

# 소프트웨어융합대학 교육과정

## 대학소개

- 소프트웨어융합대학은 4차 산업혁명을 주도할 실천적 소프트웨어 인재 양성을 위해서 컴퓨터공학과와 소프트웨어융합학과의 2학과 멀티 트랙의 체제로 2018년도 설립한다.
- 소프트웨어융합대학의 STEP(Software Talent Enhancement Program)에 기반 한 심화 소프트웨어 전문가 양성 교육을 컴퓨터공학과에서 수행한다. 이는 산업계 및 글로벌 스탠다드에 준하는 실습과 프로젝트 교육 강화 등을 통해서 빅데이터, 모바일/보안, IoT(Internet of Thing), 스마트 콘텐츠 분야의 소프트웨어 전문가를 양성하는 교육을 수행한다.
- 소프트웨어융합대학의 SWING(SoftWare Industry for Next-Generation)에 기반 한 소프트웨어 신산업 융합 전문가 양성 교육을 소프트웨어융합학과에서 수행한다. 이를 소프트웨어 핵심과 융합 목표 전공을 위한 압축 교육을 통해서 게임 콘텐츠, 데이터 사이언스, 미래자동차·로봇 및 미래를 스스로 설계하고 기획해 갈 수 있는 글로벌 리더 양성을 통한 소프트웨어 신산업 융합 전문가를 양성하는 교육을 수행한다.

## 1. 교육목적

창의력과 전문성을 갖추고 새로운 산업과 학문을 주도할 실천적 소프트웨어 인재 양성

## 2. 교육목표

- 1) 소프트웨어 중심 기업/학계 및 소프트웨어 관련 창업을 주도하기 위하여, 심화 소프트웨어 전문가 교육을 수행함으로써 소프트웨어 분야 핵심 개발 능력을 보유한 인재를 육성
- 2) 융합 목표 분야 관련 기업 및 융합 창업을 주도하기 위하여, 소프트웨어 신산업 융합 전문가 교육을 수행함으로써, 융합분야에 특화된 전공지식과 소프트웨어 개발 능력을 보유한 인재를 육성

## 3. 설치학과

- 가. 컴퓨터공학과  
나. 소프트웨어융합학과

## 4. 대학 졸업 요건

학과명	프로그램명	졸업 이수 학점	단일전공과정					다전공과정					부전공과정1			부전공과정2		
			전공학점				타 전공 인정 학점	전공학점				타 전공 인정 학점	부전공과정1			부전공과정2		
			전공 기초	전공 필수	전공 선택	계		전공 기초	전공 필수	전공 선택	계		전공 필수	전공 선택	계	전공 필수	전공 선택	계
컴퓨터 공학과	컴퓨터공학	140	18	48	30	96	12	12	27	15	54	-	15	6	21	24	6	30
소프트웨어 융합학과	소프트웨어융합학	130	15	37	36	88	12	15	21	24	60	-	18	12	30	12	9	21

※ 교양이수는 후마니타스칼리지 교양교육과정을 따라야 함

※ 전공이수는 각 전공별 교육과정 시행세칙에서 정한 졸업이수요건을 만족해야 함

※ 부전공과정2는 이수해야할 교과목이 지정되어있으므로, 각 학과 교육과정 시행세칙에서 정한 졸업이수요건을 만족해야 함

※ 2018학년도 이후 신입생(편입생, 순수외국인 제외)은 소프트웨어 기초지식 습득 및 마인드 함양을 위해, 각 학과에서 정한 SW교육을 이수해야 함

## 5. 전공별 교육과정 편성 교과목수

학과/프로그램명		편성 교과목								전공필수+ 전공선택 (B+C)	
학과명	프로그램명	전공기초 (A)		전공필수 (B)		전공선택 (C)		전공선택(교직) (D)			
		과목수	학점수	과목수	학점수	과목수	학점수	과목수	학점수		
컴퓨터공학과	컴퓨터공학	6	18	19	48	39	111	-	-	58	159
소프트웨어융합학과	소프트웨어융합학	5	15	14	37	75	194	-	-	94	244

※ 단기현장실습/장기현장실습 과목은 제외한 현황임

## 6. 졸업능력인증제

- \* 2016년 8월 이후 소프트웨어융합대학 졸업생은 TOEIC SPEAKING 5급 이상, OPIC II레벨 이상을 취득하거나 이에 준하는 공인영어성적(TOEIC 630점, New TEPS 271점, G-TELP54점(LEVEL 2), TOEFL(IBT) 72점 등)을 취득하여, 제출 기간 내 소프트웨어융합대학 행정실로 제출하여야 졸업능력인증을 Pass한다.
- \* 외국국적 외국인 특별자의 경우는 상기의 공인영어성적 또는 한국어능력시험 4급 이상을 취득하여야 한다.
- \* 수료자 중 취업자에 해당하는 경우, 졸업능력인증 면제심사를 요청할 수 있다.(2019년 8월 졸업사정대상자부터 적용)
  - 제출 기간 내 면제신청서 및 재직증명서, 4대보험 가입증명서를 소프트웨어융합대학 행정실로 제출

## 7. 소프트웨어융합대학 전공과목(영어강좌)이수안내

2008학번 이후 신입생은 전공과목의 영어강좌 3과목 이상 이수를 졸업요건으로 충족해야 하며, 편입학생의 경우 전공과목 영어강좌 1과목 이상을 이수해야 한다.

# 컴퓨터공학과 교육과정

## 학과소개

- 고도의 정보 산업 사회에서는 새로운 학문으로 컴퓨터공학이 근본이 되며 정보가 가장 중요한 자원이 될 것이다. 이들 정보를 수집하고 처리하는데 있어 컴퓨터공학 기술은 필수적이며 핵심적이다. 컴퓨터공학과에서는 정보산업에서 가장 핵심이 되는 빅데이터분석 분야, 모바일 및 보안 분야, IoT 분야, 스마트 콘텐츠 분야 등의 교육을 추구하고 있다. 프로그래밍 능력을 증진 시키는 프로그래밍언어 교육은 매우 중요하고도 기본적인이어서 이론 및 실습을 통해 철저히 교육하고 있다.
- 졸업 후에는 대학원에 진학하거나 유학을 가며, 국내의 수많은 정보통신, 컴퓨터, 인터넷, 멀티미디어, 게임, 애니메이션, 전자 상거래 관련기업, 산업체, 금융기관, 국공립 연구소에 진출할 수 있다. 그리고 벤처기업을 창업하는 졸업생이 점점 늘고 있다. 향후 정보화 사회가 본격적으로 도래할 것으로 예측됨에 따라 컴퓨터공학 전공자의 수요가 폭발적으로 증가될 것으로 예상되며, 창의력과 전문성을 가진 컴퓨터공학 전공자들이 국가경쟁력 제고에 큰 역할을 할 것으로 기대된다. 국가적, 사회적, 시대적 요구에 비추어 보아 컴퓨터공학 전공에서 교육하고 다루는 기술은 산업발전의 핵심이자 주체이며, 이러한 점에서 컴퓨터공학 전공의 향후 발전 전망은 매우 밝고 무한하다고 할 수 있다.

## 1. 교육목적

IT강국을 선도할 수 있는 창의력과 전문성을 갖춘 글로벌 컴퓨터공학 인재 양성

## 2. 교육목표

- 1) 컴퓨터공학 전문지식 습득과 정보화 사회에 대한 이해를 바탕으로 글로벌 사회에서 각 분야 리더로서 활동할 수 있는 능력 배양
- 2) 기초과학의 충실한 학습을 바탕으로 지식기반 사회에서 요구되는 창의적 능력 배양
- 3) 기술적 문제를 공식화하고, 첨단 공학 도구를 사용하여 실험을 하고 수행함으로써 당면 문제를 체계적으로 해결할 수 있는 능력 배양
- 4) 공학적 윤리의식을 갖추고 미래가치를 창출하고 산업발전을 선도할 수 있는 능력 배양

## 3. 학과별 교과목 수

학과명	구분	전공기초	전공필수	전공선택		전공과목
				산학필수	전공선택	
컴퓨터공학과	과목수	6	19	8	31	64
	학점수	18	48	18	93	177

※ 단기현장실습/장기현장실습 과목은 제외한 현황임

## 4. 컴퓨터공학과 졸업 요건

### 1) 교육과정 기본구조표

학과명	졸업 이수 학점	단일전공과정						다전공과정					부전공과정1			부전공과정2 (SW트랙)		
		전공학점					타 전공 인정 학점	전공학점				타 전공 인정 학점	부전공과정1			부전공과정2 (SW트랙)		
		전공 기초	전공 필수	전공선택 산학 필수	전공 선택	계		전공 기초	전공 필수	전공 선택	계		전공 필수	전공 선택	계	전공 필수	전공 선택	계
컴퓨터공학과	140	18	48	12	18	96	12	12	27	15	54	-	15	6	21	24	6	30

※ 교양이수는 교양교육과정을 따름

※ 전공이수는 컴퓨터공학과 교육과정 시행세칙에서 정한 졸업이수요건을 만족해야 함

※ 2018학년도 이후 신입생(편입생, 순수외국인 제외)은 소프트웨어 기초지식 습득 및 마인드 함양을 위해, 컴퓨터공학과에서 정한 SW교육을 이수해야 함.

### 2) 졸업논문

컴퓨터공학과와 '캡스톤디자인 2'를 이수하는 것으로 경희대학교 졸업을 위한 "졸업논문" 합격으로 인정한다. 단, "졸업논문(컴퓨터공학)"을 필히 수강 신청하여야 한다.

### 3) 졸업능력인증제

졸업능력인증제 pass는 졸업의 필수이며, 소프트웨어융합대학 졸업능력 인증제를 따른다.

# 컴퓨터공학과 교육과정 시행세칙

## 제 1 장 총 칙

- 제1조(학과 설치 목적) ① 본 시행세칙은 본교 학칙 제34조, 제36조, 제58조에 의거하여, 컴퓨터공학과와의 운영에 관한 사항을 규정함의 목적으로 한다.
- ② 글로벌 시대에 국제적으로 인정받을 수 있는 엔지니어의 배출을 목표로 소프트웨어중심대학 사업의 기준과 Washington/Seoul Accord의 기준을 준수하기 위함이며, 지식기반시대와 산업변화에 적극적으로 대응하기 위한 순환형 개선 시스템의 도입을 그 목적으로 한다.
- ③ 이에 따라, 컴퓨터공학과는 컴퓨터공학 분야의 공학지식의 습득과 응용을 거쳐 설계에 이르는 능력을 키우고 미래 공학현장 문제를 해결할 수 있는 공학도로 양성하기 위하여 2017학년도부터 소프트웨어중심대학 사업을 수행하며, 지식기반시대와 산업변화에 적극적으로 대응하고 이에 부합하는 공학교육을 위하여 순환형 교육개선 시스템을 도입하여 운영한다.
- ④ 소프트웨어중심대학 사업단은 과학기술정보통신부 및 정보통신기술진흥센터 (IITP)에서 주관하는 IT 교육 혁신으로 산업에 부흥한 인재 양성을 위한 사업이다. 컴퓨터공학과는 글로벌 시대를 맞이하여 국제적으로 인정받을 수 있는 전문인력 양성을 위해, 소프트웨어중심대학 사업에서 권장하는 커리큘럼을 반영한 컴퓨터공학을 설치·운영한다.

- 제2조(일반원칙) ① 컴퓨터공학과를 단일전공, 다전공, 부전공으로 이수하고자 하는 학생은 본 시행세칙에서 정하는 바에 따라 교과목을 이수할 수 있다.
- ② 교과목의 선택은 지도교수와 상의하여 결정한다.
- ③ 모든 교과목은 [별표1] 교육과정 편성표에 제시된 이수학년과 개설학기에 준해 이수할 것을 권장한다.
- ④ 전공과목의 선수과목은 [별표5]와 같으며, 선·후수과목의 체계를 준수하여 이수하여야 한다. 선·후수과목의 체계는 전산시스템에 반영되어있으며 수강신청시 자동으로 적용된다.

## 제 2 장 교양과정

- 제3조(교양과목 이수) 전문교양과목은 공학교육인증 교양과목과 본교 후마니타스 교양과정 기본구조표에서 정한 소정의 학점을 모두 만족하여야 한다.

## 제 3 장 전공과정

- 제4조(전공 및 트랙과목 이수) ① 컴퓨터공학과와 단일전공과정을 이수하고자 하는 학생은 [표1]에 명시된 전공기초, 전공필수, 산학필수, 전공선택 학점을 이수하여야 한다. 다전공자의 경우는 [표2]에서 지정한 전공기초, 전공필수 교과목을 반드시 이수해야 하고, 부전공자의 경우는 [표3]에서 지정한 전공필수, 전공선택 교과목을 반드시 이수해야 한다.
- ② 컴퓨터공학을 단일전공, 다전공, 부전공으로 이수하고자 하는 자는 본 시행세칙에서 지정한 소정의 전공이수학점을 이수하여야 하며, [별표2]에서 제시된 학년별 교육과정 이수체계를 따를 것을 권장한다.
- ③ 2008학번 이후 신입학생의 경우에는 전공과목의 영어강좌 3과목 이상을 이수해야 졸업요건이 충족되며, 편입생의 경우에는 전공과목의 영어강좌 1과목 이상을 이수해야 졸업요건이 충족된다. 단, 전공과목이라 함은 이수구분이 전공기초, 전공필수, 산학필수, 전공선택인 과목을 의미한다.
- ④ 단일전공과정 이수자는 타전공 교과목을 이수한 경우 전공선택(최대 12학점)으로 인정받을 수 있으며, 전공학점인정 타전공 교과목은 [별표4]와 같다.

[표1] 단일전공 전공과목 편성표

구분	교과목명		과목수
전공 기초 (18)	물리학 및 실험 1, 미분적분학, 선형대수, 확률 및 랜덤변수(EE), 이산구조, 미분방정식		6
전공 필수 (48)	디자인적사고(SWCON), 웹/파이선프로그래밍(SWCON), 객체지향프로그래밍, 논리회로(EE), 컴퓨터구조, 자료구조, 운영체제, 컴퓨터네트워크, 소프트웨어공학, 알고리즘분석, 데이터베이스, 오픈소스SW개발, IT기술영어 1/2/3, 캡스톤디자인 1, 캡스톤디자인 2, 졸업논문(컴퓨터공학), 기계학습		19
전공 선택 (30)	산학필수(12)	최신기술프로젝트 1·2, 최신기술콜로키움 1, 최신기술콜로키움 2(SWCON), SW스타트업비즈니스, SW스타트업프로젝트, 연구연수활동 1·2, 단기현장실습/장기현장실습	10
	공동 선택	신호와시스템(EE), 문제해결, 형식언어 및 컴파일러, 파일처리, 멀티미디어시스템, 시스템분석 및 설계, 프로그래밍언어구조론, 독립심화연구1·2	8
	빅데이터/ 인공지능분야	딥러닝, 실전기계학습, 인공지능, 빅데이터프로그래밍, 클라우드컴퓨팅, 데이터센터프로그래밍(SWCON), AI 네트워킹	7
	모바일 및 보안 분야	정보보호, 모바일프로그래밍, 웹서비스프로그래밍, 블록체인	4
	IoT 분야	리눅스시스템프로그래밍, IoT디지털시스템, IoT소프트웨어, 로봇소프트웨어, IoT 네트워크	5
	스마트콘텐츠 분야	영상처리, 컴퓨터그래픽스, 멀티미디어처리, 인간-컴퓨터상호작용, UI/UX프로그래밍, 컴퓨터비전	6

※ 산학필수는 단기현장실습/장기현장실습을 포함하여 최신기술프로젝트 1·2, 최신기술콜로키움 1·2, SW스타트업비즈니스, SW스타트업프로젝트, 연구연수 활동 1·2 중 12학점 이상을 이수하여야 한다.

[표2] 다전공 전공과목 편성표

구 분	교과목명	
전공기초 (12)	미분적분학, 선형대수, 확률 및 랜덤변수, 이산구조	
전공필수 (27)	객체지향프로그래밍, 자료구조, 컴퓨터구조, 운영체제, 컴퓨터네트워크, 소프트웨어공학, 알고리즘분석, 데이터베이스, 캡스톤디자인 2, 졸업논문	
전공선택 (15)	다전공 전공필수에 포함되지 않은 컴퓨터공학과 단일전공의 전공필수/전공선택 교과목	

※ 졸업논문을 제외한 모든 과목은 3학점임

※ 2020년 1학기부터 컴퓨터공학을 다전공으로 이수하는 학생의 경우, '객체지향프로그래밍'은 '객체지향프로그래밍 및 실습'으로 대체할 수 있으며, '자료구조'는 '자료구조 및 알고리즘'으로 대체할 수 있음

[표3] 부전공 전공과목 편성표

구 분	교과목명	
부전공과정 1	전공필수 (15)	웹/파이선프로그래밍, 객체지향프로그래밍, 자료구조, 운영체제, 알고리즘분석
	전공선택 (6)	부전공과정1 전공필수에 포함되지 않은 컴퓨터공학과 단일전공의 전공필수/전공선택 교과목
부전공과정 2 「SW트랙」	전공필수 (24)	웹/파이선프로그래밍, 디자인적사고, 컴퓨터구조, 운영체제, 데이터베이스, 알고리즘분석, 자료구조, 캡스톤디자인 1
	전공선택 (6)	웹서비스프로그래밍, 프로그래밍언어구조론

※ 부전공과정 2(SW트랙)의 대체교과목은 [별표7] 참조

제5조(타전공과목 이수) 타전공과목의 이수는 [별표4] 전공학점인정 타전공 교과목표에 명시된 과목에 대해서 전공 학점으로 인정한다.

제6조(산학필수 이수) 단기현장실습/장기현장실습을 포함하여 최신기술프로젝트 1·2, 최신기술콜로키움 1·2, SW스타트업비즈니스, SW스타트업프로젝트, 연구연수활동 1·2 중 12학점 이상의 산학필수 학점을 이수하여야 한다.

제7조(대체교과목의 지정) 컴퓨터공학과와 전공과목의 대체과목은 [별표6]과 같다.

제8조(지식창업 트랙 운영) ① 학생들이 스스로 진로를 설계하고 창업과 진로를 열어갈 수 있도록 사회적 문제의 인식과 창의적 문제해결 역량 강화를 위한 지식창업트랙을 운영한다.

② '지식창업 트랙' 신청자는 지식창업교양 6학점, 지식창업심화교양 6학점, 지식창업심화전공 6학점을 포함하는 18학점을 이수해야 한다. 지식창업트랙 교육과정은 [표4]와 같다.

[표4] 지식/창업트랙 교육과정편성표

구분(학점)		교과목명	이수 학점	이수구분	개설	주관 부서
지식 창업 교양	필수	- 창업과도전 (3) - 특허와지적재산권(3) - 아이디어에서제품까지(3)	6	* 교양 -배분이수교과 -자유이수교과	후마니타스 칼리지	지식 창업 교육 센터
지식 창업 심화 과정	창업 전공 선택	- 특허와창의적사고(3) - 지식재산권법의이해(3) - 창업과재무관리(3) - 창업전략과모의창업(3) - 지식재산창업(3) - B2B마케팅전략(3) - 비즈니스모델(3)	6	* 교양 -배분이수교과 -자유이수교과	후마니타스 칼리지	
		- SW스타트업비즈니스(3) - 캡스톤디자인 2(3)	6	* 전공 -전공선택 -전공필수 (최대 6학점 중복인정)	컴퓨터공학과	
이 수 학 점 계			18			

제9조(대학원 과목의 이수) ① 3학년까지의 평균 평점이 3.5 이상인 학생은 컴퓨터공학과 대학원 학과장의 승인을 받아 학부 학생의 이수가 허용된 대학원 교과목을 통산 6학점까지 이수할 수 있으며, 그 취득학점은 전공선택으로 인정한다.

② 또한, 학원 시행세칙에 따라 본교의 학사학위과정 재학 중 본교의 일반대학원에서 개설한 교과목을 이수하여 A학점 이상 취득한 경우에는 학사학위 취득에 필요한 학점의 초과분에 한하여 제 1항의 절차를 거쳐 6학점 이내에서 대학원 진학 시에 대학원 학점으로 인정받을 수 있다.

## 제 4 장 졸업이수요건

제10조(졸업이수학점) ① 컴퓨터공학전공은 최소 졸업이수학점은 140학점이다.

② 교양학점은 후마니타스 교양교육과정을 만족하여야 한다.

③ 졸업논문을 포함하여 해당 졸업이수요건을 충족시켜야 한다.

- 제11조(컴퓨터공학 졸업이수요건) ① 단일전공과정 : 컴퓨터공학과 학생으로서 단일전공자는 전공기초 18학점, 전공필수 48학점, 산학필수 12학점, 전공선택 18학점을 포함하여 전공학점 96학점 이상 이수하여야 한다.
- ② 다전공과정 : 컴퓨터공학과 학생으로서 타전공을 다전공과정으로 이수하거나, 타학과 학생으로서 컴퓨터공학전공을 다전공과정으로 이수하는 학생은 전공기초 12학점, 전공필수 27학점, 전공선택 15학점을 포함하여 전공학점 54학점 이상 이수하여야 한다.
- ③ 부전공과정 1 : 컴퓨터공학전공을 부전공과정 1로 이수하고자 하는 자는 전공필수 15학점을 포함하여 전공학점 21학점을 이수하여야 한다.
- ④ 부전공과정 2(SW트랙) : 컴퓨터공학전공을 부전공과정 2(SW트랙)로 이수하고자 하는 자는 [표3]에 명시된 전공필수 24학점, 전공선택 6학점을 이수하여야 한다.
- ⑤ 전공과목의 영어강좌 3과목 이상 이수를 졸업요건으로 충족해야 하며, 졸업능력인증제도를 따른다.

- 제12조(편입생 전공이수학점) ① 편입생은 전적대학에서 이수한 학점 중 본교 학점인정심사에서 인정받은 학점을 제외한 나머지 학점을 추가로 이수하여야 한다.
- ② 교과목의 선·후수 관련, 편입학생에 한하여 교과목 담당교수가 인터뷰를 통하여 선수과목의 필요 여부를 판단하여 선수과목 미이수 학생에 대하여 수강을 허용할 수 있다.

제13조(영어강좌 이수학점) 2008학번 이후 신입학생의 경우에는 전공과목의 영어강좌 3과목 이상을 이수해야 졸업요건이 충족되며, 편입생의 경우에는 전공과목의 영어강좌 1과목 이상을 이수해야 졸업요건에 충족된다. 단, 전공과목이라 함은 이수구분이 전공기초, 전공필수, 산학필수, 전공선택인 과목을 의미한다.

제14조(졸업논문) 컴퓨터공학과와 '캡스톤디자인2'를 이수하는 것으로 경희대학교 졸업을 위한 "졸업논문" 합격으로 인정한다. 단, "졸업논문(컴퓨터공학)"을 필히 수강 신청하여야 한다.

제15조(SW교양 졸업요건) 2018학년도 이후 입학생(편입생, 순수외국인 제외)은 SW교양 또는 SW코딩 교과목에서 총 6학점을 이수하여야 한다. SW교양 및 SW코딩 교과목 개설 및 운영에 관한 세부사항은 소프트웨어 교육교과운영시행세칙을 따른다.

## 제 5 장 기 타

제16조(학생상담) 학생상담의 시기, 방법 등 세부사항은 컴퓨터공학과와 내규를 따른다.

제17조(보칙) 본 내규에 정하지 않는 사항은 컴퓨터공학과 학과회의 의결에 따른다.

## 부 칙

제1조(시행일) 본 시행세칙은 2019년 3월 1일부터 시행한다.

- 제2조(경과조치) ① 2010학번 이후 학생 중 컴퓨터공학을 단일전공 또는 다전공하는 학생들은 응용과학대학 교육과정인 '물리학 1' 및 '물리학실험'을 '물리학 및 실험 1'로 대체 인정한다.
- ② 2011년 이전 입학생 중 컴퓨터공학전문 프로그램(ABEEK)은 단일전공과정과 다전공과정 졸업 이수 요건을 전공기초 21학점, 전공필수 33학점, 전공선택 27학점으로 변경한다.
- ③ 2012, 2013년 입학생 중 컴퓨터공학전문 프로그램(ABEEK)은 단일전공과정과 다전공과정 졸업 이수 요건을 전공기초 21학점, 전공필수 54학점, 전공선택 24학점으로 변경한다.
- ④ 2009년 이후 2017년 이전 입학생 중 컴퓨터공학 프로그램(일반형)은 단일전공과정과 다전공과정의 졸업 이수 요건을 전공기초 21학점으로 변경한다.



- ⑤ 제외된 전공기초 이수교과목(미분적분학 2, 물리학 및 실험 2, 일반화학, 일반생물) 대신 전공 교과목의 추가이수를 권장한다.
- ⑥ 2017학번 이전 학생 중 공학교육인증프로그램(ABEEK)을 이수중인 학생은 본인의 졸업요건을 준수하며, [별표1]의 과목 중 설계학점 12학점을 이수해야 한다.
- ⑦ 2017학번 이전 학생 중 공학교육인증프로그램(ABEEK) 이수를 포기하고자 하는 학생은 공학교육인증지원시스템에서 프로그램 포기 신청 후 '공학인증프로그램 이수포기' 신청서를 출력하여 프로그램 PD 및 프로그램 운영위원회의 심의를 거쳐 포기를 할 수 있다.
- ⑧ 2017학번 이전 학생 중 전공필수인 '시스템분석 및 설계'를 수강하지 않은 학생은 '소프트웨어공학' 이수를 전공필수로 인정할 수 있다.
- ⑨ 2017년도 이전 졸업자 중 컴퓨터공학전문 프로그램(ABEEK)의 졸업이수요건은 [표5]을 적용하고, 컴퓨터공학 프로그램(일반형)의 졸업이수요건은 [표6]을 적용하며, 교과목 변경으로 인한 대체과목 일람은 [별표6]을 적용한다.
- ⑩ 부전공과정2(SW트랙)의 대체교과목은 [별표7]을 적용한다.
- ⑪ 2017년도 이전 입학생의 경우, 각 학년별 선수과목 대신 [별표5]의 선수과목 지정표에 따른 선수과목을 따를 수 있다.
- ⑫ 2017년도 이전 학생 중 컴퓨터공학전문 프로그램(ABEEK)의 '현장연수활동'은 '현장실습', '단기현장실습' 또는 '장기현장실습'으로 '현장연수활동' 이수를 인정할 수 있다. 관련하여 '현장실습시행세칙'에 따른다.
- ⑬ 2017, 2018년도 입학한 학생들의 졸업요건인 TOPCIT 최소 1회 이상 응시는 소급적용하여 삭제한다. 단, 본인의 IT역량 확인을 위해 주기적인 TOPCIT 응시를 장려한다.
- ⑭ 컴퓨터공학과를 다전공으로 이수하고자 하는 학생은 응용과학대학 교육과정인 '미분적분학 1'을 '미분적분학'으로 대체 인정한다.
- ⑮ 컴퓨터공학과에 전과한 학생은 응용과학대학 교육과정인 '미분적분학 1'을 '미분적분학'으로 대체 인정한다.
- ⑯ 2019년 2월 이후 졸업자가 2018년도 이후 컴퓨터공학과 개설교과목을 수강하면 해당 교과목은 컴퓨터공학 전문 프로그램(ABEEK)으로 인정한다.
- ⑰ 컴퓨터공학과를 단일전공으로 이수하고자 하는 2017년도 이전 입학생이 [별표4]의 전공학점인정 타전공 교과목표의 교과목을 수강한 경우 전공선택(최대 12학점)으로 인정받을 수 있다.
- ⑱ 컴퓨터공학과 전공기초 교과목은 타 단과 대학의 다음의 [별표 8]의 유사 교과목으로 대체 가능함.

[표5] 입학년도에 따른 컴퓨터공학전문 프로그램(ABEEK) 졸업이수 요건표

입학년도	졸업이수학점	전공이수학점				타 전공 인정학점
		전공기초	전공필수	전공선택	합계	
2007년	130	21	33	27	81	12
2008 - 2011년	136	21	33	27	81	12
2012 - 2017년	143	21	54	24	99	12

[표6] 입학년도에 따른 컴퓨터공학 프로그램(일반형) 졸업이수 요건표

입학년도	졸업이수학점	단일전공과정				다전공과정				타 전공 인정학점
		전공기초	전공필수	전공선택	합계	전공기초	전공필수	전공선택	합계	
2004 - 2005년	130	15	15	34	64	15	15	34	64	12
2006 - 2007년	130	15	15	39	69	15	15	39	69	12
2008년	136	15	15	39	69	15	15	39	69	12
2009 - 2011년	136	21	15	39	75	21	15	39	75	12
2012 - 2014년	136	21	42	12	75	21	24	12	57	12
2015 - 2017년	130	21	42	12	75	21	24	12	57	12
2018년	140	18	45	33	96	12	27	15	54	12

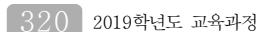
[별표1]

## 컴퓨터공학과 교육과정 편성표

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		부전공1	부전공2 (SW트랙)	P/F 평가	비고
					이론	실기	실습	설계		1학기	2학기				
1	전공 기초	물리학 및 실험 1	APHY1002	3	2		2		1	○					
2		미분적분학	AMTH1009	3	3				1	○					
3		선형대수	AMTH1004	3	3				1	○					
4		확률 및 랜덤변수	EE211	3	3				2	○	○				
5		미분방정식	AMTH1001	3	3				1		○				
6		이산구조	CSE201	3	3				2	○	○				
7	전공 필수	디자인적사고	SWCON103	3				3	1	○	○	○	○		
8		웹/파이선프로그래밍	SWCON104	3	2		2		1	○	○	○	○		
9		객체지향프로그래밍	CSE103	3	2		2		1	○	○	○			
10		논리회로	EE209	3	3				2	○	○	○			
11		컴퓨터구조	CSE203	3	3				2	○	○	○	○		
12		자료구조	CSE204	3	2		2		2	○	○	○	○		
13		운영체제	CSE301	3	3				3	○	○	○	○		
14		컴퓨터네트워크	CSE302	3	3				3	○	○	○			
15		소프트웨어공학	CSE327	3	3				3	○	○	○			
16		알고리즘분석	CSE304	3	2		2		3	○	○	○	○		
17		데이터베이스	CSE305	3	3				3	○	○	○	○		
18		오픈소스SW개발	CSE223	3				3	2	○	○	○			
19		IT기술영어 1	CSE208	1	1				2		○	○		○	
20		IT기술영어 2	CSE308	1	1				3	○		○		○	
21		IT기술영어 3	CSE309	1	1				3		○	○		○	
22		캡스톤디자인 1	CSE406	3				3	4	○	○	○	○		
23		캡스톤디자인 2	CSE405	3				3	4	○	○	○			
24		졸업논문	CSE403	0					4	○	○	○		○	
25		기계학습	SWCON202	3	3				2	○	○				
26	전공 선택	신호와시스템	EE210	3	3				2	○	○	○			
27		문제해결	CSE321	3	2		2		3		○	○			
28		형식언어 및 컴파일러	CSE322	3	3				3	○		○			
29		파일처리	CSE323	3	3				3	○		○			
30		멀티미디어시스템	CSE324	3	3				3	○		○			
31		시스템분석 및 설계	CSE303	3	3				3		○	○			
32		프로그래밍언어구조론	CSE328	3	3				3		○	○	○		
33		SW스타트업비즈니스	CSE330	3	3				3	○		○			

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		부전공1	부전공2 (SW트랙)	P/F 평가	비고
					이론	실기	실습	설계		1학기	2학기				
34		SW스타트업프로젝트	CSE334	3				3	3		○	○			
35		최신기술프로젝트 1	CSE436	3				3	4	○		○			
36		최신기술프로젝트 2	CSE437	3				3	4		○	○			
37		최신기술콜로키움 1	CSE438	2	2				4	○		○		○	
38		최신기술콜로키움 2	SWCON302	2	2				2-4		○	○		○	
39		단기현장실습		*1) 3/6			6/12		2-4	○	○	○			
40		장기현장실습		*1) 9/12			18/24		2-4	○	○	○			
41		연구연수활동 1	CSE496	1			1		2-4	○		○			
42		연구연수활동 2	CSE497	1			1		2-4		○	○			
43		독립심화학습 1 (컴퓨터공학과)	CSE495	3			3		3	○		○		○	
44		독립심화학습 2 (컴퓨터공학과)	CSE499	3			3		3		○	○		○	
45		인공지능	CSE338	3	3				4	○		○			
46		빅데이터프로그래밍	CSE434	3	3				3		○	○			
47		클라우드컴퓨팅	CSE335	3	3				3	○		○			
48		데이터센터프로그래밍	SWCON221	3	2		2		2	○	○	○			
49		정보보호	CSE423	3	3				4	○		○			
50		모바일프로그래밍	CSE435	3	3				4		○	○			
51		웹서비스프로그래밍	CSE433	3	3				4	○		○	○		
52		리눅스시스템프로그래밍	CSE332	3	2		2		3		○	○			
53		IoT디지털시스템	CSE439	3	2		2		4		○	○			
54		IoT소프트웨어	CSE440	3	2		2		4	○		○			
55		로봇소프트웨어	CSE427	3	2		2		4		○	○			
56		영상처리	CSE426	3	3				4	○		○			
57		컴퓨터그래픽스	CSE428	3	3				4	○		○			
58		멀티미디어처리	CSE430	3	3				4		○	○			
59		인간-컴퓨터상호작용	CSE431	3	3				4		○	○			
60		UI/UX프로그래밍	CSE224	3	3				2	○		○			
61		컴퓨터비전	CSE441	3	3				4		○	○			
62		블록체인	CSE442	3	3				4		○				
63		AI 네트워크	CSE443	3	3				4	○					
64		IoT 네트워크	CSE444	3	3				4	○					
65		답러닝	CSE339	3	3				3	○	○				
66		실전 기계학습	CSE340	3				3	3	○	○				

1) 단기현장실습/장기현장실습은 각 활동별로 3학점, 6학점, 9학점, 12학점을 산학필수 학점으로 이수함



[별표3]

# 권장 이수 가이드라인

1 학 년		2 학 년		3 학 년		4 학 년	
1 학 기	2 학 기	1 학 기	2 학 기	1 학 기	2 학 기	1 학 기	2 학 기
<div>컴퓨터공학과 권장 이수 가이드라인</div> <div> <div>전공기초</div> <div>전공필수</div> <div>교양</div> <div>교양</div> </div>		<div>신학필수 12 학점</div> <ul style="list-style-type: none"> <li>1개월 인턴십 2회 (방학 중)</li> <li>신학필수 2과목 이수 (학기 중)</li> </ul>					
후마니타스 교양1	후마니타스 교양2	물리학및실험1	미분적분학	선형대수	웹/파이선프로그래밍		
후마니타스 교양3	후마니타스 교양4	후마니타스 교양5	미분방정식	디자인적 사고	객체지향프로그래밍		
후마니타스 교양6	공통선택1	핵심 및 랜덤변수	이산구조	오픈소스SW개발	논리회로		
IT기술영어1	후마니타스 교양7	후마니타스 교양8	공통선택2	자료구조	컴퓨터구조		
IT기술영어2	후마니타스 교양9	공통선택3	운영체제	컴퓨터네트워크	소프트웨어공학		
IT기술영어3	후마니타스 교양10	후마니타스 교양11	분야선택1	데이터베이스	알고리즘분석		
			분야선택2	분야선택3	캡스톤디자인1		
			분야선택4	졸업논문	캡스톤디자인2		

[별표4]

### 전공학점인정 타전공 교과목표

순번	과목개설학과명	과목코드	교과목명	학점	인정이수구분	비고
1	소프트웨어융합학과	SWCON207	수치해석프로그래밍	3	전공선택	
2	소프트웨어융합학과	SWCON211	게임프로그래밍입문	3	전공선택	
3	소프트웨어융합학과	SWCON212	게임PX디자인	3	전공선택	
4	소프트웨어융합학과	SWCON311	게임그래픽프로그래밍	3	전공선택	
5	소프트웨어융합학과	SWCON312	체감형기술이론 및 실습	3	전공선택	
6	소프트웨어융합학과	SWCON313	가상/증강현실이론 및 실습	3	전공선택	
7	소프트웨어융합학과	SWCON314	게임공학	3	전공선택	
8	소프트웨어융합학과	SWCON331	로봇프로그래밍	3	전공선택	
9	소프트웨어융합학과	SWCON3XX	게임콘텐츠캐릭터디자인	3	전공선택	
10	소프트웨어융합학과	SWCON3XX	로봇센서데이터처리	3	전공선택	
11	소프트웨어융합학과	SWCON425	데이터사이언스 및 시각화	3	전공선택	
12	소프트웨어융합학과	SWCON4XX	자연어처리	3	전공선택	
13	소프트웨어융합학과	SWCON491	게임프로그래밍을 통한 인공지능과 데이터분석	3	전공선택	
14	소프트웨어융합학과	SWCON492	소프트웨어융합네트워킹	3	전공선택	

[별표5]

## 컴퓨터공학과 선수과목 지정표

순번	학과명	교과목명(후수과목)			선수과목			비고
		학수번호	교과목명	학점	학수번호	교과목명	학점	
1	컴퓨터 공학과	CSE204	자료구조	3	CSE103	객체지향프로그래밍	3	
2		CSE203	컴퓨터구조	3	EE209	논리회로	3	
3		CSE304	알고리즘분석	3	CSE204	자료구조	3	
4		CSE406	캡스톤디자인 1	3	SWCON103	디자인적사고	3	
5		CSE405	캡스톤디자인 2	3	CSE406	캡스톤디자인 1	3	단일전공 이수자만 해당
6		CSE321	문제해결	3	SWCON104	웹/파이선프로그래밍	3	
7		EE210	신호와시스템	3	AMTH1009	미분적분학	3	
8		CSE223	오픈소스SW개발	3	SWCON104	웹/파이선프로그래밍	3	
9		CSE323	파일처리	3	CSE204	자료구조	3	
10		CSE424	IoT디지털시스템	3	CSE203	컴퓨터구조	3	
11		CSE433	웹서비스프로그래밍	3	SWCON104	웹/파이선프로그래밍	3	
12		CSE431	멀티미디어처리	3	CSE103	객체지향프로그래밍	3	

※ 우측 선수과목 수강 시에 좌측 후수과목 수강을 허용함

[별표6]

### 대체과목 일람표

순번	학과명	구 교과과정		현행 교과과정	
		교과목명	학점	교과목명	학점
1	컴퓨터공학과	미분적분학 1	3	미분적분학	3
2		기초공학설계	3	디자인적사고	3
3		프로그래밍기초	3	웹/파이선프로그래밍	3
4		고급객체지향프로그래밍	3	객체지향프로그래밍	3
5		컴퓨터공학개론	3	오픈소스SW개발	3
6		창의적설계 1	3	캡스톤디자인 1	3
7		창의적설계 2	3	캡스톤디자인 2	3
8		창의적설계	3	캡스톤디자인 1	3
9		창의적종합설계(컴퓨터공학)	3	캡스톤디자인 1	3
10		졸업연구	3	캡스톤디자인 2	3
11		설계프로젝트 A/B/C/D	12 (각 3학점)	최신기술프로젝트 1·2, 최신기술콜로키움 1·2, SW스타트업비즈니스, SW스타트업프로젝트 연구연수활동 1·2 (단, 단기현장실습/ 장기현장실습은 제외)	12
12		현장연수활동	1-3	단기현장실습/장기현장실습	3/6/9/12
13		현장실습	3-6	단기현장실습	3-6
14		머신러닝	3	딥러닝	3



[별표7]

부전공과정 2(SW트랙) 이수자를 위한 웹/파이썬프로그래밍 강좌 대체이수 인정 과목

순번	학과	교과목명
1	원자력공학과	공학프로그래밍입문
2	건축공학과	공학프로그래밍입문
3	물리학과	정보물리학
4	교양학과(배분이수교과 논리, 분석, 수량세계)	프로그래밍을통한논리적사유연습
5	교양학과(자유이수 자유이수 기타)	프로그래밍입문

[별표8]

### 전공기초 교과목 강좌 대체 이수 인정 교과목

순번	컴퓨터공학과 전공기초 교과목명	대체 인정 교과목
1	APHY1002 물리학 및 실험1	APHY100 물리학실험1(응용과학대학 응용물리학과) APHY1000 물리학1(응용과학대학 응용물리학과) PHYS1101 물리학및실험1 (이과대학 물리학과) 중 한 과목
2	AMTH2002 미분적분학1	MATH1101 미적분학및연습1 (이과대학 수학과)
3	AMTH1003 미분적분학2	MATH1102 미적분학및연습2 (이과대학 수학과)
4	AMTH1009 미분적분학	MATH1101 미적분학및연습1 (이과대학 수학과) MATH1102 미적분학및연습2 (이과대학 수학과) 중 한 과목
5	AMTH1001 미분방정식	MATH2411 미분방정식1 (이과대학 수학과) MATH2412 미분방정식2 (이과대학 수학과) 중 한 과목
6	AMTH1004 선형대수	MATH2111 선형대수학1 (이과대학 수학과) MATH2112 선형대수학2 (이과대학 수학과) 중 한과목

## 컴퓨터공학과 교과목 해설

- 확률 및 랜덤변수 (Probability and Random Variables)

전산학에서 응용할 수 있는 제반 기초 이론을 습득하고 실제적인 응용 확률통계와 통계 소프트웨어 패키지를 사용하는 방법을 익힌다.  
This course studies basic probability and random variable theories that can be used in the field of computer engineering, and also introduces statistics theory and software packages.

- 이산구조 (Discrete Structures)

수학적인 관점에서 논리적인 디지털 컴퓨터 구조를 이해하기 위해 형식논리, 알고리즘 증명, 재귀, 집합, 순열과 조합, 이항정리, 이진관계, 함수 및 행렬, 그래프, 트리, 그래프 알고리즘, 프로그램의 검증, 부울 대수와 컴퓨터 논리 등에 관하여 배운다.

In order to understand the logical structure of digital computer from mathematical viewpoints, this course is designed to learn formal logic, proof of algorithm, recursion, set, permutation and combination, binomial theorem, binary relation, function and matrix, graph, tree, graph algorithm, program verification, Boolean algebra, and computer logic.

- 디자인적사고 (Design Thinking)

새로운 소프트웨어 개발을 위한 디자인 중심의 창의적 설계 접근방법과 아이디어 발상을 위한 이론적 프로세스를 팀워크를 통해 학습 한다.

We will learn the design-oriented creative design approach for new software development and the theoretical process for idea development through teamwork.

- 웹/파이썬프로그래밍 (Web/Python Programming)

웹 프로그래밍과 파이썬 프로그래밍의 기초적인 내용을 배우도록 한다. 웹 프로그래밍은 HTML5/CSS3/Javascript를 사용하는 WebApp을 개발함으로써, 클라이언트 개발을 가능하게 한다. 아울러 Node.js를 통한 서버 프로그래밍까지 할 수 있도록 한다. 파이썬은 기초 문법에 대한 이해를 수행할 수 있도록 한다.

Learn the basics of Web programming and Python programming. Web programming enables client development by developing WebApp using HTML5 / CSS3 / Javascript. It also allows server programming through Node.js. Python makes it possible to understand basic grammar.

- 객체지향프로그래밍 (Object-Oriented Programming)

객체지향 프로그래밍 기초에서 배운 데이터 형, 입출력, 선택문, 반복문, 함수, 배열, 포인터, 문자열 등을 기본으로 하여 클래스, 함수 오버로딩, 연산자 오버로딩, 상속, 가상함수, 템플릿, 네임스페이스 등의 고급 객체지향 프로그래밍 기법을 배우고 이를 실습을 통해 익힌다.

Based on the basic knowledge of object-oriented programming such as data type, I/O, selection, iteration, function, array, pointer, string, etc., this course provides advanced techniques on object-oriented programming like class, function overloading, operator overloading, inheritance, virtual function, template, name space.

- 논리회로 (Logic Circuit)

디지털 논리회로의 기본요소인 논리소자 특성 이해 및 디지털 논리회로(조합회로, 순서회로)에 대한 설계방법을 익혀 실제적 응용 디지털 회로설계와 컴퓨터의 기본구조 설계에 관해 학습한다.

This course introduces design and implementation of digital logic circuits. By understanding of logic device property and design method in digital system, it focuses on basic design for computer architecture and practical digital circuit using combinational and sequential circuit.

- 컴퓨터구조 (Computer Architecture)

컴퓨터 구조 설계의 기초 이론으로서 기본적인 컴퓨터 시스템의 구성과 설계에 대한 개념과 기법을 소개한다. 데이터의 표시방법, 레지스터의 전송과 마이크로 동작, 컴퓨터 소프트웨어를 포함하여 연산장치, 제어장치, 입출력장치의 구조와 설계기법을 학습함으로써 컴퓨터를 설계할 수 있는 지식을 습득하고 명령포맷, CPU 내부구조, 하드 와이어드 제어에 의한 제어 유닛 설계, 마이크로프로그램 제어에 의한 제어 유닛 설계, 인터럽트, DMA 등에 의한 I/O 처리 기술을 배운다. 이 과목을 수강하기 전에 논리회로를 수강할 것을 권고한다.

Basic concepts of computer architecture and organization include data representation, register transfer, micro-operation, system software etc. This course studies on I/O interface techniques which include instruction format, CPU structure, control unit design using hardwired-control and micro-programmed control, interrupt, DMA.

- 자료구조 (Data Structures)

자료 추상화, 배열, 리스트, 스택, 큐, 트리, 그래프 등의 자료구조와 그러한 자료구조를 활용할 수 있는 알고리즘을 배운다. 이 과목을 통해서 학생들은 전산학의 지식을 확대하고 프로그래밍 기술을 향상시킬 수 있다.

This course focuses on data abstraction, data structures such as array, list, stack, queue, tree, graph and algorithms that utilize those data structures. From this course, the students can expand their knowledge of computer science and sharpen their programming skills.

- 운영체제 (Computer Operating System)

운영체제는 사용자 프로그램의 수행과 주변장치나 기억공간과 같은 다양한 자원 할당을 감시한다. 이 과목에서는 멀티프로그래밍, 시분할, 그리고 비동기적 프로세서의 개념을 소개한다. 특히 동기화, 스케줄링, 교착, 메모리관리, 가상메모리관리, 파일 시스템, 디스크 스케줄링, 정보공유, 보호/보안 및 분산운영체제와 같은 주제를 중점적으로 학습한다.

Operating systems monitor the execution of user programs and the allocation of various resources such as memory space and peripheral devices. In this class, we introduce the basic concepts of multiprogramming, timesharing and asynchronous processes. The course focuses on synchronization, scheduling, deadlock, memory management, virtual memory management, file system, disk scheduling, information sharing, protection and security, and distributed operating system.

- 컴퓨터네트워크 (Computer Networks)

컴퓨터 네트워크를 구성하는 각종 네트워킹 장치들의 계층 모델, 특성, 동작 방법, 그리고 운용 기술에 대하여 학습한다. 또한 이들 장치를 상호 연결한 인터넷네트워크의 구성과 동작 방법에 대하여 소개한다. 본 과목의 수강을 통하여 컴퓨터 네트워크의 구성과 동작 방법에 대하여 소개한다. 본 과목의 수강을 통하여 컴퓨터 네트워크의 7계층 구조와 인터넷 4계층 구조를 이해할 수 있고, 간단한 LAN(Local Area Network)을 설계할 수 있으며, 계층 모델을 기반으로 한 컴퓨터 네트워크의 이론적 이해 및 분석력을 함양함으로써 컴퓨터 네트워킹 개념에 대한 이론과 실용 기술을 체득할 수 있다.

This course deals with layered models, characteristics, operations and management of networking devices. Also, the course introduces to the internetworking among networked devices. Students through this course can understand about configuration of computer network and its operations. As a core architecture, this course deals with OSI 7 layers and 4 layered architecture for Internet. So, students can obtain the capability to design LANs through theoretical understanding and analytical learning.

- 소프트웨어공학 (Software engineering)

소프트웨어 공학 분야는 프로그램이 방대하고, 오랜 기간 동안 많은 프로그래머들이 참여하는 경우 발생하는 문제를 다룬다. 본 강좌에서 학습하는 분야는 프로그래밍 프로젝트의 설계와 구성, 시험과 프로그램 신뢰도, 소프트웨어 비용의 성격과 발생원인 인지, 여러 프로그래머간의 협조, 사용자 친화적 인터페이스 설계 및 문서화 등이다.

The field of software engineering deals with problems that arise when programs are large, when they involve many programmers, and when they exist over long periods of time. Topics will include organizing and designing a

---

programming project, testing, and program reliability, identifying the nature and sources of software costs, coordinating multiple programmers, documentations and design of friendly use interfaces.

- 알고리즘분석 (Algorithm Analysis)

알고리즘의 기본적인 이해를 하고 대표적인 알고리즘의 형태를 배운다. 알고리즘 방법을 divide-and-conquer, dynamic programming, greedy algorithms, branch-and-bound 등으로 분류하고, 각각의 특성을 이해하도록 한다. 아울러 기본적인 복잡도 문제를 살펴본다. 본 과목을 이수 후 새로운 문제에 대한 해결 방법을 도출할 수 있는 능력을 키운다.

Basic concepts of algorithms are presented and fundamental types of algorithms are discussed. Algorithms are classified into divide-and-conquer, dynamic programming, greedy algorithm, and branch-and-bound and properties of the algorithms are discussed. Also, fundamental complexity problems are introduced. Students will be able to enhance their capability to solve new problems through the course.

- 데이터베이스 (Database)

데이터베이스 시스템을 이루는 기본 구성 요소에 대한 이론을 소개하고, ER-모델 및 관계데이터 모델을 중심으로 한 데이터베이스 설계 이론과 동시성 제어, 회복, 트랜잭션 관리와 같은 데이터베이스 관리 시스템을 구현하는 이론을 소개한다.

This course is to provide the basic understanding about database systems and introduce database design techniques based on ER-model and relational data model. It also deals with theoretical issues for implementing DBMS(Data Base Management Systems) such as concurrency control, recovery, and transactions managements.

- 오픈소스SW개발 (Open Source SW Development)

오픈소스 SW 개념을 소개하고 활성화의 중요성을 이해한다. 그리고, SW개발 전 과정에서 널리 사용되는 오픈소스 솔루션들을 실습하며 자신의 SW를 개발하여 오픈소스로 공개하는 프로젝트를 진행한다.

This course introduces the concepts of open source SW and discusses importance of its invigoration. Then it deals with how to use open source solutions that are widely used in SW development processes. It also offers students an opportunity to develop his own SW and publish it as open source.

- IT기술영어 1/2/3 (IT Technical English 1/2/3)

IT 기초이론과 최근의 연구동향 및 컴퓨터공학의 전문용어를 습득함으로써 논문 및 원서해석에 필요한 기초능력을 기른다.

By acquiring the IT basic theory and recent research trends and terminology of computer engineering, learn to improve basic skills needed article and books.

- 캡스톤디자인 1 (Capstone Design 1)

이 과목에서는 급변하는 컴퓨터, 멀티미디어, 인터넷 등 컴퓨터 전 분야에 관련된 새롭고 다양한 주제를 일정 소규모의 학생들이 그룹을 형성하여 교수들과 심도 있게 학습할 수 있는 기회를 제공한다. 교과 진행방법은 관심 주제에 대한 세미나, 산업 현장 인턴 실습, 실제 프로젝트 수행, 연구논문 작성 등이다. 이 과목을 효과적으로 수행하기 위해서 학생들은 학기 초에 관심분야에 대한 학업 계획서를 작성하여 관련 교수와 상담을 통해 학습 주제를 결정하도록 되어 있다.

This course offers students an opportunity to study new and various subjects related to computer, multimedia, and Internet technologies as a group with their supervisor. The course includes seminar, industry internship, project study, and research paper writing. In order to efficiently complete this course, students are advised to determine their own research topics by preparing their study plans and contacting their supervisor in the beginning of the semester.

- 캡스톤디자인 2 (Capstone Design 2)

이 과목에서는 급변하는 컴퓨터, 멀티미디어, 인터넷 등 컴퓨터 전 분야에 관련된 새롭고 다양한 주제를 일정 소규모의 학생들이 개인별로 교수들과 심도 있게 학습할 수 있는 기회를 제공한다. 교과 진행방법은 관심 주제에 대한 세미나, 산업 현장 인턴실습, 실제 프로젝트 수행, 연구논문 작성 등이다. 캡스톤디자인 1을 수강한 학생들이 다음 학기에 수강하는 과목으로 캡스톤디자인 1에

---

이어서 연구 활동을 계속해 나가며, 지정된 날에 연구 결과에 대한 최종 발표를 하고, 보고서를 작성하여 제출한다.

This course offers students an opportunity to study new and various subjects related to computer, multimedia, and Internet technologies with their supervisor. The course includes seminar, industry internship, project study, and research paper writing. The course is provided for the students who have already completed the Special Topics in Capstone Design 1. The students are advised to continue their researches and studies based on the mid-result of the Special Topics in Capstone Design 1, present their final research results, and submit the corresponding reports.

- 졸업논문 (Graduation Thesis)

컴퓨터공학과는 캡스톤디자인 1 및 캡스톤디자인 2를 이수하는 것으로 졸업을 위한 “졸업논문” 합격 여부를 결정한다.

Bachelor of Engineering in Computer Engineering decides acceptance of graduation thesis by completing Capstone Design 1 and Capstone Design 2.

- 신호와시스템 (Signals and Systems)

연속 및 이산 신호와 시스템의 수학적 표현기법, 분석 및 신호 합성에 관한 기본 개념과 변환기법을 다룬다. Fourier 변환, Z-변환, Laplace 변환 등을 기초로 한 신호와 시스템 분석 방법에 관한 기본이론 및 필터링, 변조 등의 응용 예를 강의한다.

Signals and Systems provides basic theory for mathematical modeling and analysis of electrical circuits, communications, control, image processing, and electromagnetics. Signals and systems are analyzed in the time and frequency domains. This course covers basic continuous and discrete time signals, system properties, linear time invariant systems, convolution, continuous and discrete time Fourier analysis.

- 문제해결 (Problem Solving)

다양한 유형의 문제 해결을 위한 기본 이론들을 알고리즘 개발을 통해 습득한다. 또한 개발한 알고리즘을 컴퓨터 프로그램으로 구현함으로써 프로그래밍 실력을 배양한다.

This course introduces fundamental theories for solving various types of problems through algorithm development. It also emphasizes how to implement the algorithms developed in computer programming languages.

- 형식언어 및 컴파일러 (Formal Languages and Compiler)

전산학의 근간을 이루는 형식 언어에 대한 기본적인 이해와 컴파일러와의 연관관계를 배운다. 구체적으로, 언어의 문법적 표현과 분류, 유한상태 기계, push-down 자동장치, 정규언어, 튜링기계, 파싱 기법의 요소들을 이해하고 응용할 수 있는 능력을 키운다.

In this course, elementary knowledge of formal languages on which computer science is based and the relation of the formal language and compiler concept are discussed. In detail, language grammars and expressions, finite state machines, push-down automata, regular language, turing machine, parsing methods are studied.

- 파일처리 (File Processing)

대용량 기억장치에 데이터를 조작하는 개념과 기법을 익히고, 대용량 기억장치의 활용능력을 배양하며, 자료구조와 파일처리 기법의 응용을 위한 기초지식을 습득할 수 있다.

This course deals with the concepts and techniques for manipulating data in mass storage devices such as hard disks. This course allows students to learn basic knowledge for application of data structures and file processing techniques. It also helps students expand the ability to utilize mass storage devices for managing a large amount of data.

- 멀티미디어시스템 (Multimedia System)

멀티미디어의 정의 및 기본 개념, 멀티미디어 시스템 요구사항과 기술동향, 코딩과 압축, 전송기술을 배운다. 그리고 멀티미디어 시스템 설계 시 필요한 기술 등 응용 방향을 제시하며 실습을 수행한다.

This course provides the basic concepts and the requirements of the multimedia system, video coding and compression, and the technology for transmissions. This course also provides the required technology to design of the

multimedia system and practical applications, followed by the performing the practices.

- 시스템분석 및 설계 (Systems Analysis and Design)

시스템 개발과정을 소개하며, 소프트웨어 시스템 분석 및 설계 시에 확장성과 재사용을 용이하게 하기 위한 구조적 방법과 객체 지향적 방법을 익힌다.

This course introduces the system development processes and focuses on the structural and object-oriented methodologies in software system analysis and design to support scalability and reusability.

- 프로그래밍언어구조론 (Structures of Programming Languages)

이 과목은 프로그래밍 언어의 기본적인 개념을 다루는데, 1) 다양한 언어 구조에 대한 설계 문제들을 토의하고, 2) 그런 구조들을 일반적인 언어에서는 어떻게 설계 선택을 했는지 조사하며, 3) 설계 선택 대안들을 비교하는 내용으로 구성된다.

This course treats the fundamental concepts of programming languages by 1) discussing the design issues of the variable language constructs, 2) examining the design choices for these constructs in some of the most common languages, and 3) critically comparing design alternatives.

- SW스타트업비즈니스 (SW Start-Up Business)

실리콘밸리의 수많은 성공한 창업가들이 컴퓨터공학과 출신으로 이미 컴퓨터공학은 창업가들로 하여금 가장 핵심적이고 중요한 기술로 부각되고 있다. 구글, 마이크로소프트, 샤오미 등에서 제시하는 차세대 기술에 대해 연구하고 이러한 기술을 기반으로한 모의 창업을 통해 졸업 후 사업가 역량을 배양한다.

Many successful entrepreneurs major computer engineering from the Silicon Valley. Computer engineering has emerged as the most critical and important skills of entrepreneurs. This course researching on a next-generation technology proposed by Google, Microsoft and Xiaomi. Students simulate a mock business based on research result and improve business skills needed at the company after graduation.

- SW스타트업프로젝트 (SW Start-Up Project)

산업사회가 고도화되면서 창업의 비중이 점차 높아지고 있는 추세이다. 오픈소스SW를 기반으로 창업아이템 선정, 사업기획, 제품 샘플 개발까지의 일련의 과정을 포함한 모의창업을 연계하여 실제 창업 프로세스를 경험한다.

As the industrial society becomes more sophisticated, the proportion of start-ups is gradually increasing. Based on open source software, students experience the actual start-up process by linking mock startups including a series of processes from startup item selection, business planning, and product sample development.

- 단기현장실습 (Short-Term Internship)

관련 기업에서 실무 경험을 통해 전공지식을 응용한다. (활동기간에 따라 학점 부여)

This course gives a chance to apply theoretical knowledge in a field.

- 장기현장실습 (Long-Term Internship)

관련 기업에서 장기 실무 경험을 통해 전공지식을 응용한다. (활동기간에 따라 학점 부여)

This course gives a chance to apply theoretical knowledge in a field through long-term practical experience.

- 연구연수활동 1/2 (Research Internship in Computer Engineering 1/2)

연구실에서 진행하는 연구에 대한 실무 경험을 통해 전공지식을 응용한다.

This course gives a chance to apply research knowledges in laboratories.

- 인공지능 (Artificial Intelligence)

인공지능 분야는 전통적인 Symbolic AI 와 신경망에 기반한 Connectionist AI 로 나뉜다. 최근에 기계학습의 딥러닝 분야에서

Connectionist AI 가 각광을 받고 있으므로 기계학습 및 딥러닝 과목을 따로 만들어 Connectionist AI을 깊이 배우고, 인공지능 과목에서는 Symbolic AI 분야를 깊이 학습하는 것을 목표로 한다. 딥러닝이 잘 해결할 수 없는 문제 영역이 어떤 것들인지 이해하고, Symbolic AI 분야 해결 방법들을 배운다.

Artificial intelligence is divided into traditional Symbolic AI and neural network-based Connectionist AI. Recently, Connectionist AI has been in the spotlight in deep learning of machine learning. Our department create separate courses for machine learning and deep learning to learn Connectionist AI deeply. In AI course, students learn deeply about Symbolic AI field. Students can understand what problem areas deep learning cannot solve and learn solutions for Symbolic AI problems.

- 딥러닝 (Deep Learning)

딥러닝의 배경지식인 기초 수학(선형대수, 확률, 정보이론)을 리뷰하고, 기본 개념과 다양한 딥러닝 기술의 이론 지식을 학습한다. Basic mathematics (linear algebra, probability, information theory), which is a background knowledge of deep learning, is reviewed. Students learn the concepts and theoretical knowledge of various deep learning technologies.

- 실전 기계학습 (Machine Learning in action)

딥러닝/머신러닝 기본 지식을 실제 문제에 응용할 수 있는 실습 과목이다. 학생들은 머신러닝/딥러닝 기본 모델링 기법을 배우고 프로젝트/competition에 참여하여 다양한 실전 경험을 쌓는다.

This course learns how to apply machine learning and deep learning technology to practical applications. The course allows the students to learn basic models and optimization techniques by participating Kaggle-based challenges and performing projects for practical applications.

- 빅데이터프로그래밍 (Big Data Programming)

대량의 정형 또는 비정형 데이터의 집합체인 빅데이터의 5V (규모, 다양성, 속도, 정확성, 가치) 요소에 대해 학습하고, 빅데이터로부터 가치를 추출하고 결과를 분석하기 위한 하둡의 맵리듀싱에 대해 학습한다.

Learn 5V elements (Volume, Variety, Velocity, Veracity, Value) of Big Data, a collection of large quantities of structured or unstructured data, and learn Hadoop and MapReducing for extracting values from Big Data and analyzing results.

- 클라우드컴퓨팅 (Cloud Computing)

클라우드 컴퓨팅의 개념을 이해하고 프로젝트를 진행한다. 가상화 기술(Virtual Machine)의 특징을 이해하고 IaaS, PaaS, SaaS 등의 개발 프로젝트를 진행한다. 아마존과 구글의 클라우드 서비스 활용 방법과 PaaS(Azure, Openshift, Cloudfoundry)를 이용한 클라우드 서비스의 개발 실습과 OpenStack을 통한 가상 자원의 프로비저닝 및 운영기술에 대한 실무를 배양한다.

Understand the concept of cloud computing and proceed with the project. Understand the features of virtual machine and develop projects such as IaaS, PaaS, SaaS. Learn how to use cloud service of Amazon and Google and cloud service development practice using PaaS (Azure, Openshift, Cloudfoundry) and cultivate practical skill of provisioning and operating virtual resources through OpenStack.

- 데이터센터프로그래밍 (Datacenter Programming)

데이터센터와 서버 프로그래밍에 대한 이론을 이해하고 실습을 수행한다. 대표적인 기술로서 클라우드 컴퓨팅(OpenStack 등), 컨테이너 기술(Docker 등), 빅데이터 분석(Spark, Hadoop 등)에 대해서 이해하고, 실습을 통해서 직접 프로그램을 개발하여 본다.

Understand the theory and practice of data center and server programming. As a representative technology, students understand about cloud computing (OpenStack etc.), container technology (Docker etc.), big data analysis (Spark, Hadoop, etc.) and develop program directly by practice.

- AI 네트워킹 (AI Networking)

본 과정에서는 인터넷프로토콜의 핵심 기술을 다룬다. 또한 SVM, CNN, RNN, 강화학습과 같은 머신러닝의 중요기술을 기반으로



네트워킹 응용을 어떻게 구현할 것인가에 대해 강의한다. 그리고 프로젝트를 통해 수강생들은 시가반 인터넷 응용과 IoT네트워킹에 대한 설계 능력과 구현능력을 배양한다.

In this course, student can learn on the key technology of internet Protocol such as TCP/IP. In addition, it introduces the machine learning algorithms such as SVM, CNN, RNN, etc. Also, the network related applications using machine learning will be learned. Through project, the student can obtain the designing and implementing capabilities of AI based internet application and IoT systems.

- 정보보호 (Information Security)

본 과정에서는 네트워크보안의 기본개념과 암호화 알고리즘, 인터넷보안 메커니즘과 무선망 보안등에 대하여 다룬다. 그리고 국내외 보안기술표준화동향 등에 대해서도 강의한다.

Students learn about the basic concepts of network security, Internet security mechanism, and security for wireless networks through this course. In order to provide the opportunity for understanding the state-of-art security technology, this course also introduces the current domestic and international standardization status.

- 모바일프로그래밍 (Mobile Programming)

오픈소스 자바 프로그래밍 개발 환경에서 모바일 프로그래밍의 기본 개념을 배우고 여러 응용 프로그래밍을 구현함으로써 실제 업무에 적용 가능한 실무능력을 향상시키고, 최신 적용가능 라이브러리 및 테크닉에 대해 학습한다.

This course designed for learning a fundamental concept of the mobile programming based on the open source JAVA development environment and improving practical skills, and learn the latest applicable programming libraries and techniques.

- 웹서비스프로그래밍 (Web Service Programming)

본 과목에서는 HTML/CSS를 기본으로 하여 javascript를 이용한 Node.js 등과 같은 각종 웹 프레임워크를 이용한 웹서비스 백엔드 및 프론트엔드를 개발하는 방법을 소개한다.

This course introduce how to implement web service front-end and back-end using HTML/CSS, javascript, and the diverse web frameworks for Node.js.

- 블록체인 (Blockchain)

본 강의에서는 블록체인기술의 기본원리, 플랫폼, 어플리케이션에 대한 다양한 토픽들을 다룬다. 블록체인의 어플리케이션으로서 암호화폐 메커니즘과 그 비즈니스 어플리케이션 등에 대해 강의한다.

This lecture covers various topics on basic principles, platforms and application of blockchain technology. The lecture also deals with the mechanism of cryptocurrency and its business applications.

- 리눅스시스템프로그래밍 (LINUX System Programming)

가장 널리 사용되고 있는 운영체제 중의 하나인 LINUX 환경에서의 프로그램 개발 환경을 습득하여, UNIX 전문가로서의 계기를 제공한다. LINUX 사용법 및 개발환경에서 시작하여, file I/O, file & directory, process & thread, signal, IPC, synchronization, socket 등의 고급 프로그래밍 기법을 LINUX system call을 직접 사용함으로써 실습한다.

This course provides an opportunity for a LINUX expert. First, various development tools in UNIX are presented such as vi, make, gcc, gdb. Next, this course introduces various LINUX system calls and gives a lot of programming practices on file I/O, file & directory, process & thread, signal, IPC, synchronization, and sockets.

- IoT디지털시스템 (IoT Digital System)

소프트웨어와 하드웨어를 포함하는 SoC(System on a Chip)의 기본 설계 개념과 Altera DE2 board를 사용 구현 방법을 다룬다. 아울러 FPGA구현을 위한 Verilog-HDL 코딩방법과 실제적인 응용에 접근한 프로젝트들을 익힌다.

This course deals with the basic design concept of SoC(System on a Chip) including Hardware/Software and the

---

implementation method using Altera DE2 training board. And, also this course provides the Verilog-HDL coding method for FPGA implementation and some projects about practical application.

- IoT 소프트웨어 (IoT Software)

컴퓨터 공학 기초 과목의 이해를 바탕으로 Embedded System에서의 응용 프로그램 개발을 경험한다. 즉, Intel PXA255 기반의 Embedded System Toolkit을 사용하여, cross-compilation 개발 환경을 구축하고 Embedded Linux kernel porting, 각종 device control 등을 통해 Embedded System 개발 능력을 갖춘 후, 팀을 이루어 창의적인 응용 프로그램을 개발한다.

With a basic knowledge on computer engineering, this course gives an experience on embedded system programming. In this course, we develop the cross-development environment in Linux and port O/S kernel & various device drivers to an embedded system toolkit with an Intel PXA255 processor.

- 로봇 소프트웨어 (Robot Software)

Part 1. 내장형 시스템의 일종인 휴대폰 단말기에서 WPI 기반 응용 프로그램을 작성하는 기법을 익힌다.

Part 2. 로봇축구 시스템을 이해하고 전략을 작성하여봄으로써 임베디드 시스템의 응용 기법을 익힌다. 로봇축구는 시물 로봇과 마이크로봇을 다룬다.

Part 1. This course introduce how to develop application programs on embedded systems such as mobile phones by using WPI platform.

Part 2. This course introduce what is robot soccer system as an application of the embedded system. Programming the strategies for SIMULOSOT and MICROSOT is the main part of the project.

- IoT 네트워크 (Internet of Things)

본 과목은 사물인터넷과 디지털화라는 두가지 핵심 개념을 통해 디지털 트랜스포메이션이 어떻게 이루어지는지 강의한다. 이를 위해 IoT 네트워킹 기술과 4차 산업사회를 주도하는 최신 ICT 기술에 대하여 학습한다. IoT가 무엇인지, IoT가 무엇을 하는지, 그리고 IoT를 통한 디지털화가 4차 산업사회에 어떠한 영향을 주는지에 대한 개념을 기술적 측면에서 논한다.

This course teaches how digital transformation is realized through two key concepts: the Internet of Things (IoT) and digitization. To this end, students learn about IoT networking technologies and the latest ICT technologies that lead the fourth industrial society. The concept of what the IoT is, what the IoT does, and how IoT digitalization affects the fourth industrial society are discussed from technical points of view.

- 영상처리 (Image Processing)

2차원 신호인 디지털영상신호의 표현, 영상신호처리의 기본 단계, 영상신호처리 시스템의 요소, 디지털영상의 기초, 푸리에 변환, FFT, DCT를 포함한 영상변환, 영상신호의 향상, 및 영상신호의 복구에 대하여 강의한다.

This course teaches representation of 2D digital image signal, basic processing steps of image signal, elements of image signal processing system, image transform including Fourier transform, FFT and DCT, enhancement and restoration of image signal.

- 컴퓨터그래픽스 (Interactive Computer Graphics)

2D와 3D 객체의 생성과 디스플레이를 위한 기본적인 기술들을 소개한다. 주요 강의 내용은 그래픽스를 위한 자료구조, 그래픽 프로그래밍 언어, 기하학적 변환, shading, 가시화 등을 포함한다.

This course introduce techniques for the interactive generation and display of two and three dimensional objects. The topics to be covered will include data structure for graphics, geometric transformation, shading, visualization, and languages for graphics.

- 멀티미디어처리 (Multimedia Processing)

영상과 그래픽스, 그리고 비디오의 표현체계, 컬러체계, 비디오 코딩에 필요한 각종 알고리즘의 기본적인 개념, 비디오

무손실/손실 압축 방법, JPEG, MPEG 등의 국제 표준 등의 멀티미디어 통신에 필요한 기본 알고리즘에 관한 연구를 수행한다. This course provides the data representation, color models of the image, graphics and video, and also provides the basic and detailed concepts of the algorithms, lossless and lossy video compression, the international standards such as JPEG and MPEG for multimedia communications.

- 인간-컴퓨터상호작용 (Human-Computer Interaction)

본 과목은 인간-컴퓨터 상호작용 전반에 걸친 기본 원칙들과 그 원칙들의 응용사례들을 소개하는 과목으로 상호작용 가능한 컴퓨터 시스템을 사용자 중심의 방법론 안에서 디자인 하는 방법, 그리고 이러한 시스템을 사용성 측면에서 평가하는 방법을 배우게 된다. 특히, 컴퓨터 공학은 물론 인지과학, 사회과학, 상호작용 디자인 등 여러 학문분야가 다양하게 유기적으로 연결되어 있는 학문으로써, 본 과목에서 다룬 기본 원칙과 방법론들은 상호작용 가능한 모든 소프트웨어 및 하드웨어 시스템에 적용 가능함으로 통신, 협동, 교육, 의료 등 인간들의 삶의 질을 향상시키는데 아주 중요한 역할을 한다고 볼 수 있다.

This course aims at introducing basic principles and application examples of human-computer interaction(HCI). The students will learn the way how to design interactive computer systems through a user-centered approach and how to evaluate such systems in terms of usability. The principles, methods, techniques, and tools to be thought in this course are based on diverse areas including computer science, cognitive science, social science, and interaction design. The HCI principles and methods are used in designing and evaluation of most software and hardware systems that interact with human such as education, entertainment, and medical application.

- UI/UX프로그래밍 (UI/UX Programming)

본 교과목은 UI/UX 프로그래밍에 필요한 모든 과정을 순차적으로 학습하고, 실제 인터페이스 구현에 적용해 보는 수업이다. 우선, UI/UX의 정의 및 최적의 사용자 경험을 제공하기 위한 필요한 요소들에 대한 이론적인 배경을 배우고, 이를 달성하기 위해 따라야 하는 디자인 프로세스에 대한 실습을 수행한다. 또한, 인터페이스 프로그래밍 관련 기초 프로그래밍 기술을 학습하여, 수업 프로젝트를 통해 학생이 스스로 모든 과정을 직접 겪어보는 학습과정을 거친다.

This course aims to learn all the steps required for UI/UX programming sequentially and applies it to real interface implementation. First, students will learn the theoretical background about the definition of UI/UX and the necessary factors to provide the optimal user experience, and conduct a practical training on the design process to achieve this. In addition, students will learn basic programming skills related to interface programming, and through the class project, the student goes through a process of learning all of the processes themselves.

- 컴퓨터비전 (Computer Vision)

본 과목은 사람이 시각 정보를 이용하여 지능적인 판단과 행동을 하는 것과 마찬가지로, 컴퓨터(기계)도 사람과 같이 시각 정보를 획득, 처리 하고 이를 이용하여 객체 인식 및 행동 인식과 같은 지능적인 일을 수행 할 수 있도록 하는 최신 연구 분야를 공부한다. 기초적인 영상 분석 및 기계 학습 방법들을 활용하여 Feature Extraction, Visual Tracking, Shape Matching, Pattern Detection, Object Recognition, Gesture Recognition, Scene Understanding 및 3D Reconstruction 와 같은 세부 연구 분야 들을 공부한다.

Computer vision is an interdisciplinary research field that deals with how computers can be made for gaining high-level understanding from digital images or videos. In this class we study following topics: Feature Extraction, Visual Tracking, Shape Matching, Pattern Detection, Object Recognition, Gesture Recognition, Scene Understanding, 3D Reconstruction.

- 최신기술프로젝트 1/2 (New Technology Project 1/2)

본 과목에서는 산업계 최신 트렌드를 반영한 주제를 기반으로 심화 프로젝트를 수행한다. 최신 트렌드로 빅데이터 분석, 모바일 및 보안, IoT, 스마트컨텐츠 등의 분야를 다룬다.

This course aims to carry out further projects based on topics that reflect the latest trends in the industry. The latest trends cover big data analysis, mobile and security, IoT, and smart content.

---

- 최신기술콜로키움 1/2 (New Technology Colloquium 1/2)

본 과목에서는 유명 회사의 CTO를 초빙하여 관련 분야의 다양한 주제에 대한 특강을 진행한다. CTO Talk Concert를 통해 산업계의 특징과 노하우 등을 간접적으로 학습한다.

In this course, CTOs who work in famous companies are invited and give a lecture on various topics in related fields. Through the CTO Talk Concert, students learn indirectly the characteristics and know-how of the industry.

- 독립심화학습 1/2(컴퓨터공학과) (Individual In-depth Study 1/2)

논문 작성 및 제출, 특허 출원, 소프트웨어 도서 출간, 첨단 기술 백서 출간 등을 설계 및 진행

Students design his/her own course to write, submit and/or publish a technical paper/patent/books on software/technical white-paper.

– 특허 출원 (Patent Writing): 학생은 지도교수의 지도하에, 본인의 아이디어를 특허화 하고, 최종 출원하는 절차를 수행해 본다. 특허 작성을 희망하는 학생은 계획서와 실적서를 제출하고, 실질적인 작성 여부 등에 기반하여 지도교수로 부터 P/F 중 적합한 학점을 부여 받는다.

Under the supervision of an advisor, the student will patent his ideas and conduct the final application procedure. A student who wishes to write a patent must submit a proposal and an achievement book, and receive appropriate credits from the advisor based on whether he / she is actually writing or not.

– 소프트웨어 도서출간 (Software Books Publishing): 학생이 컴퓨터공학과 학생 혹은 대외 외부인에게 도움을 줄 수 있는 소프트웨어 도서를 출간하도록 한다. 도서는 전자도서 형태를 지행하며 오픈소스 형태로 배포되어야 한다. 학생은 작성 계획과 작성한 도서를 지도교수에게 제출하고, 실질적인 작성 여부 등에 기반하여 지도교수로 부터 P/F 중 적합한 학점을 부여 받는다.

Have students publish a software book that can be used by computer engineering students or outsiders. Books must be distributed in an open-source format and in electronic form. The student will submit the preparation plan and the book he / she has prepared to the supervisor and receive appropriate credits from the supervisor based on whether the essay is actually written or not.

– 첨단 기술 백서 출간 (Published High Technology White-paper): 학생이 컴퓨터공학과 학생 혹은 대외 외부인에게 도움을 줄 수 있는 첨단 기술에 대한 도서를 출간하도록 한다. 도서는 전자도서 형태를 지행하며 오픈소스 형태로 배포되어야 한다. 학생은 작성 계획과 작성한 도서를 지도교수에게 제출하고, 실질적인 작성 여부 등에 기반하여 지도교수로 부터 P/F 중 적합한 학점을 부여 받는다.

Have students publish books on cutting edge technologies that can help computer engineering students or outsiders. Books must be distributed in an open-source format and in electronic form. The student will submit the preparation plan and the book he / she has prepared to the supervisor and receive appropriate credits from the supervisor based on whether the essay is actually written or not.