

소프트웨어 위기에 즈음한 전산교육의 실태와 현장 요구사항

한양대학교 | 원유 집*

1. 개요

과거 수십년간 소프트웨어가 지금처럼 화두가 된 적이 없었다. 지난 수년간 반도체, 플래쉬 메모리, 인터넷, 지능형 로봇, 와이브로 등 많은 IT 분야의 핵심 기술이 국가 경쟁력 제고의 견인차로서 각광받고, 주목받고 집중육성 되었다. 지난 정부시절 IT 라는 키워드가 이공학의 여타 분야(기계, 화공, 생명공학 등)에 비하여 상대적으로 주목을 받았던 것이 사실이나, 구체적인 내용을 들여다보면 하드웨어 관련 기술에 대한 내용(투자, 성과 등)이 대부분을 이루고 있다. 반도체, 조선, 철강 등의 수출에 힘입어 우리나라가 세계11위의 경제대국, 아시아 4위의 경제 규모를 이룩하게 되었다는 사실에 그 누구도 의문을 제기하지 않는다. 관련분야의 기술을 선도하고, 세계최고의 위치를 확립할 수 있도록 혼신의 힘을 기울여 온 관련분야종사자들에게 감사할 따름이다. 그런데, 문제는 이제 게임의 판도가 바뀌고 있다는 사실이다. 제품의 경쟁력이 하드웨어를 잘 제작하는 것에서 소프트웨어에 의해 결정되고 있으며, 새로이 정의되고 있는 경쟁에 대해서 국내 산업구조, 기반구조, 교육구조의 경쟁력이 여타 경쟁국에 비해 매우 취약하다는 사실이다.

“아이폰은 팔리고, 옴니아는 왜 안팔리는가?”

현재 우리나라가 직면한 IT 분야의 문제는 “아이폰은 팔리고, 옴니아는 왜 안팔리는가?”라는 매우 간단한 문장으로 요약된다. 답은 하드웨어 보다 소프트웨어가 더 중요하며, 소프트웨어 보다는 콘텐츠가 더 중요하며, 콘텐츠 보다는 뭔가 새로운 “간지”가 더 중요하기 때문이다. 문제는 ‘간지’라는 것이 혹은 ‘포스’라는 것이 전통과 평판에 의해 오랜 시간 응축된 에너지가 폭발적으로 발현될 때 생긴다는 사실이다. 소프

트웨어의 중요성이 점차 강조되면서, 소프트웨어 경쟁력이 해당 제품의 경쟁력이 되면서, 국내 소프트웨어의 기술수준이 다양한 각도에서 재조명받고 평가받고 있다. 소프트웨어 개발인력에 대한 불합리한 인건비 책정방식, 다단계의 도급방식으로 이루어지는 각종 소프트웨어 발주사업, 3D라고 치부되는 소프트웨어 개발에 대한 인식 등 소프트웨어 기술력 제고에 많은 장애요인이 있다. 본 고에서는 교육이라는 측면에서 소프트웨어 기술력 제고에 대학이 어떤 역할을 해야 하는가를 간단히 짚어본다.

2. 현장의 소리

국내 소프트웨어 산업의 기술력 제고를 위해 대학이 제대로 된 소프트웨어 전문인력을 배출하는 것이 매우 중요하다. 불행하게도 국내기업들은 대학이 배출하는 소프트웨어 인력의 수준에 대해 많은 의문사항을 제기하고 있는 것이 사실이다. 모 대기업이 성공적으로 운영하고 있는 “소프트웨어 멤버십” 과정을 예로 들도록 하겠다. 소프트웨어 개발에 관심있는 대학생들을 전국적으로 모집하여 동아리를 구성한다. 회원이 되면 2~3명의 그룹을 이루어 스스로 프로젝트를 정의하고, 설계, 개발, 문서화까지 수행하는 기회를 갖는다. 프로젝트는 매우 실용적인 것들이다. 로봇 제작, 플래쉬 메모리의 구동알고리즘 제작 등이다. 멤버십 프로그램을 운영하는 해당 기업측에서는 예산을 할당하여, 회원들에게 장학금, 숙식까지도 일부 제공한다. 프로젝트에 필요한 장비를 지원하는 것은 물론이다. 대부분이 대학생들이나 회원들은 소프트웨어 개발에 관련된 이론과 실무를 학습할 수 있는 매우 좋은 기회를 갖는 것이다. 해당 프로그램을 운영하는 기업측에서는 자사가 필요로 하는 소프트웨어 개발역량을 미리 교육할 수 있는 기회를 갖는다. 이러한 프로그램을 운영한 결과는 대박이었다. 학생들 사이에

* 종신회원

서는 이 프로그램이 대기업 입사 티켓을 거머쥐는 것으로 인식되어 엄청난 가입 경쟁이 전개되었다. 기업측에서는 고급 소프트웨어 인력을 채용 전에 미리 교육, 검증할 수 있게 되었다. 이로 인하여 신입사원 채용 리스크를 줄이며, 고도로 훈련된 인력을 확보할 수 있는 시스템이 확립할 수 있게 된 것이다. 해당 기업의 각 부서에서는 소프트웨어 멤버십 출신이면 믿고 뽑을 수 있다는 신뢰가 쌓였으며, 소프트웨어 멤버십 출신 신입사원을 확보하는 것이 부서장 평가 지표 중의 하나가 되어버렸다. 그야말로, 누이 좋고 매부 좋은 시스템인 셈이다. 물론 이런 시스템이 생겨남에 따라, 해당 시스템을 갖추지 못한 여타 기업들은 소프트웨어 인력을 채용할 수 있는 기회마저 박탈당했다고 할 수 있다. 그러나, 좋은 회사에 취업하는 것, 능력있는 인재를 채용하는 것은 각각 학생과 기업의 몫이므로 소프트웨어 멤버십이라는 시스템이 가져온 소프트웨어 분야의 대졸 신입사원 채용시장의 구조적 변동은 언급하지 않기로 하겠다. 이제는 다수의 대기업들이 유사한 시스템을 개발 운영 중에 있다.

우리가 여기서 주목해야 할 사실은 기업이 소프트웨어 인력교육에, 자사 직원들의 재교육이 아닌 일반인을 위한 소프트웨어 교육에 팔을 걷어붙이고 있다는 사실이다. 이윤 추구를 목적으로 하는 기업이 자원을 할당하여 소프트웨어 관련 인력교육 시스템을 직접 운영하고 있다는 사실은 소프트웨어 교육의 중추를 담당하는 있는 대한민국의 대학들에게는 낮간지러운 일이다. 대학이 배출하는 소프트웨어 전문 인력의 품질이 기업의 요구사항과 거리가 있다는 것을 한치의 가감없이 적확히 시사하고 있는 것이다. 본 장에서는 대학이 배출하는 소프트웨어 전문인력들에게 있어 강화되어야 할 사항들에 대한 기업의 의견을 간략히 정리해보도록 한다.

소프트웨어 공학 교육의 강화

기업에서 소프트웨어 개발을 담당하고 있는 인력들의 첫 번째 공통적인 의견은 대학에서 “소프트웨어의 품질”에 대한 교육을 강화시켜달라는 것이다. SSD (Solid State Disk)에 탑재되는 전형적인 시스템 소프트웨어에서, 물류 관리 시스템 등의 응용 프로그램에 이르기 까지 소프트웨어가 대형화 되고, 협업에 의한 프로젝트 진행 등이 보편화됨에 따라 개발 프로세스에 대한 지식이 점차 중요해 지고 있다. 그러나, 대부분 대학의 교육과정이 이를 적절히 반영하고 있지 못하다는 의견이었다. 학부교육이 프로그램 기법에 대한 내용(알고리즘, 자료구조, C 프로그래밍 등)은 심

도있게 다루고 있으나, “양질의 코드”에 대한 교육은 상대적으로 매우 미진하다. 가장 대표적인 것이 “예외처리”이다. 저자도 기업과 소프트웨어 개발 과제를 수행할 때, 기업측에서 예외없이 항상 요청하는 것이 제발 “예외처리” 좀 제대로 해달라는 것이다. “양질의 코드”의 정의, 이를 개발하는 방법에 대한 교육이 대폭강화 되어야 한다.

소프트웨어 개발 ≠ 프로그래밍

모 대기업의 연구원으로 다년간 재직했던 인사는 전산전공 학부과정에서 소프트웨어 개발의 단계(software life cycle), 각 단계별 수행해야 할 작업, 단계별로 사용가능한 도구에 대한 교육이 대폭 강화되어야 한다는 점을 지적하고 있다. 소프트웨어 개발은 프로그래밍이 아니다. 정식 전산교육을 받은 사람이거나, 소프트웨어 분야에서 다년간의 개발 경험을 보유한 사람이 본다면 너무나 당연한 이야기이다. 하지만, 소수의 전산 전공관련학과를 제외한 대부분의 대학과 소프트웨어 전문 교육기관에서는 소프트웨어 개발은 프로그래밍이고, 프로그래밍은 곧 C 혹은 자바 프로그래밍이다라는 전제하에서 소프트웨어 교육을 접근하고 있다. 이는 마치 ‘건축은 곧 벽돌쌓기이며 벽돌쌓는 기술이 건축에서 제일 중요하다.’라고 주장하는 것과 진배없다. 소프트웨어 개발 교육은 초기의 기획, 설계단계부터 마지막의 문서화 작업까지의 각 단계가 동일한 중요성을 가지고 진행되어야 한다. 스마트 폰에 탑재되는 소프트웨어가 천만라인을 넘고, LCD TV에 탑재되는 소프트웨어가 수백만 라인이 넘는다. 수백명의 소프트웨어 인력이 개발에 참여한다. 개발과정의 효율성을 기하고, 경쟁력있는 소프트웨어를 개발하기 위해서는 소프트웨어 개발과정에 대한 교육이 대폭강화되어야 한다. 소프트웨어 생명주기에서 테스트, 디버깅, 문서화 등의 작업이 구현(implementation, programming)만큼 중요하다는 인식과 이를 증거하는 다양한 학부수준의 교육 과정이 강화되어야 한다.

프로젝트 기반 교육의 강화

자연과학과 공학은 분명한 차이점이 있다. 자연과학은 근원적인 문제의 해결을 추구하는 것이고, 공학은 현실에 존재하는 문제를 좀더 잘 해결하는 방법을 개발하는 것이다. 학부수준의 전산교육에서 반드시 강화되어야 하는 부분이 실제 구현을 통하여 학습내용을 체화하는 과정이다. 이것을 흔히 Learning By Doing 이라고 한다. 해당 주제에 대해서는 기존 전자공학과 학부과정(대부분의 대학들)을 유심히 살펴볼

필요가 있다. 대부분의 전자공학과 학부 교육과정에서는 실험을 대단히 중시하고 있다. 1학점짜리 과목임에도 불구하고, 많은 교수들이 애정과 관심을 갖고 실험과목의 내용 개발에 직접 참여하고, 교육에 필요한 장비와 환경을 구축하는데 전폭적인 지원을 하고 있다. 아마도 실험교육의 중요성을 이미 인지하고 있기 때문이라 생각된다. 전자공학과와 실험과목에서는 남뎀부터 시작해서, 간단한 앰프설계, 마이크로 콘트롤러의 설계, 어셈블리 프로그래밍, 그리고 고학년이 되면 VLSI 설계까지 다룬다. 이를 통하여 학생들은 일반 수업시간에 배웠던 신호처리, 아나로그 회로 설계, 고주파 회로 설계 등 다양한 이론들을 실제 구현해 볼 수 있는 기회를 갖게된다. 실험의 내용은 매년 새로이 개편되고 있다.

이에 반해 전산학과와 학부교육과정에서는 수업에서 배운 내용을 실습해 볼 수 있는 기회가 상대적으로 부족하다. 수업에 다른 개념을 숙제를 통해 실제 구현해 보는 단편적인 형태의 실습은 많이 보편화되어있다. 이제는 이에서 한걸음 나아가 한학기 분량의 강의에서 구체적인 최종 결과물을 설정하고 이를 단계별로 구현해보는 교육과정의 개발이 필요하다. 예를 들어, 간단한 웹 브라우저를 제작하면서, 제작과정에서 등장하는 다양한 공학적 문제점들을 해결하는 방법을 학습하는 것이다. HTML을 파싱하는데 적합한 자료구조의 설계, 다중 쓰레드를 이용하여 브라우저의 속도를 향상시키고, 이들을 병렬화 하는 알고리즘을 개발하고, 화면에 보기 좋게 나열하는 UI를 개발하는 등의 작업이 그것이다. 실험/실습과목이 매학년 매 학기에 개설되어야 한다. 전산관련 사설 교육기관이나 대기업에서 운영하는 소프트웨어 멤버십 과정은 한가지 특징을 가지고 있다. 실제적인 것을 만들어 보는 기회를 학생들에게 부여하며 이를 강조한다. 대학은 이제 동급대학들 뿐 아니라 사설 교육기관이나 비공식적으로 운영되는 동아리등과도 신입생 유치 경쟁을 벌이게 되었다. 대학은 높은 가치를 추구하고 근본원리에 충실한 교육을 해야 한다. 이를 부정하고자 함은 아니지만, 졸업장 발부하는 것으로 학교의 역할이 완료되는 시대는 한참전에 끝났다. 소프트웨어 기술발전에 공헌할 수 있는 인재를 양성함에 있어, 프로젝트 수행을 강조하는 내용이 전산관련 대학 교육과정에서 심도있게 반영되어야 한다.

현장형 인재의 함정

최근 들어 “수요 지향형”, 내지는 “현장형” 인재양성을 기치로 내 건 대학들이 급격히 증가하고 있다.

대졸 신입사원 재교육에 상당시간을 투자해야 한다는 기업들의 불멘소리가 정부에 전달되고 해당 목소리를 반영한 각종 교과과정 개편 사업이 실행된 결과이다. 매우 바람직한 현상이며 전산학과 교육과정도 이를 적절히 반영하고 있다. 한가지 간과해서는 안 될 사실은 현장에 직접 적용이 가능한 지식의 경우 지식수명이 매우 짧을 수 있다는 것과 근본 원리에 대한 이해 없이 실행 지식만을 습득할 경우, 모방은 할 수 있으나 새로운 기술의 창조는 어렵다는 사실이다. 프로그래밍 언어에 대한 교육을 예로 들어보자. 다양한 프로그래밍 언어를 교육하는 것은 매우 중요하다. 그러나, 이와 더불어 주어진 프로젝트를 가장 효과적으로 달성할 수 있는 프로그래밍 언어를 상황에 적합하게 선택하고 사용할 수 있는 역량을 함양할 수 있도록, 즉 프로그래밍 언어의 원리와 작동 방식에 대한 교육이 반드시 수반되어야 하는 것이다. 또 하나의 예를 들어보자. EXT3 파일 시스템에서는 메타데이터를 파일 시스템의 시작 부분에 배치하고 있다. EXT3 파일 시스템 자료구조를 설명해주는 것보다, 파일 시스템 설계 이면에 존재하는 설계원리 및 철학 등을 이해하는 것이 더욱 중요한 것이다. 3년이 지난 책들은 거의 무용지물이 되는, 급격하게 발전하는 전산분야에서 현장지식을 교육과정에 적절히 반영하는 것은 매우 중요하다. 그러나, 보다 더 중요한 것은 교육과정을 이수한 인력들이 급격한 기술발전에 동적으로 효율적으로 대응할 수 있는 역량을 함양하는 것이다. 재미있게도 이를 위해서는 전산분야의 기본 지식들, 알고리즘, 자료구조, 운영체제 등의 근본 원리에 대한 심도 이해가 전제되어야 하며, 이를 위한 교육이 강화되어야 한다는 사실이다.

3. 맺음말

소프트웨어 위기라고 한다. 국내 소프트웨어 기술력의 부재가 소프트웨어 시장에서의 경쟁력 저하뿐 아니라, 산업 전반의 경쟁력 저하로 파급될 위험에 이미 처해있다. IT 분야 뿐 아니라 우리가 일상생활에서 접하는 거의 모든 문제들이 소프트웨어 때문에 발생되고 있다. 국산 스마트폰의 판매부진, 자동차의 급발진, 인터넷 쇼핑몰에서 주문한 옷의 배송지연 등, 다양한 문제들이 실제로 소프트웨어 불량에서 연유되었거나 연유되었을 것으로 추측되고 있다. 이로 인하여 직간접적으로 지불되는 비용은 실로 엄청나다. 얼마 전 정부에서 연간 1조원씩 3년간 총 3조원을 투자하여 세계 최고의 소프트웨어를 만들어 보겠다는

야심찬 계획을 발표하였다. 정말 반가운 사실이 아닐 수 없다. 그러나, 한편으로는 소프트웨어 기술개발 정책을 하드웨어 식으로 추진하는 것은 아닐까 염려가 되기는 하지만, 어떤 형태로든 소프트웨어 기술력 제고에 관심이 관심과 투자가 집중되고 있다는 것은 매우 고무적인 일이다.

소프트웨어 분야의 전반적인 기술력 제고를 위해서는 대학들은 잘 훈련된 소프트웨어 전문인력을 배출해야하고, 산업계는 고급인력을 채용하여 수익을 창출해야한다. 그리고, 창출된 수익의 일부를 환원하여 대학의 소프트웨어 교육과정 개선에 재투자 하여야 한다. 불행하게도, 아직까지 소프트웨어 분야에서는 이러한 선순환 구조가 정착되지 않았다. 이러한 선순환 구조를 정착시키기 위한 작은 노력으로 소프트웨어 교육에 있어서 대학이 개선해야할 사항을 기업들의 요구사항을 빌려서 기술하였다.

우리나라에는 세계 최고 기술이 많다. 반도체, 철강, 조선, 핸드폰 등이 그것이다. 근자의 소프트웨어 위기에 대한 국가적 인식과 전폭적 투자를 기화로 하여 수년 후 국내 소프트웨어 기술이 우리나라가 보유한 세계 최고 기술 리스트에 다른 분야와 더불어 당당하게 어깨를 견줄 수 있기를 바란다.



원유집

1990 서울대학교 계산통계학과 학사(졸업)

1992 서울대학교 계산통계학과 석사 졸업

1997 University of Minnesota 박사 졸업

1997~1999 Intel 연구원

1999~현재 한양대학교 전자컴퓨터통신공학과
부교수

관심분야: 운영체제, 컴퓨터네트워크, 성능평가

E-mail : yjwon@ece.hanyang.ac.kr
