE 프로그램 입학생 및 졸업생수

개강년도 (개강 8월)	입학생수	졸업년도 (졸업 2월)	졸업생수
2003	15	2005	14
2004	14	2006	13
2005	11	2007	7
2006	11	2008	10
2007	9	2009	3
합계	60명	합계	47명

들 중 일부는 필수과목 간에 실용성의 차이가 큰 쪽으로 있다고 보고 있다. 선택과목 중에서는 ‘소프트웨어 프로세스’와 ‘개별 연구’ 과목이 좋은 평가를 받았다. 또한, 교수진 대부분은 스튜디오 멘토링을 프로그램의 중요한 요소라고 생각하는 반면, 학생들은 실제 산업체의 프로젝트를 수행한다는 것 자체를 더 중요하게 여기는 경향이 있었다.

프로그램이나 교과목, 스튜디오와 관련하여 조정이나 개선할 부분에 대한 의견은 매우 다양하여 본 지면에 요약하기는 힘들다, CMU 교수가 거론한 “CMU 방식의 교육이 한국의 조직문화에 맞지 않는 부분이 있으므로 서양의 소프트웨어공학을 한국 기업문화의 장점과 필요성에 맞게 현재 프로그램을 수정할 필요가 있다”라는 주장에 대하여는 더 검토가 필요하다고 생각된다.

프로그램에 추가나 보완하여야 할 부분으로, 소프트웨어 시험이나 품질과 관련한 교과목, 영어 의사소통 및 발표를 돕는 교과목 등이 거론되었다. 특히, 학생들은 보다 실용적인 교과목이나 초청강연, 실제 산업체 전문가의 멘토링 등을 추가할 것을 제안하였다. 또한, KAIST 교수진은 필수과목에 소프트웨어 시험과 유지보수 과목을 추가할 것을 제안한 반면, 학생들은 현재 필수과목 내에서 보다 실제적인 예제들이 다루어지기를 기대하고 있다. 선택과목 관련으로는, 교수진

과 학생들 모두 선택과목이 더 늘어나야 한다고 보고 있다. 끝으로, 스튜디오 프로젝트 관련하여, CMU 교수진은 한국의 문화와 기업체의 환경에 맞는 부트캠프(Bootcamp) 강의가 제공될 필요가 있다고 지적한 반면, 학생들은 한국 산업체 전문가로부터의 실용적인 경험담 또는 멘토링을 추가할 것을 많이 제안하였다.

MSE와 같은 프로그램이 산업체에 미치는 영향을 분석하기란 쉬운 일은 아니다. 영향을 측정하는데 상당한 시일이 요구되며, 이를 위한 관찰을 진행하며 정확히 측정하려는 결단이 필요하기 때문이다. 그럼에도 불구하고, 상기 조사 결과 본 프로그램이 산업체에 미치는 영향에 대한 긍정적인 증거를 관찰할 수 있었다.

〈표 2〉 첫 5년간의 MSE/MSIT-SE 졸업생 취업 현황

	대기업	연구소/공공 기관	중소기업	유학 및 해외취업
직장	삼성전자(7) 삼성SDS (4) KT(4) LG CNS(2) SK C&C (2) 현대모터스(3) 포스데이터(1) LG 전자(1) SK텔레콤 (1) 동부(1)	ETRI (8) KISTI (1) KCB(1) 한국은행 (1) KFTC(1)	NHN(2) 티맥스(1) 크로센트(1) SAIGE Korea(1)	캐나다 브리티쉬 컬럼비아 대학교(1) 미국 매사추세츠 주립대학교-암허스트분교(1) VistaPrint (1) A.T. Kearney(1)
합계 (47명)	26명	12명	5명	4명

본 프로그램이 배출한 학생들의 성과에 대한 또 다른 긍정적인 증거로서, 졸업생들의 취업상황과 논문발표 실적이 있다. 〈표 2〉에 나타난 바와 같이, 모든 졸업생은 유망한 기업이나 연구소에 취업하는데 큰 어려움 없었으며, 일부는 박사과정 유학 및 취업을 위해 해외로 진출하였다. 본 프로그램의 학생은 CMU의 당초 1년 과정의 프로그램에 이어 논문연구를 한 학기 수행하여 석사학위 논문을 제출해야 한다. 이 논문요건은

KAIST측에 의해 MSE 프로그램에 추가된 사항인데, 과거 5년간 학생들과 KAIST 교수진은 석사학위 논문연구를 통하여 국제학술지와 컨퍼런스에 논문을 게재하거나 발표하는데 많은 노력을 기울였다. 그 결과 2009년 8월 25일 현재, SCI 저널논문 1편, SCIE 논문 2편, 기타 국제저널 논문 1편, 국제 학술대회논문 17개의 총 21개의 논문이 게재 또는 발표되었다.

### 5. 소프트웨어 공학 참조 커리큘럼

IEEE Computer Society와 ACM에서 출판한 “소프트웨어 공학 2004” [3]의 주요 목적은 교육 기관과 인증 기관에게 학부 과정의 소프트웨어 공학 교육과 관련된 가이드라인을 제공하는 것인데, 커리큘럼을 작성함에 있어 유용한 다음과 같은 가이드라인을 제공한다:

- 커리큘럼 작성자들과 교수들은 반드시 결과 관점에서 생각해야 한다.
- 과정을 설계한 사람은 내용의 범위와 혁신을 위한 유연성을 허용하도록 균형을 이루도록 하여야 한다.
- 많은 소프트웨어 공학 개념들, 원리들, 그리고 이슈들은 학생들이 소프트웨어 공학 사고방식을 개발하는데 도움을 줄 수 있도록 커리큘럼 전반에서 반복적인 테마의 형태로 지도되어야 한다.
- 어떤 소프트웨어 공학 주제는 학습하는데 배경지식을 필요로 하므로, 이러한 주제들은 커리큘럼의 마지막 부분에서 가르쳐야 하고 커리큘럼 앞부분에서는 배경이 되는 지식을 가르쳐야 한다.
- 학생들은 소프트웨어 공학 이외에도 적용 분야에 대해서도 반드시 학습해야 한다.

소프트웨어공학 지식체계에 대한 가이드(The Guide to the Software Engineering Body of Knowledge: SWEBOK)는 일반적으로 동의된 소프트웨어 공학 분야의 범위를 정의하고, 이 분

〈표 3〉 SWEBOK와 소프트웨어 공학 2004에 대한 MSE 커리큘럼 비교

	SWEBOK [4]	Software Engineering 2004[3]
현재 MSE 프로그램에서 다루는 주제	<ul style="list-style-type: none"> <li>- 소프트웨어 요구사항</li> <li>- 소프트웨어 설계</li> <li>- 소프트웨어 개발</li> <li>- 소프트웨어 형상관리</li> <li>- 소프트웨어 유지보수</li> <li>- 소프트웨어 공학 관리</li> <li>- 소프트웨어 공학 프로세스</li> <li>- 소프트웨어 공학 도구와 방법</li> </ul>	<ul style="list-style-type: none"> <li>- 전산 기초</li> <li>- 수학 및 공학 원론</li> <li>- 현장적용기술(Professional Practice)</li> <li>- 소프트웨어 모델링과 분석</li> <li>- 소프트웨어 설계</li> <li>- 소프트웨어 진화</li> <li>- 소프트웨어 관리</li> <li>- 소프트웨어 프로세스</li> </ul>
현재 MSE 프로그램에서 다루지 않는 주제	<ul style="list-style-type: none"> <li>- 소프트웨어 품질</li> <li>- 소프트웨어 시험</li> </ul>	<ul style="list-style-type: none"> <li>- 소프트웨어 품질</li> <li>- 소프트웨어 검증(Verification &amp; Validation: V&amp;V)</li> </ul>

야의 지식 본체(Body of Knowledge)의 주제들을 제시하기 위하여 제정되었다[4]. 〈표 3〉은 SWEBOK와 소프트웨어 공학 2004에 비추어 MSE 커리큘럼을 비교한 결과이다.

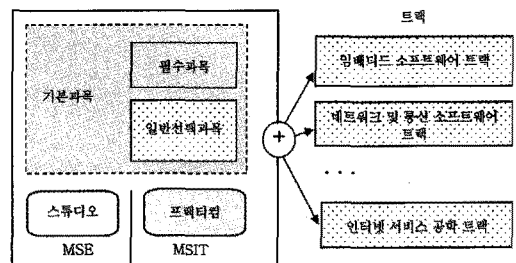
〈표 3〉의 두 번째 열에서 보는 바와 같이 지식 본체는 10개의 소프트웨어 공학 지식 영역(Knowledge Area: KA)으로 구성되며 이 지식 영역은 교육 과정의 인증, 교육 및 훈련 과정의 개발, 전문가 인증 또는 전문가 자격부여와 같은 목적으로 사용될 수 있다. 학부 과정을 위한 교육 커리큘럼은 IEEE CS와 ACM 간의 공동 노력으로 개발되었다[3]. 그 지식 영역들은 표 3의 세 번째 열에 있다. MSE 과정이 소프트웨어 개발과 소프트웨어 공학 도구 및 방법론 지식 영역을 독립된 과목으로 지정하지는 않았지만, 개발에 대한 학습은 프로그래밍 과목에서 이루어지며, 소프트웨어 공학 도구 및 방법론은 다른 과목에서 충분히 학습이 이루어지고 있다.

**6. KAIST-CMU MSE 프로그램의 발전방향**

지금까지 소개한 KAIST-CMU MSE 프로그램에 대한 회고적 성찰과 참조 커리큘럼에 대한 조사를 바탕으로 이 절에서는 KAIST-CMU MSE 프로그램의 발전 방향에 대해 조명해 보기로 한다. 우선 단기적으로는 커리큘럼을 보완하

여 소프트웨어공학 전반에 대한 교육을 더욱 충실하게 다지고, 중장기적으로는 산업계의 수요를 바탕으로 특화된 교육 트랙들을 제공하는 것이 필요한 것으로 보인다.

(그림 3)은 MSE 프로그램의 발전된 커리큘럼 아키텍처를 보여주고 있다. 여기서 필수과목(Core Courses)은 프로그램에 있는 학생들 모두가 들어야 하는 과목들이며, 일반선택과목(General Elective Courses)은 학습의 폭을 넓혀 줄 수 있는 선택과목들을 의미한다. 이 두 종류를 모두 포함한 과목들을 기본과목(Base Courses)이라고 부르기로 한다.



(그림 3) MSE 프로그램의 발전된 커리큘럼 아키텍처

프로그램의 단기적인 발전방향으로 새로운 커리큘럼의 아키텍처에서는 (그림 3)의 오른쪽의 트랙(Track)들을 가까운 미래에 제공한다. 학생들은 기본과목들로 구성된 기본 프로그램을 선

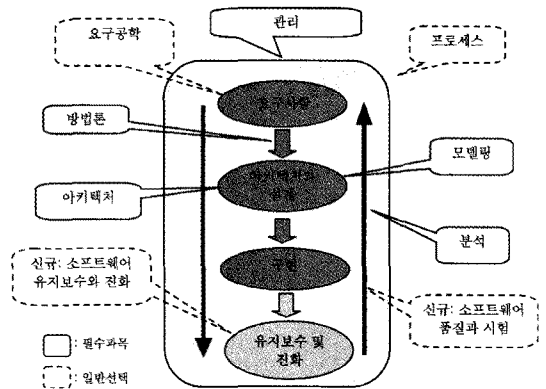
〈표 4〉 주요 트랙의 핵심 교과목 구성

트랙	T1. 임베디드 소프트웨어 트랙	T2. 네트워크 및 통신 소프트웨어 트랙	T3. 인터넷 서비스 공학트랙
과목	<ul style="list-style-type: none"> <li>- 임베디드 시스템 공학</li> <li>- 임베디드 실시간 시스템</li> <li>- 임베디드 제어시스템</li> <li>- 소프트웨어 엔지니어를 위한 하드웨어</li> <li>- 소프트웨어 개발도구</li> </ul>	<ul style="list-style-type: none"> <li>- 무선네트워킹</li> <li>- 네트워크 아키텍처</li> <li>- 보안 소프트웨어 시스템</li> <li>- 개방형 분산 처리</li> <li>- 네트워크 상황인식</li> <li>- 인터넷과 공공정책</li> <li>- 모바일커머스</li> <li>- 모바일-퍼베이시브 컴퓨팅 서비스</li> <li>- 웹, 보안, 개인정보보호</li> </ul>	<ul style="list-style-type: none"> <li>- 인터넷과 공공정책</li> <li>- 유비쿼터스 컴퓨팅</li> <li>- 온라인 정보설계</li> <li>- 가상 환경 관리</li> <li>- 기업가 정신</li> <li>- 웹 서비스</li> <li>- 웹 응용 개발</li> <li>- e-Payment</li> </ul>

택하거나 특정 트랙을 선택하여 공부할 수 있게 된다. 임베디드 소프트웨어 트랙, 네트워크 및 통신 소프트웨어 트랙, 인터넷 서비스 공학 트랙 등이 그 예이다. 이 트랙들에서 제공하게 될 핵심 교과목들은 〈표 4〉와 같다.

새로운 커리큘럼 아키텍처에서는 트랙에 속하는 과목들이 일반선택과목으로부터 독립하여 나가고, 공통적이고 특정 도메인에 종속적이지 않은 과목들만이 일반선택과목으로 남게 된다. 각 트랙은 해당 트랙에 대한 전문성을 확보해 줄 수 있을 만큼 풍부한 수의 과목이 깊이 있는 내용들로 구성되어 제공되게 된다.

프로그램 전체는 이렇게 새로운 구조를 갖게 되는데, 이와 같이 트랙에 특성화된 과목들을 제외하고 난 후 남는 기본과목들도 (그림 4)에서와 같이 새롭게 구성할 계획이다. (그림 4)는 현재의 MSE프로그램의 기본과목들과 소프트웨어 개발 단계와 연관하여 그들간의 관계를 보여주고 있다. 그리고 이러한 기본과목들 외에 추가적으로 필요한 과목들이 점선으로 표시되어 있고, 소프트웨어 개발 단계와 연관하여 그들의 역할을 보여주고 있다. 기존의 커리큘럼에서는 요구사항, 설계, 구현의 세 단계에 초점을 두어 기본과목들을 구성한 반면 새로운 커리큘럼에서는 '유지보수와 진화'의 단계를 추가하여 네 개의 소프트웨어 개발 단계를 고려하여 기본과목들을 구성한다. 이렇게 네 단계를 고려하는 이유는 최



(그림 4) 소프트웨어 개발단계를 중심으로 한 MSE 기본과목들간의 관계

근에 임베디드 소프트웨어, 모바일 소프트웨어, 웹 기반 소프트웨어 등의 다양한 도메인에서 소프트웨어의 진화와 신뢰성이 특별히 강조되고 있기 때문이다. 이를 위해 새로운 커리큘럼에서는 '소프트웨어 유지보수와 진화(Software Maintenance and Evolution)' 과목을 추가하고, 소프트웨어 신뢰성을 전문적으로 다루는 과목인 '소프트웨어 품질 및 시험(Software Quality and Testing)'을 포함한다.

이러한 새로운 커리큘럼은 SWEBOK에 명시된 핵심 주제들과 소프트웨어공학 교육의 새로운 추세를 반영한다. 특히 '소프트웨어 유지보수와 진화' 과목과 '소프트웨어 품질 및 시험' 과목의 내용은 SWEBOK과 IEEE/ACM에서 제시하는 가이드라인에 따라 구성되도록 할 것이다.

〈표 5〉 새 교과목의 주요 구성 내용

과목명	유지보수와 진화	소프트웨어 품질과 시험	
과목내용	<ul style="list-style-type: none"> <li>- 유지보수비용추정</li> <li>- 소프트웨어 유지보수 측정</li> <li>- 유지보수 프로세스와 활동</li> <li>- 프로그램 이해</li> <li>- 재공학</li> <li>- 역공학</li> <li>- 소프트웨어 진화</li> <li>- 진화계획</li> </ul>	<ul style="list-style-type: none"> <li>- 소프트웨어 품질개념 및 문화</li> <li>- 품질의 가치와 비용</li> <li>- 모델과 품질특성</li> <li>- 품질 향상</li> <li>- 소프트웨어 품질 표준</li> <li>- 소프트웨어 품질 프로세스</li> <li>- 프로세스 보증</li> <li>- 제품 보증</li> <li>- 소프트웨어 품질 보증</li> <li>- 검증</li> <li>- 검토와 감사</li> <li>- 결함분류</li> </ul>	<ul style="list-style-type: none"> <li>- 소프트웨어 품질 측정</li> <li>- 시험 프로세스</li> <li>- 시험 대상</li> <li>- 시험 기법</li> <li>- 명세기반</li> <li>- 코드기반</li> <li>- 결함기반</li> <li>- 용도기반</li> <li>- 응용종류별</li> <li>- 기법의 선정과 결합</li> <li>- 시험대상 프로그램의 평가</li> <li>- 시험수행의 평가</li> </ul>

〈표 5〉는 이 과목들에서 다루게 될 주요 내용을 열거하고 있다.

그리고 현재의 ‘소프트웨어 분석(Software Analysis)’ 과목이 소프트웨어 자산의 품질을 분석을 통하여 확보하는데 초점을 맞추는데 비하여, ‘소프트웨어 품질 및 시험’ 과목은 소프트웨어의 품질, 시험에 대한 프로세스적 측면, 그리고 분석적인 접근 이외의 품질확보 방법에 대해서도 심도 있게 다룰 수 있도록 구성된다. 이들 새로운 일반선택과목들은 기존의 MSE 프로그램에서 제공하고 있던 ‘요구공학(Requirements Engineering)’과 ‘소프트웨어 프로세스(Software Process)’ 과목 등과 함께 소프트웨어 시스템의 라이프사이클(Lifecycle) 전반에 대해 다룰 수 있도록 하며, 이와 관련하여 보다 진보적이고 심화된 내용들을 다룰 수 있도록 할 것이다.

중장기적 발전 방향으로 KAIST-CMU MSE 프로그램은 산업계의 수요를 반영하여 다양한 트랙을 제공할 것이며, 특히 다음의 트랙들을 포함한다:

- T1. 임베디드 소프트웨어 트랙
- T2. 네트워크 및 통신 소프트웨어 트랙
- T3. 인터넷 서비스 공학 트랙
- T4. 인간컴퓨터 상호작용 트랙
- T5. 정보 및 보안 트랙

이러한 트랙들은 우리나라가 국제 경쟁력을 가지고 있는 산업분야들과 밀접한 관련을 가지고 있고, 향후 산업계의 요구를 구체적으로 반영하여 트랙 별 교과목을 구성할 것이다.

이와 같은 프로그램의 특화된 트랙 외에도 MSE 커리큘럼의 기반 위에 개설될 수 있는 프로그램으로 산업체와 정부의 CTO들을 대상으로 하는 소프트웨어공학 관리 프로그램(Software Engineering Management, MSIT-SEM)이 있다.

## 7. 결론

이 논문에서 우리는 KAIST-CMU MSE/MSIT-SE 프로그램의 과거 5년간에 대하여 회고적 성찰을 하였다. 또한 한국의 산업체의 특성과 요구를 더 잘 반영하도록 하기 위하여 이 프로그램을 개선하고 확장하는 방법에 대하여 생각을 해 보았다.

MSE 프로그램의 개선과 진화를 넘어서 KAIST는 국내에서 MSE와 유사한 프로그램을 운용할 계획을 갖고 있다. 그 일환으로 MSE의 축소판인 학위과정과 비 학위 과정을 제공할 계획이다. KAIST와 CMU에서는 엄격한 입학 기준을 적용하기 때문에 소수의 수준 높은 학생들만이 KAIST 학위나 혹은 KAIST와 CMU 학위



를 양 교로부터 받을 수 있다. 한편 국내에는 MSE 프로그램이나 이와 유사한 프로그램으로부터 큰 혜택을 받을 수 있는 많은 재능 있는 미래의 소프트웨어 엔지니어가 있다. MSE 프로그램을 그런 사람들에게 까지 제공함으로써, 그들도 MSE프로그램의 설립자들의 비전을 실현하는 소프트웨어공학 분야의 "변화의 주체"[6]가 될 수 있을 것이다.

KAIST-CMU MSE 프로그램은 2011년부터는 본 프로그램에 대한 큰 변화를 계획하고 있는데, 지난 7년 동안 졸업을 위하여 필수적인 사항이던 졸업논문을 없애고, 1년 6개월의 프로그램을 1년 기간의 프로그램으로 바꾸는 것이다. 이와 같은 변화를 계획하게 된 배경에는 교육기간을 줄임으로써 산업계에서 본 프로그램에 더욱 용이하게 접근할 수 있을 것이며 따라서 본 프로그램이 소프트웨어 산업발전에 더 큰 효과(impact)를 줄 수 있을 것이라는 판단이 있었다.

KAIST-CMU MSE 프로그램은 오랫동안 성공적으로 운영된 프로그램으로 평가 받아, 세계의 여러 우수 대학교가 이를 벤치마킹 하였고, 대표적인 사례로 포르투갈의 코임브라 대학교 (University of Coimbra)의 MSE 프로그램이 있다. 이 프로그램은 KAIST-CMU MSE 프로그램을 모델로 탄생하여, 벌써 여러 해 동안 성공적으로 운영되었고, 포르투갈은 이 프로그램을 자국의 대표적인 교육 프로그램을 넘어서 유럽의 소프트웨어 프로그램이 되도록 발전시키고자 노력하고 있다. 이 논문에서 KAIST-CMU MSE 프로그램은 회고적 성찰을 통하여 과거를 돌이켜 보고 이를 바탕으로 더 발전된 프로그램의 미래 모습을 내다 보았다. 그 결과를 토대로 앞으로 KAIST-CMU MSE 프로그램은 소프트웨어 글로벌 리더 양성과정으로서의 그 역할을 더욱 충실히 수행해 나갈 것이다.

## 참고문헌

- [1] Sungwon Kang, Inyoung Ko, Jongmoon Baik, Hojin Choi, Danhyung Lee, "KAIST-CMU MSE Program - the Past and the Future," Proceeding of the 23rd IEEE-CS Conference on Software Engineering Education and Training (CSEET'2010), Pittsburgh, United States, March 9-12, 2010.
- [2] Capers Jones, Assessment and Control of Software Risks, Englewood Cliffs, N.J.: PTR Prentice-Hall, 1994.
- [3] IEEE & ACM JTFCC, Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, IEEE & ACM, August, 2004.
- [4] A. Abran, J. Moore, SWEBOK: Guide to Software Engineering Body of Knowledge, 2004 Version, IEEE Computer Society, 2004. (Also available as ISO/IEC TR 19759:2005)
- [5] Stevens Institute of Technology, Graduate Software Engineering Reference Curriculum Version 0.50, (GSwERC v0.5), October 31, 2008.
- [6] David Garlan, David P. Gluch, James E. Tomayko, "Agents of Change: Educating Software Engineering Leaders," Computer, Volume 30, Issue 11, Pages: 59- 65, November 1997.
- [7] 강성원, 한동수, "소프트웨어 개발 전문가 양성 프로그램," 한국정보과학회, 정보과학회지, 제20권, 제3호, pp. 57-63, 2002년 3월.

### 저자약력



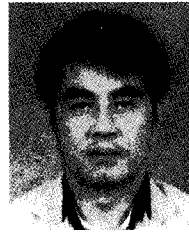
**강성원**

1982년 서울대학교 사회과학대학(정치학사)  
 1989년 Univ. of Iowa 전산학(전산학석사)  
 1992년 Univ. of Iowa 전산학(전산학박사)  
 1993년~2001년 한국통신 연구개발본부 선임연구원  
 1995년~1996년 미국 국립표준기술연구소(NIST) 객원연구원  
 2001년~2005년 한국정보통신대학교 조교수  
 2003년~현재 미국 Carnegie-Mellon University 소프트웨어공학석사과정 겸임교수  
 2005년~2009년 한국정보통신대학교 부교수  
 2009년~현재 KAIST 부교수  
 관심분야 : 소프트웨어 아키텍처, 엔터프라이즈 아키텍처, 소프트웨어 시험, 형식기법  
 이 메 일 : sungwon.kang@kaist.ac.kr



**고인영**

1990년 서강대학교 전산학과 (학사)  
 1992년 서강대학교 전산학과 (석사)  
 2003년 미국 Univ. of Southern California (박사)  
 1993년~1996년 공군사관학교 교관 (전임장사)  
 1997년~2003년 미국 USC Information Sciences Institute(ISI) 연구조교  
 2003년~2003년 미국 USC ISI Postdoctoral Research Associate  
 2004년~2009년 한국정보통신대학교(ICU) 공학부 조교수  
 2009년~현재 한국과학기술원(KAIST) 전산학과 부교수  
 관심분야 : 소프트웨어공학, 웹공학  
 이 메 일 : iko@kaist.ac.kr



**백중문**

1993년 조선대학교 전산통계학과(이학학사)  
 1996년 Univ. of Southern California 전산학(전산학석사)  
 2000년 Univ. of Southern California 전산학(전산학박사)  
 2001년~2005년 Motorola Labs. 수석연구원  
 2005년~2009년 한국정보통신대학교 부교수  
 2006년~현재 미국 Carnegie-Mellon University 소프트웨어공학석사과정 겸임교수  
 2009년~현재 KAIST 부교수  
 관심분야 : 소프트웨어 신뢰성, 소프트웨어 프로세스 모델링, 소프트웨어 식스시그마, 소프트웨어 품질보증  
 이 메 일 : jbaik@kaist.ac.kr



**최호진**

1982년 서울대학교 컴퓨터공학과 (공학사)  
 1985년 영국 뉴캐슬대학교 전산학과 (이학석사)  
 1995년 영국 임페리얼칼리지 컴퓨팅학과 (공학박사)  
 1982년~1989년 (주)데이콤 정보통신연구소 선임연구원  
 1995년~1996년 임페리얼칼리지 IC-PARC 박사후연구원  
 1997년~2002년 한국항공대학교 전자정보통신컴퓨터공학부 조교수  
 2002년~2009년 한국정보통신대학교 공학부 부교수  
 2002년~2003년 미국 카네기멜론대학교 방문교수  
 2003년~현재 미국 카네기멜론대학교 소프트웨어 공학석사과정 겸임교수  
 2006년~2008년 한국정보통신대학교 IT연계교육원장  
 2009년~현재 KAIST 전산학과 부교수  
 관심분야 : 인공지능, 데이터마이닝, 소프트웨어공학, 바이오/의료정보학  
 이 메 일 : hojinc@kaist.ac.kr



**이 단 영**

1971년 서울대학교 공과대학(원자력공학 학사)

1983년 미국 Arthur D. Little(경영과학 석사)

1990년 미국 Virginia Commonwealth Univ.

(정보시스템 박사)

1972년~1996년 KIST/SERI 연구원

1997년~1998년 ETRI 연구원

1999년~2000년 LGCNS 부사장

2000년~2003년 한국소프트웨어진흥원 원장

2003년~2009년 한국정보통신대학교 교수

2004년~현재 미국 Carnegie-Mellon University

소프트웨어공학석사과정 겸임교수

2009년~현재 KAIST 교수

관심분야 : 소프트웨어 요구공학, 소프트웨어 프로세스,

소프트웨어 프로덕트라인

이 메 일 : danlee@kaist.ac.kr