

## 교과목 개요

### SEP501 컴퓨팅 시스템 개요 (Computing System Overview) 3:0:3

본 과목은 컴퓨팅 시스템에 대한 전반적인 지식을 다루는 과목으로, 컴퓨터 구조 및 운영 체제와 네트워크의 기본 개념, 시스템의 발전 과정 및 향후 발전 추세 등을 소개하며 이들 시스템 관련된 기술의 종합적인 연관 관계를 살펴본다. 이 과목을 통하여 소프트웨어 개발자들이 숙지해야 하는 최소한의 컴퓨팅 시스템에 대한 지식 배양을 목표로 한다.

### SEP502 소프트웨어 시스템 개요 (Software System Overview) 3:0:3

본 과목은 소프트웨어 개발자들에게 필요한 소프트웨어 시스템에 대한 전반적인 지식을 소개하는 과목으로, 프로그래밍 언어, 데이터베이스 및 인공지능 등 소프트웨어 시스템 관련된 기본 개념 및 원리를 다루고, 이들 소프트웨어 기술이 소프트웨어 시스템 개발에 사용되는 사례들을 소개함으로써 소프트웨어 개발자들의 실무 적용 능력 배양을 목표로 한다.

### SEP503 소프트웨어 이론 개요 (Computer Science for Software Engineers) 3:0:3

소프트웨어 엔지니어가 소프트웨어공학의 대학원 수준의 연구를 시작하는 데 필요한 각종 기본적인 주제(고급 이산수학, UML 2.0 및 디자인 도구, 고급 자료구조 및 알고리즘, 프로그램 실행, 객체지향 기법)를 학습한다. 또한, 시간이 허용된다면, 오늘날 소프트웨어공학 분야에서 대두되고 있는 주제(미들웨어, 디자인 패턴, 컴포넌트 기술)도 다룬다. 본 수업에서는 Java를 공식적인 프로그래밍 언어로 활용한다.

### SEP521 소프트웨어 공학 원리 (Principles of Software Engineering) 3:1:3

소프트웨어 공학 전반에 걸친 기본 개념 및 원리를 소개한다. 소프트웨어 공학의 일반적인 기초 지식을 습득하고, 개발 각 단계별 활동 및 분야별 소프트웨어공학 주요 이슈를 다루고, 다양한 개발 프로세스 모델을 비교 분석한다. 또한 소프트웨어 산업 동향과 소프트웨어 개발 기술의 현황 및 향후 동향에 대해서 다룬다.

### SEP522 요구공학 (Requirements Engineering) 3:1:3

요구 사항 추출, 명세 및 검증 기법에 대해 다룬다. 소프트웨어 개발의 설계 이전 단계인 타당성 조사, 비즈니스 모델링 기법 및 요구 사항 분석 등에 대한 기존의 기법을 배우고, 산업체에서 활용되는 정형화된 분석 기법 및 관련 도구를 이용한 실습을 통하여 요구 사항 분석 실무 능력을 배양한다.

### SEP523 소프트웨어 설계 (Software Design) 3:1:3

소프트웨어 설계 원리 및 개념, 소프트웨어 아키텍처 및 아키텍처 스타일, 소프트웨어 설계 notation 및 객체 지향, 컴포넌트 기반 개발 기법, 설계 패턴, refactoring 및 설계 평가 기법 등을 숙지하고, 다양한 CASE 도구를 이용하여 실전적인 modeling 경험을 배운다.

### SEP524 소프트웨어 품질 보증 (Software Quality Assurance) 3:1:3

소프트웨어 개발 각 단계별로 검증 기법을 소개하고 실제 사례 연구를 통해 품질 검증 관련 실무 능력을 배양한다. Inspection, Walkthrough 등의 Static Testing 기법과 소스 코드에 적용되는 Dynamic Testing 기법 등을 소개하며, 소프트웨어 시험의 각 단계별 Test 데이터 생성, 시험 수행 및 결과 보고서 작성을 하며, 이 과정에서 State-of-the-art 도구들을 사용한다.

### SEP525 소프트웨어 프로세스 (Software Process) 3:0:3

소프트웨어 프로세스 및 프로세스 개선의 기본 개념과 원리를 다룬다. ISO9000, ISO 12207, ISO 15504 등의 소프트웨어 품질 관련 표준과 CMM, Boothtrap, Trillium 등의 소프트웨어 프로세스 개선 모델 등에 대해 알아보고, 향후 소프트웨어 품질 및 프로세스 평가 및 개선 기법의 발전 방향을 소개한다.

### SEP526 소프트웨어 제품라인 공학 (Software Product Line Engineering) 3:1:3

소프트웨어 제품라인 공학은 유사한 일군의 소프트웨어 제품들을 동시에 개발하는 기술로 다양한 고객의 니즈에 맞게 특성화된 소프트웨어 제품이 경쟁력을 갖는 소프트웨어 시장 환경에서 필수적인 기술로 알려져 있다. 그럼에도 불구하고 우리나라에서는 아직 낮은 제품라인 기술수준으로 인하여 소프트웨어 경쟁력 확보에 어려움을 겪고 있다. 이 과목은 소프트웨어 제품라인 공학의 최신 기술과 이론을 학습하고 이를 적용해 봄으로써, 현장에서 소프트웨어 제품라인 개발에 활용하고 기술을 리드할 수 있는 능력개발을 목표로 한다.

- SEP527 이동 소프트웨어 개발 (Mobile Software Development) 3:0:3**  
 모바일컴퓨팅 환경에서 발생하는 소프트웨어의 특징에 대한 전반적 해석을 수업의 목표로 한다. 일차적으로 모바일장치의 컴퓨터 구조적 특성을 살펴보고, 이에 따른 구성이 되는 운영체제의 특이점을 분석한다. 이러한 모바일환경이 응용프로그램에 미치는 영향을 살펴본다. 예제적으로 응용프로그램이나 응용서비스에 대한 설계를 진행한다.
- SEP531 정보검색 및 자연어처리 (Information Retrieval and Natural Language Processing) 3:1:3**  
 본 과목에서는 정보검색과 자연어처리에 대한 전통적 및 근대적 기법을 공부하며, 실무에서 성공적으로 사용되고 있는 기계학습 기반의 언어자원을 소개한다.
- SEP532 인공지능 이론과 실제 (Theory and Practice for Artificial Intelligence) 3:1:3**  
 인공지능 및 기계학습 시스템을 구축하고 현장에서 운용하기 위한 필수 이론 및 실제 기술을 습득하는 과목이다. 최신 기계학습 및 딥러닝에 등장하는 필수 수학 지식을 배우고 이에 기반한 소프트웨어를 구현하여 클라우드 환경에서 실습하는 숙제와 프로젝트를 수행한다.
- SEP533 지식표현과 의미론 (Representation of Knowledge and Semantics) 3:1:3**  
 시맨틱웹은 차세대 웹 기술의 원형으로 주목되고 있으며, 온톨로지 기술의 발전에 따라, 그 실현 가능성이 높아지고 있다. 본 과목에서는 온톨로지의 기초 개념 및 정의, 지식표현과 온톨로지 모델링, 온톨로지 합성 및 매핑 등 온톨로지 연산처리, 온톨로지 버전 등 관리기법, 온톨로지 개발 방법론, 온톨로지 개발 도구, 온톨로지와 시맨틱웹, 온톨로지 응용시스템 구현 사례 등 온톨로지 공학의 전반적인 기초 개념과 이론을 소개한다.
- SEP537 소프트웨어 시스템 모델링 (Models of Software Systems) 3:0:3**  
 본 과목에서는 소프트웨어 시스템을 형식언어로 모델링하는 방법, 사용자가 원하는 시스템의 성질을 검증할 수 있는 모델을 만드는 방법, 그리고 사용자가 요구하는 성질들을 모델들이 갖고 있는지를 엄밀하게 검증할 수 있는 방법을 공부한다. 또한, 팀별 Lab을 통하여 이론과 실습을 연결한다. Lab의 최종 결과물을 제출하기에 앞서 중간발표와 강사와의 질의 응답하는 시간을 통하여, Lab 수행 방향의 올바름을 확인한다.
- SEP539 소프트웨어 개발방법론 (Methods: Deciding What to Design) 3:0:3**  
 성공적인 소프트웨어 개발을 위해서는 실제 해결해야 하는 문제와 개발할 소프트웨어 시스템간의 간격을 줄이기 위한 효과적인 방법들에 대한 이해가 필수적이다. 이 과목에서는 소프트웨어 개발 문제를 분석하고 이해하기 위한 다양한 방법들을 학습하고 문제 해결방안의 도출을 위한 다양한 요건과 제약사항들에 대해 이해할 수 있는 능력을 기르도록 한다.
- SEP542 기업 시스템 통합 (Enterprise System Integration) 3:0:3**  
 기업 정보 시스템의 구조, 특징 및 구성 요소들을 소개하고, 기업 내 / 기업 간 애플리케이션 통합 및 비즈니스 프로세스 통합을 위한 기본 개념, 제반 계산 기법 및 관련 표준에 관하여 교육한다. 특히, 주요 기업 통합 시나리오들을 Application Server를 이용해 직접 구현해 봄으로써, 기업 통합 기술의 작동 원리 및 이에 관한 실무 능력을 배양할 수 있도록 한다.
- SEP543 컴퓨팅 보안 (Computing Security) 3:0:3**  
 암호학, 네트워크 보안, 인증 기술 등 컴퓨터 보안의 여러 가지 요소기술에 대한 소개와 더불어 보안제품의 개발 및 평가는 어떻게 이루어지고 있는가를 다룬다. 또한 정보보호를 위한 관리 및 정책에 관한 기술 및 인터넷 보안을 위한 기술 (예: VPN, SSL, PGP, 스마트카드 등)에 대해 공부한다.
- SEP544 인터넷 서비스 및 인프라 (Internet Service and Infra) 3:0:3**  
 인터넷 컴퓨팅이란 인터넷상에서 발생하는 모든 종류의 컴퓨팅을 포함하며, 이는 곧, 현재 컴퓨터 공학의 거의 모든 분야가 이에 해당함을 의미한다. 본 과목은, 이 중에서, 현재 Core Internet Computing을 구성하는, GRID Computing, Web-Centric Applications, Multimedia Streaming, Networking Middleware 등을 심도 있게 다룬다.
- SEP545 데이터베이스 설계 (Database Design) 3:1:3**  
 본 과목은 데이터베이스 설계의 전반적인 방법론은 소개한다. 특히 데이터 모델링과 아울러 프로세스 모델

링을 취급한다. 데이터모델에 독립적인 설계 방법을 다루고, 그 결과적인 설계를 특정 데이터모델에 해당하는 설계로 변환한다. 학생들은 데이터베이스 설계의 여러 단계에 대하여 프로젝트를 통하여 실습할 기회를 갖는다.

**SEP547 IT 서비스공학 (IT Service Engineering) 3:0:3**

최근 KAIST 졸업생들이 IT서비스(SI 및 컨설팅) 업계에 진출하거나 기업 내 IT추진 부서에 취업하는 사례가 많아지고 있으나 현업에서 직접 필요로 하는 교육 프로그램이 미진한 실정이다. 1) 기업정보시스템을 기획/구축/운영하는 최신 IT서비스의 이해, 2) IT서비스 산업 및 기업 내 IT서비스 조직의 이해, 3) IT서비스 전략/엔지니어링/매니지먼트의 개념 정립, 4) 제반 IT서비스 프로세스의 방법론 습득, 5) 사례를 통한 Best Practice의 습득 등을 목표로 한다.

**SEP561 내장형 컴퓨팅 (Embedded Computing) 2:3:3**

임베디드 시스템은 많은 곳에서 다량으로 활용되며 시스템 특성에 의한 새로운 복잡도 때문에 새로운 설계 기법이 필요하며 기존의 어셈블리 프로그래밍과 논리설계가 아닌 상위 수준의 도구와 하드웨어/소프트웨어의 협정이 주요 내용이 된다. 본 강좌에서는 이 모든 것을 통합한 설계방법을 다루며 추세와 현안들을 소개한다. 단일목적에 위한 프로세서와 일반 프로세서를 이용한 소프트웨어의 설계와 활용방법을 소개하며 시스템의 메모리, 버스, 등의 개념을 다룬다. 그리고 보다 진보된 계산 유형과 제어 시스템, 반도체 기술, 그리고 설계도구등을 소개한다.

**SEP562 내장형 시스템 소프트웨어 (Embedded System Software) 3:0:3**

내장형 시스템을 위해 시스템 소프트웨어의 측면에서 지원되어야 할 내용들을 이론적인 배경과 실험을 통해 공부한다. 내장형 시스템을 위한 운영체제의 기능과 구조, 실시간 스케줄링 방법, 디바이스 드라이버의 작성, 전력소모 감소를 위한 방법 등에 대해 살펴본다.

**SEP564 내장형 운영체제 (Embedded Operating System) 3:0:3**

본 과목의 목표는 내장형 운영체제를 설계, 개발하기 위해 요구되는 운영체제 개념과 구현 능력을 제공하는 것이다. 내장형 운영체제를 위한 부트로더, 프로세스 관리, 메모리 관리, 입출력 장치 관리 및 파일 시스템 등에 대해 살펴본다.

**SEP565 모바일 소프트웨어 (Mobile Software) 3:0:3**

본 과목은 내장형 시스템에서 사용되는 네트워크 관련 기술들을 심도 있게 다루는 것을 목표로 한다. 내장형 시스템의 네트워크 구축에 필요한, Ethernet, IEEE 1394, Infrared communications, Wireless LAN, TCP/IP 프로토콜 스택 구현 등의 기술을 강의 및 실험을 통해 살펴본다.

**SEP581 소프트웨어 프로젝트 관리 및 실무 (Project Management and Professional Practice) 3:0:3**

소프트웨어 프로젝트 관리, group dynamics/psychology, 의사소통 및 발표 능력, 직업의식, 직업윤리, 및 실무에 필요한 기본적인 개념과 지식을 다룬다. 실제 소프트웨어 프로젝트 관리에서 발생할 수 있는 상황을 소개하고 이에 대한 대비 능력을 배양하며 관리자로서 기술적 역할 및 윤리 의식에 대하여 함께 다룬다.

**SEP582 정보화 조직 경영 및 리더쉽 (IT Organization and Leadership) 3:0:3**

본 과목에서는 기업정보화 책임자인 CIO의 역할과 책임, 정보화 조직관리 및 경영, 정보기술의 운영, 정보화 동향분석 등 정보화 조직경영능력을 배양하는 것을 목적으로 한다. 주요내용으로는 CIO의 비전과 역할, 정보시스템 조직관리, 벤처기업과 정보화, 정보화를 통한 경영혁신, 기업정보화 전략, 정보화 리더쉽 등 제반 관련 분야를 종합적으로 공부한다. 또한 성공적인 기업체의 CIO, 벤처기업의 CEO/CIO, SI업체의 책임 컨설턴트 등을 연사로 초빙하여 현장실무 경험을 듣는다.

**SEP583 소프트웨어 산업과 에코 시스템(Software Industry and Ecosystem) 3:0:3**

본 과목에서는 수강생들은 소프트웨어 산업의 구조 및 특징, 소프트웨어 생태계, 소프트웨어 서비스 산업과 제품산업을 포함한 국제 IT 산업 환경을 배우게 된다. 현재의 기술과 전망, 요구되는 소프트웨어 능력과 기술, 소프트웨어 인력의 양성 및 소프트웨어 산업의 발전을 위한 국가적인 정책, 관련 법, 소프트웨어 플랫폼의 역사, 지적 재산권 및 공개소프트웨어 등을 포함한다. IT 기술이 전통 산업의 경쟁력을 신장하기 위하여 어떻게 쓰이고 있는가에 대한 조사 수행 등을 통하여 IT기술을 산업계 현장에서 적용하는데 도움을 주고자 한다.

- SEP584 소프트웨어 기업 경영 (Management of Software Business) 3:0:3**  
 먼저 SW 산업의 작동 원리와 이러한 원리를 이해하는데 기반이 되는 경제 이론에 대해 살펴 본 뒤 SW 기업의 경영 전략에 대해 심도 있게 논한다. 또한 현재까지 성공하고 실패한 SW기업의 사례를 분석하며 최종적으로 SW 기업 창업 및 경영 시 고려해야 할 사항에 대해 분야별로 세부적으로 논한다.
- SEP591 소프트웨어 개발관리 (Managing Software Development) 3:0:3**  
 본 교과목에서는 소프트웨어 엔지니어에게 필요한 프로젝트 팀 리더, 소프트웨어개발과 프로젝트 엔지니어링과의 관계, 비용과 일정의 예측, 소프트웨어 프로세스 이해 등의 프로젝트 관리에 필요한 지식과 기술을 습득한다.
- SEP592 소프트웨어 특강 (Special Topics in Software) 3:0:3**  
 빅 데이터와 데이터과학은 업계에서 많이 회자되고 있지만, 학문적 주제로는 아직 잘 정의되어 있지 않다. 본 과목에서는 “빅 데이터와 데이터과학이란 무엇인가?”라는 중심 질문에 대하여 사례를 바탕으로 탐구하여 보고자 한다.
- SEP601 소프트웨어 개발 스튜디오 I (Software Development Studio I) 0:9:3**  
 소프트웨어개발 스튜디오 과목에서 학생들은 소프트웨어공학의 기본 과목들을 통해 배운 개념과 원리를 실제 프로젝트를 수행하는데 적용하고 체험하게 된다. 학생들은 팀 단위로 전문 멘토의 지도를 받으면서 스튜디오 프로젝트를 수행하게 된다. 특히 매주 팀 단위 미팅을 통해 프로젝트 수행과정에서의 선택과 결정 사항 등에 대해 토의하게 된다. 소프트웨어개발 스튜디오 I에서 학생들은 프로젝트 관리 기법들을 체험하고 고객의 요구사항을 효과적으로 수집, 표현, 분석하기 위한 방법들을 익히게 된다.
- SEP602 소프트웨어 개발 스튜디오 II (Software Development Studio II) 0:9:3**  
 소프트웨어개발 스튜디오 과목에서 학생들은 소프트웨어공학의 기본 과목들을 통해 배운 개념과 원리를 실제 프로젝트를 수행하는데 적용하고 체험하게 된다. 학생들 & 팀 단위로 전문 멘토의 지도를 받으면서 스튜디오 프로젝트를 수행하게 된다. 특히 매주 팀 단위 미팅을 통해 프로젝트 수행과정에서의 선택과 결정 사항 등에 대해 토의하게 된다. 소프트웨어개발 스튜디오 II에서 학생들은 고객의 요구사항을 소프트웨어 설계에 반영하고 분석하기 위한 방법들을 익히게 된다.
- SEP603 소프트웨어 개발 스튜디오 III (Software Development Studio III) 0:27:9**  
 소프트웨어 개발 스튜디오 과목에서 학생들은 소프트웨어공학의 기본 과목들을 통해 배운 개념과 원리를 실제 프로젝트를 수행하는데 적용하고 체험하게 된다. 학생들은 팀 단위로 전문 멘토의 지도를 받으면서 스튜디오 프로젝트를 수행하게 된다. 특히 매주 팀 단위 미팅을 통해 프로젝트 수행과정에서의 선택과 결정 사항 등에 대해 토의하게 된다. 소프트웨어 개발 스튜디오 III에서 학생들은 설계한 소프트웨어의 실제 개발과 시험을 수행하게 된다.
- SEP608 소프트웨어 아티팩트 분석 (Analysis of Software Artifact) 3:0:3**  
 분석(analysis)은 대상의 속성을 알아내기 위해 아티팩트를 체계적으로 검사하는 것이다. 본 수업에서는 소프트웨어 아티팩트(주로 코드) 분석, 디자인, 아키텍처 및 테스트 스위트 분석에 중점을 둔다. 또한, 학습한 아이디어를 실제 아티팩트에 적용하기 위해 이론적 토론과 실습을 병행한다.
- SEP609 소프트웨어 아키텍처 (Architecture for Software Systems) 3:0:3**  
 본 과목에서는 소프트웨어 아키텍처의 기본개념과 설계원리, 모델링과 분석 원리, 다양한 소프트웨어 아키텍처 모델링 언어 및 도구들을 공부하며, 이들을 실제세계의 문제에 적용하는 실습을 수행한다.
- SEP701 소프트웨어 개발 프로젝트 (1) (Software Development Project (1)) 1:15:6**  
 본 과목은 입학 1년 이후 (또는 교과목 21 학점 이수 후) 신청할 수 있으며 본 과정에서 숙지한 지식을 실제 소프트웨어 시스템 개발에 적용하여 실무를 경험하는 과목으로써 적절한 규모의 실질적인 소프트웨어 개발을 요하는 문제를 정의하고, 이를 분석하고, 설계하여 발표한다. 각 그룹 또는 학생별로 수행하여야 할 개발 활동과 제출하여야 하는 결과물을 정하여 제안서 형식으로 발표한다. 이 제안서 발표 시, 관련 분야의 교수진으로 위원회를 구성하여 Pass/Fail을 결정한다. Pass시 SEP702 소프트웨어 개발 프로젝트(2)를 신청할 수 있다.

**SEP702 소프트웨어 개발 프로젝트 (2) (Software Development Project (2))** **1:15:6**

본 과목은 SEP701 소프트웨어 개발 프로젝트 1을 이수한 후 신청할 수 있으며 SEP701에서 제안한 내용을 구현 및 시험하여 그 결과를 시연 및 발표한다. 각 그룹 또는 개인별로 학생들이 제안한 개발 활동을 개발 단계별 과정에 지도 교수가 참여하여 전체 프로젝트 진행을 안내한다. 프로젝트 결과 최종 발표 시, 관련 분야의 교수진으로 위원회를 구성하여 Pass/Fail을 결정한다.

**SEP965 개별연구 (Independent Study in M.S.)** **0:6:1**

타 교과목에서 배운 이론을 토대로 하여, 실험적인 면에 적용하는 능력을 배양하기 위한 실험 및 성능평가 위주의 과목으로 학생들에게 개별적인 프로젝트를 부여하여 해결 하도록 한다. 수강 신청 전에 담당교수와 의 상의를 거쳐야 한다.

**SEP966 세미나 (석사) (Seminar)** **1:0:1**

최근의 연구 활동 및 앞으로의 연구방향에 대하여 내·외부의 전문가들을 초청하여 강의를 듣고 관심사항들에 대하여 토론을 한다.