

소프트웨어융합대학 컴퓨터공학과 교육과정 요약표(2024)

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 프로그래밍어 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

- 컴퓨터공학 전문지식 습득과 정보화 사회에 대한 이해를 바탕으로 글로벌 사회에서 각 분야 리더로서 활동할 수 있는 능력 배양
- 기초과학의 충실한 학습을 바탕으로 지식기반 사회에서 요구되는 창의적 능력 배양
- 기술적 문제를 공식화하고, 첨단 공학 도구를 사용하여 실험을 하고 수행함으로써 당면 문제를 체계적으로 해결할 수 있는 능력 배양
- 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	5	16	13	44	74
	학점수	15	45	33	120	213

※ 단기현장실습/장기현장실습 과목은 제외한 현황임

4. 컴퓨터공학과 졸업 요건

1) 교육과정 기본구조표

학과명	졸업 이수 학점	단일전공과정						다전공과정					부전공과정		
		전공학점					타 전공 인정 학점	전공학점				타 전공 인정 학점	부전공과정		
		전공 기초	전공 필수	전공선택		계		전공 기초	전공 필수	전공 선택	계		전공 필수	전공 선택	계
				산학필 수	전공선 택										
컴퓨터공학과	130	15	45	12	15	87	6	12	27	15	54	-	15	6	21

※ 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

컴퓨터공학과의 ‘졸업프로젝트’를 이수하는 것으로 경희대학교 졸업을 위한 “졸업논문” 합격으로 인정한다. 단, “졸업논문(컴퓨터 공학)”을 필히 수강 신청하여야 한다.

소프트웨어융합대학 컴퓨터공학과 교육과정 시행세칙(2024)

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 [별표4]와 같으며, 선·후수과목의 체계를 준수하여 이수하여야 한다. 선·후수과목의 체계는 전산시스템에 반영되어있으며 수강신청시 자동으로 적용된다.

제 2 장 교양과정

제3조(교양이수학점) 교양과목은 교양교육과정 기본구조표에서 정한 소정의 교양학점을 취득하여야 한다.

제 3 장 전공과정

- 제4조(전공 및 트랙과목 이수)** ① 컴퓨터공학과와의 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 한다.
- 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.

[표1] 단일전공 전공과목 편성표

구분	교과목명	과목수
전공기초 (15)	선형대수, 미분적분학, 미분방정식, 확률및랜덤변수(EE), 이산구조	5

구분	교과목명		과목수
전공필수 (45)	객체지향프로그래밍, 디자인적사고(SWCON), 웹/파이선프로그래밍(SWCON), 논리회로(EE), 컴퓨터구조, 자료구조, 오픈소스SW개발방법및도구(SWCON), 운영체제, 컴퓨터네트워크, 알고리즘, 데이터베이스, 소프트웨어공학, 기계학습, 캡스톤디자인, 졸업프로젝트, 졸업논문(컴퓨터공학)		16
전공선택 (27)	산학필수(12)	SW스타트업비즈니스, SW스타트업프로젝트, 빅데이터프로그래밍, 빅데이터프로젝트, 클라우드컴퓨팅, 클라우드프로젝트, 모바일/웹서비스프로그래밍, 모바일/웹서비스프로젝트, 블록체인, 최신기술콜로키움1, 최신기술콜로키움2(SWCON), 단기현장실습/장기현장실습, 연구연수활동1·2	15
	공동 선택	신호와시스템(EE), 회로와신호(SWCON), 컴파일러, 프로그래밍언어론, 리눅스시스템프로그래밍, 독립심화학습1·2	7
	인공지능/ 빅데이터 분야	빅데이터프로그래밍, 딥러닝, 실전기계학습, 빅데이터프로젝트, AI네트워킹, 인공지능프로그래밍(AI), 고급딥러닝(AI), 빅데이터마케팅(AI), 자연어처리(SWCON), 강화학습(SWCON)	10
	지능형시스템 및 보안 분야	클라우드컴퓨팅, 클라우드프로젝트, 모바일/웹서비스프로그래밍, 모바일/웹서비스프로젝트, 정보보호, 소프트웨어보안, 웹보안, 마이크로서비스프로그래밍(SWCON), 플스택서비스네트워킹(SWCON), 플스택서비스프로그래밍(SWCON)	10
	지능형로봇 및 AIoT 분야	영상처리, 컴퓨터비전, AIoT소프트웨어, AIoT네트워크, AIoT디지털시스템, 로봇프로그래밍(SWCON), 3D데이터처리(SWCON)	7
메타버스 분야	UI/UX프로그래밍, 메타버스시스템, 컴퓨터그래픽스, 메타버스테이터처리, 인간-컴퓨터상호작용, 가상/증강현실이론및실습(SWCON), 게임인터랙티브테크놀로지(SWCON), 게임프로그래밍입문(SWCON), 게임엔진기초(SWCON), 게임그래픽프로그래밍(SWCON), 인공지능과게임프로그래밍(SWCON), 실감미디어컴퓨팅기초	12	

※ 모든 학생은 제 5조의 산학필수 학점을 취득하여야 함

[표2] 다전공 전공과목 편성표

구분	교과목명
전공기초 (12)	선형대수, 미분적분학, 확률및랜덤변수, 이산구조
전공필수 (27)	객체지향프로그래밍, 컴퓨터구조, 자료구조, 운영체제, 컴퓨터네트워크, 알고리즘, 데이터베이스, 소프트웨어공학, 졸업프로젝트, 졸업논문(컴퓨터공학)
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 '객체지향프로그래밍및실습'으로 대체할 수 있으며, '자료구조'는 '자료구조및알고리즘'으로 대체할 수 있음

※ 2023년 1학기부터 컴퓨터공학을 다전공으로 이수하는 학생의 경우, 전공필수 과목인 '컴퓨터구조'의 선수과목인 '논리회로'를 해제 함

[표3] 부전공 전공과목 편성표

구분	교과목명	
부전공과정	전공필수 (15)	객체지향프로그래밍, 웹/파이선프로그래밍, 자료구조, 운영체제, 알고리즘
	전공선택 (6)	부전공과정 전공필수에 포함되지 않은 컴퓨터공학과 단일전공의 전공필수/전공선택 교과목

※ 위 부전공 전공필수/전공선택 과목이 단일전공 및 다전공의 전공기초/전공필수 교과목과 중복되는 경우, 최대 2과목에 한하여 컴퓨터공학과 단일전공의 전공필수/전공선택 교과목으로 대체 이수 가능함

※ 2023년 1학기부터 컴퓨터공학을 부전공으로 이수하는 학생의 경우, '객체지향프로그래밍'은 '객체지향프로그래밍및실습'으로 대체할 수 있으며, '자료구조'는 '자료구조및알고리즘'으로 대체할 수 있음

※ 2023년 1학기부터 컴퓨터공학을 부전공으로 이수하는 학생의 경우, 전공필수 과목인 '컴퓨터구조'의 선수과목인 '논리회로'를 해제 함

제5조(산학필수 이수) 단기현장실습/장기현장실습을 포함하여 연구연수활동1·2, 최신기술콜로키움1, 최신기술콜로키움2(SWCON), SW스타트업비즈니스, SW스타트업프로젝트, 빅데이터프로그래밍, 빅데이터프로젝트, 클라우드컴퓨팅, 클라우드 프로젝트, 모바일/웹서비스프로그래밍, 모바일/웹서비스프로젝트, 블록체인 중 12학점 이상을 이수하여야 한다.

제6조(대체교과목의 지정) 컴퓨터공학과와 전공과목의 대체과목은 [별표3]과 같다.

제7조(지식창업 트랙 운영) ① 학생들이 스스로 진로를 설계하고 창업과 진로를 열어가 수 있도록 사회적 문제의 인식과 창의적 문제해결 역량 강화를 위한 지식창업 트랙을 운영한다.

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

[표4] 지식/창업트랙 교육과정 편성표

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) - 졸업프로젝트(3)	6	* 전공 -전공선택 -전공필수 (최대 6학점 중복인정)	컴퓨터공학과	
이수학점 계			18			

제8조(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

제9조(졸업이수학점) ① 컴퓨터공학전공은 최소 졸업이수학점은 130학점이다.

- ② 교양학점은 후마니타스 교양교육과정을 만족하여야 한다.
- ③ 졸업논문을 포함하여 해당 졸업이수요건을 충족시켜야 한다.

제10조(컴퓨터공학 졸업이수요건) ① 단일전공과정 : 컴퓨터공학과 학생으로서 단일전공자는 전공기초 15학점, 전공필수 45학점, 산학필수 12학점, 전공선택 15학점을 포함하여 전공학점 87학점 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 : 컴퓨터공학전공을 부전공과정으로 이수하고자 하는 자는 전공필수 15학점을 포함하여 전공학점 21학점을 이수하여야 한다.
- ④ 전공과목의 영어강좌 3과목 이상 이수를 졸업요건으로 충족해야 한다.

제11조(졸업능력인증제도) 졸업능력인증제도는 폐지하며, 경과조치를 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 '졸업프로젝트'를 이수하는 것으로 경희대학교 졸업을 위한 "졸업논문" 합격으로 인정한다. 단, "졸업논문(컴퓨터공학)"을 필히 수강 신청하여야 한다.

제15조(SW교육 졸업요건) 2018학년도 이후 입학생(편입생, 순수외국인 제외)은 후마니타스칼리나 단과대학에서 개설되는 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(보칙) 본 내규에 정하지 않는 사항은 컴퓨터공학과 학과회의 의결에 따른다.

제17조(마이크로디그리 이수) 마이크로디그리 과정을 이수하고자 하는 자는 [별표6] 마이크로디그리 이수체계에도에서 지정한 소정의 학점을 충족하여야 한다.

제18조(외국인 학생의 한국어 능력 취득) 한국어트랙 외국인 학생은 졸업 전까지 한국어능력시험(TOPIK) 4급 이상을 취득하여야 한다.

부 칙

제1조(시행일) 본 시행세칙은 2024년 3월 1일부터 시행한다.

제2조(경과조치) ① 2010학번 이후 학생 중 컴퓨터공학을 단일전공 또는 다전공하는 학생들은 응용과학대학 교육과정인 '물리학1' 및 '물리학실험'을 '물리학및실험1'로 대체 인정한다.

- ② 2009년 이후 2017년 이전 입학생 중 컴퓨터공학과 단일전공과정과 다전공과정의 졸업 이수 요건을 전공기초 21학점으로 변경한다.
- ③ 제외된 전공기초 이수교과목(미분적분학2, 물리학및실험2, 일반화학, 일반생물) 대신 전공 교과목의 추가이수를 권장한다.
- ④ 2017학번 이전 학생 중 전공필수인 '시스템분석및설계'를 수강하지 않은 학생은 '소프트웨어공학' 이수를 전공필수로 인정할 수 있다.
- ⑤ 2017년도 이전 입학생의 경우, 각 학번별 선수과목 대신 [별표2]의 선수과목 지정표에 따른 선수과목을 따를 수 있다.
- ⑥ 컴퓨터공학과를 다전공으로 이수하고자 하는 학생은 응용과학대학 교육과정인 '미분적분학 1'을 '미분적분학'으로 대체 인정한다.

- ⑦ 컴퓨터공학과에 전과한 학생은 응용과학대학 교육과정인 '미분적분학1'을 '미분적분학'으로 대체 인정한다.
- ⑧ 컴퓨터공학과 전공기초 교과목은 타 단과 대학의 다음의 [별표4]의 유사 교과목으로 대체 가능하다.
- ⑨ 2018학년도 이후 컴퓨터공학과를 부전공으로 이수하고자 하는 학생 중 부전공 전공필수 및 전공선택 교과목이 단일전공 및 다전공의 전공기초 및 전공필수 교과목과 중복되는 경우 최대 2과목에 한하여 컴퓨터공학과 단일전공의 전공필수 및 전공선택 교과목으로 대체 이수를 인정한다.
- ⑩ 2023학년도 이전 입학생의 경우, 전공필수 교과목인 IT기술영어1/2/3의 폐지에 따라 미이수자에 한해 임의의 전공선택 교과목을 IT기술영어1/2/3의 전공필수 대체교과목으로 인정한다. 단, IT기술영어1/2/3을 모두 이수하였을 경우 이에 해당되지 않으며, 부분 이수한 경우 해당 학점은 전공선택 학점으로 인정한다.
- ⑪ (졸업능력인증제 폐지에 따른 경과조치) 졸업능력인증제 폐지는 2023학년도부터 모든 재적생에서 적용하되, 2023.03.01. 이전 수료자는 희망자에 한하여 이수면제 처리한다.

[표5] 입학년도에 따른 컴퓨터공학과 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

***별표 내용 표기 유무 확인부탁드립니다.**

[별표1]

컴퓨터공학과 교육과정 편성표

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		부전공	P/F 평가	비고	
					이론	실기	실습	설계		1학기	2학기				
1	전공 기초	미분방정식	AMTH1001	3	3				1		○				
2		선형대수	AMTH1004	3	3				1	○					
3		미분적분학	AMTH1009	3	3				1	○					
4		이산구조	CSE201	3	3				2	○	○				
5		확률및랜덤변수	EE211	3	3				2	○	○				
6	전공 필수	객체지향프로그래밍	CSE103	3	2		2		1	○	○	○			
7		컴퓨터구조	CSE203	3	3				2	○	○	○			
8		자료구조	CSE204	3	2		2		2	○	○	○			
9		운영체제	CSE301	3	3				3	○	○	○			
10		컴퓨터네트워크	CSE302	3	3				3	○	○	○			
11		알고리즘	CSE304	3	2		2		3	○	○	○			
12		데이터베이스	CSE305	3	3				3	○	○	○			
13		소프트웨어공학	CSE327	3	3				3	○	○	○			
14		졸업논문(컴퓨터공학)	CSE403	0					4	○	○	○	○		
15		졸업프로젝트	CSE405	3				3	4	○	○	○			
16		캡스톤디자인	CSE406	3				3	4	○	○	○			
17		논리회로	EE209	3	3				2	○	○	○			
18		디자인적사고	SWCON103	3	2		2		1	○	○	○			
19		웹/파이선프로그래밍	SWCON104	3	2		2		1	○	○	○			
20		오픈소스SW개발방법및도구	SWCON201	3	3				2	○	○	○			
21		기계학습	SWCON253	3	2			1	2	○	○	○			
22		전공 선택	인공지능프로그래밍	AI1002	3	2		2		1		○	○		
23			고급딥러닝	AI3001	3	3				3	○	○	○		
24			빅데이터마이닝	AI3004	3	3				3-4	○		○		
25	실감미디어컴퓨팅기초		CSE104	3	3				1	○	○				
26	UI/UX프로그래밍		CSE224	3	3				2	○		○			
27	컴파일러		CSE322	3	3				3	○		○			
28	메타버시스템		CSE324	3	3				3	○		○			
29	프로그래밍언어론		CSE328	3	3				3		○	○			
30	SW스타트업비즈니스		CSE330	3	3				3	○		○			
31	딥러닝		CSE331	3	3				3	○	○	○			
32	리눅스시스템프로그래밍		CSE332	3	2		2		3		○	○			
33	SW스타트업프로젝트		CSE334	3				3	3		○	○			
34	클라우드컴퓨팅		CSE335	3	3				3	○		○			
35	실전기계학습		CSE340	3				3	3	○	○	○			
36	정보보호		CSE423	3	3				4	○		○			
37	영상처리		CSE426	3	3				4	○		○			
38	컴퓨터그래픽스		CSE428	3	3				4	○		○			
39	메타버스테이터처리	CSE430	3	3				4		○	○				

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		부전공	P/F 평가	비고
					이론	실기	실습	설계		1학기	2학기			
40	전공 선택	인간-컴퓨터상호작용	CSE431	3	3				4		○	○		
41		빅데이터프로그래밍	CSE434	3	3				3		○	○		
42		빅데이터프로젝트	CSE436	3				3	4	○		○		
43		클라우드프로젝트	CSE437	3				3	4		○	○		
44		최신기술콜로키움1	CSE438	2	2				4	○		○	○	
45		AIoT디지털시스템	CSE439	3	2		2		4		○	○		
46		AIoT소프트웨어	CSE440	3	2		2		4	○		○		
47		컴퓨터비전	CSE441	3	3				4		○	○		
48		블록체인	CSE442	3	3				4	○	○	○		
49		시네트워킹	CSE443	3	3				4	○		○		
50		AIoT네트워크	CSE444	3	3				4	○		○		
51		모바일/웹서비스프로그래밍	CSE450	3	3				4	○		○		
52		모바일/웹서비스프로젝트	CSE451	3	3				4		○	○		
53		소프트웨어보안	CSE452	3	3				4		○	○		
54		웹보안	CSE453	3	3				4	○		○		
55		독립심화학습1(컴퓨터공학과)	CSE495	3			3		3	○		○	○	
56		연구연수활동1	CSE496	1			2		2-4	○		○	○	
57		연구연수활동2	CSE497	1			2		2-4		○	○	○	
58		독립심화학습2(컴퓨터공학과)	CSE499	3			3		3		○	○	○	
59		신호와시스템	EE210	3	3				2	○	○	○		
60		게임프로그래밍입문	SWCON211	3	2		2		2	△	○	○		포
61		개인엔진기초	SWCON212	3	2		2		2		○	○		
62		마이크로서비스프로그래밍	SWCON221	3	2		2		2	○	○	○		
63		회로와신호	SWCON254	3	3				2	○		○		
64		최신기술콜로키움2	SWCON302	2	2				2-4		○	○	○	
65		게임그래픽프로그래밍	SWCON311	3	2		2		3	○	△	○		포
66		게임인터랙티브테크놀로지	SWCON312	3	2			1	4	○		○		
67		가상/증강현실이론및실습	SWCON313	3	2		2		3		○	○		
68		로봇프로그래밍	SWCON331	3	3				3-4		○	○		
69		3D데이터처리	SWCON366	3	3				3	○		○		
70		플스택서비스프로그래밍	SWCON370	3	3				3-4	○	△	○		
71		인공지능과게임프로그래밍	SWCON491	3	2		2		3-4	○		○		
72		플스택서비스네트워킹	SWCON492	3	3				2-4	△	○	○		
73		자연언어처리	SWCON493	3	3				4	○		○		
74		강화학습	SWCON495	3					4	○		○		
75		단기현장실습			*1) 3/6			6/12	2-4	○	○	○		
76	장기현장실습			*1) 9/12			18/24	2-4	○	○	○			

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

※ 비교의 '포'는 포트폴리오 교과목으로서, 반드시 교과목내 결과물을 개발하고, 학부 과정 동안 개인 실적으로서 관리해야 함

[별표2]

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	단일전공 이수자만 해당
3		CSE304	알고리즘	3	CSE204	자료구조	3	
4		CSE406	캡스톤디자인	3	SWCON103	디자인적사고	3	
5		CSE405	졸업프로젝트	3	CSE406	캡스톤디자인	3	단일전공 이수자만 해당
6		EE210	신호와시스템	3	AMTH1009	미분적분학	3	
7		SWCON201	오픈소스SW개발방법및도구	3	SWCON104	웹/파이선프로그래밍	3	
8		CSE424	AIoT디지털시스템	3	CSE203	컴퓨터구조	3	

※ 우측 선수과목 수강 시에 좌측 후수과목 수강을 허용함

[별표3]

대체과목 일람표

순번	학과명	구 교과과정		현행 교과과정	
		교과목명	학점	교과목명	학점
1	컴퓨터공학과	미분적분학1	3	미분적분학	3
2		기초공학설계	3	디자인적사고	3
3		프로그래밍기초	3	웹/파이선프로그래밍	3
4		고급객체지향프로그래밍	3	객체지향프로그래밍	3
5		컴퓨터공학개론	3	오픈소스SW개발방법및도구	3
6		창의적설계1	3	캡스톤디자인	3
7		창의적설계2	3	졸업프로젝트	3
8		창의적설계	3	캡스톤디자인	3
9		창의적종합설계(컴퓨터공학)	3	캡스톤디자인	3
10		졸업연구	3	졸업프로젝트	3
11		설계프로젝트A/B/C/D	12 (각 3학점)	최신기술콜로키움1·2, SW스타트업비즈니스, SW스타트업프로젝트, 빅데이터프로그래밍, 빅데이터프로젝트, 클라우드컴퓨팅, 클라우드프로젝트, 모바일/웹서비스프로그래밍, 모바일/웹서비스프로젝트, 연구연수활동1·2 (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
15		오픈소스SW개발	3	오픈소스SW개발방법및도구	3
16		모바일프로그래밍	3	모바일/웹서비스프로그래밍	3
17		웹서비스프로그래밍	3	모바일/웹서비스프로그래밍	3
18		로봇소프트웨어	3	로봇프로그래밍	3
19		멀티미디어시스템	3	메타버스시스템	3
20		메타버스처리	3	메타버스데이터처리	3
21		체감형기술이론및실습	3	게임인터랙티브테크놀로지	3
22		형식언어및컴파일러	3	컴파일러	3
23		프로그래밍언어구조론	3	프로그래밍언어론	3
24		모바일프로그래밍	3	모바일/웹서비스프로그래밍	3
25		IoT디지털시스템	3	AIoT디지털시스템	3
26		IoT소프트웨어	3	AIoT소프트웨어	3
27		IoT네트워크	3	AIoT네트워크	3

[별표4]

전공기초 교과목 강좌 대체 이수 인정 교과목

순번	컴퓨터공학과 전공기초 교과목명	대체 인정 교과목
1	AMTH1002 미분적분학1	MATH1101 미적분학및연습1(이과대학 수학과)
2	AMTH1003 고급미분적분학	MATH1102 미적분학및연습2(이과대학 수학과)
3	AMTH1009 미분적분학	MATH1101 미적분학및연습1(이과대학 수학과) MATH1102 미적분학및연습2(이과대학 수학과) 중 한 과목
4	AMTH1001 미분방정식	MATH2411 미분방정식1(이과대학 수학과) MATH2412 미분방정식2(이과대학 수학과) 중 한 과목
5	AMTH1004 선형대수	MATH2111 선형대수학1(이과대학 수학과) MATH2112 선형대수학2(이과대학 수학과) 중 한 과목

컴퓨터공학과 교과목 해설

• 미분방정식 (Differential Equations)

Homogeneous와 non-homogeneous Linear Differential Equations의 해, 미분방정식의 응용, Laplace transformation, Inverse transform, Series Solutions of Differential Equations 등을 공부한다.

In this course, we will study Differential Equations(in means the ordinary differential equations) and their applications. Moreover, we will consider the elementary course of Fourier Series.

• 선형대수 (Linear Algebra)

역행렬, 선형계, 행렬식, 가우스 소거법, 내적, 벡터공간, 일차독립, 기저, Kernel and range, 선형변환, Eigenvalues and Eigenvectors, 대각화, 최소자승법 등을 공부한다.

The course treats linear systems, Gaussian elimination, inverse matrix, determinant, inner product, vector space, linear independence, basis, kernel and range, linear transformations, eigenvalues and eigenvectors, diagonalization, and least-square method.

• 미분적분학 (Calculus)

일변수 함수의 미분, 적분 이론과 그 응용에 대하여 공부한다.

In this course, we study the derivatives and integral theories of functions(functions of one variable), the partial derivatives of functions of several variables, and their applications.

• 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.

• 확률및랜덤변수 (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.

• 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.

• 컴퓨터구조 (Computer Architecture)

컴퓨터 구조 설계의 기초 이론으로써 기본적인 컴퓨터 시스템의 구성과 설계에 대한 개념과 기법을 소개한다. 데이터의 표시방법, 레지스터의 전송과 마이크로 동작, 컴퓨터 소프트웨어를 포함하여 연산장치, 제어장치, 입출력장치의 구조와 설계기법을 학습함으로

써 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.

• 알고리즘 (Algorithms)

알고리즘의 기본적인 이해를 하고 대표적인 알고리즘의 형태를 배운다. 알고리즘 유형을 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 알고리즘을 설계할 수 있는 능력을 키운다.

Basic understanding of algorithms and characteristics of algorithm types are learned. The types of algorithms are classified into divided-and-conquer, dynamic programming, greedy algorithms, branch-and-bound, searching, etc and each characteristic is understood. In addition, we look at the basic complexity issues. After completing this course, students will develop the ability to design algorithms to new problems.

- **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.

- **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 exit 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.

- **졸업논문 (Graduation Thesis)**

컴퓨터공학과는 캡스톤디자인과 졸업프로젝트를 이수하는 것으로 졸업을 위한 “졸업논문” 합격 여부를 결정한다.

Bachelor of Engineering in Computer Engineering decides acceptance of graduation thesis by completing Capstone Design and Graduation Project.

- **졸업프로젝트(Graduation Project)**

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 인턴십, 실제 프로젝트 수행, 연구논문 작성 등이다. 캡스톤디자인을 수강한 학생들이 다음 학기에 수강하는 과목으로 캡스톤디자인에 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. The students are advised to continue their researches and studies based on the mid-result of the Special Topics in Capstone Design, present their final research results, and submit the corresponding reports.

- **캡스톤디자인 (Capstone Design)**

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 인턴십, 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.

- **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.

- **디자인적사고 (Design Thinking)**

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

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.

- **오픈소스SW개발방법및도구 (Opensource Software Development Methods and Tools)**

소프트웨어 개발을 위한 방법론과 도구에 대해서 오픈소스 소프트웨어를 기반으로 학습하며, 리눅스 운영체제에 대한 프로젝트를 수행한다. 소프트웨어 개발 방법론은 최근에 가장 많이 쓰이고 있는 Agile 개발 방법에 대해서 학습하며, 이를 소프트웨어 산업에서 적용하는 다양한 활용 방법에 대해서 학습한다. 소프트웨어 개발 도구는 오픈소스 소프트웨어를 중심으로 하여, 소프트웨어의 설계, 개발, 시험, 검증, 팀작업 등의 전반적인 부분에서 활용 가능한 도구에 대해서 학습한다. 또한 리눅스 운영체제를 한 학기 동안 직접 설치하고 활용함으로써, 이후 소프트웨어융합학과의 교육과정에서 리눅스를 활용하는 기본 능력을 함양한다.

Learn about methodologies and tools for software development, and work on projects for the Linux operating system. The software development methodology learns about the agile development method, which is used most recently, and the various application methods applied to the software industry. Software development tools focus on open source software, and learn about tools that can be used in the overall aspects of software design, development, testing, verification, and team work. In addition, by installing and using the Linux operating system directly for one semester, students will develop the basic ability to use Linux in the course of software convergence.

- **기계학습 (Machine Learning)**

기계학습은 지능적인 응용 시스템을 구축할 수 있는 기반이 되었다. 본 과목에서는, 기본적인 기계학습 알고리즘을 소개하는 것을 시작으로, 실제 응용을 중심으로 다양한 기술과 이론을 소개한다. 이러한 알고리즘의 사용 사례와 제한 사항들에 대한 논의를 진행 하고, 프로그래밍을 통해 훈련과 검증 과정을 구현한다.

Machine learning has become a pillar on which you can build intelligent applications. This course will begin with the introduction of basic machine learning algorithms, and various techniques and theories are introduced with a focus on practical applications. The use cases and limitations of these algorithms will be discussed, and training and validation will be implemented with programming language.

- **인공지능프로그래밍 (AI Programming)**

이 과목에서는 인공지능 수학에서 배우는 인공지능을 위한 수학적 개념을 실제로 컴퓨터로 구현한다. 이를 위해 NumPy, Tensorflow, Pytorch와 같은 기본적인 파이썬 라이브러리를 학습하고, 기본적인 신경망을 구현할 수 있는 능력도 기른다.

This course aims at giving how to implement mathematical concepts for artificial intelligence. For this, the students learn basic Python libraries such as NumPy, Tensorflow, and Pytorch, and they also study how to construct some

simple neural networks using the libraries.

- **고급딥러닝 (Advanced Deep Learning)**

기본적인 CNN, RNN 등의 딥러닝 토픽을 학습한 후, Transformer 등의 최신 딥러닝 토픽을 배운다. 컴퓨터비전과 자연언어처리에 대한 딥러닝 응용을 통해 실질적 문제 해결에 딥러닝을 적용한다.

This course offers hot topics of deep learning such as Transformer after providing some basic topics about deep learning including CNN and RNN. It allows the students to apply the learned deep learning techniques to real world problems in computer vision and natural language processing.

- **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.

- **실감미디어컴퓨팅기초 (Immersive Media Computing Fundamental)**

본 교과목은 실감미디어 기술 및 콘텐츠 구현을 위해 필요한 컴퓨팅 기술을 대한 기초를 다루는 컴퓨팅 개론 수업이다. 본 수업은 컴퓨터 기반 실감미디어 도구들을 사용하기 위해서 이해해야 하는 컴퓨팅의 기초 지식 및 컴퓨팅적 사고를 기반으로, 실감미디어 도구들에서 사용하는 여러 컴퓨팅 언어들에 대한 개론적인 코딩 학습을 다룬다.

This course is an introductory computing class dealing with the basics of computing technology necessary for realizing immersive media technology and contents. This class covers the basic knowledge of computing and computational thinking that must be understood in order to use computer-based immersive media tools as well as an introductory coding training for various computing languages used in the tools.

- **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.

- **컴파일러 (Compiler)**

컴퓨터과학의 근간을 이루는 컴파일러의 개념을 배운다. 구체적으로, 어휘분석, 구문분석, 언어의 문법적 표현과 분류, 유한상태 기계, push-down 자동장치, 정규언어, 파싱 기법의 요소들을 이해하고 응용할 수 있는 능력을 키운다.

Students learn the concept of a compiler, which is the basis of computer science. Learn the relationship between compilers and formal languages. Specifically, it develops the ability to understand and apply the elements of lexical analysis, syntax analysis, grammatical expression and classification of language, finite state machine, push-down automata, regular language, and parsing technique.

- **메타버스시스템 (Metaverse System)**

메타버스 시스템은 인간의 오감과 관련된 데이터를 처리하는 방법에 대한 기본적인 방법론을 다루는 과정이다. 본 과정은 메타버스

시대에 멀티미디어 데이터를 서로 교환하는 메타버스 시스템을 구현하기 위한 각 기본 기능(이미지, 그래픽, 애니메이션, 비디오, 사운드, 오디오)에 대한 개념적 접근을 수행한다.

Metaverse system is the process of dealing with basic methodologies on how to handle data related to human five senses. This course conducts a conceptual approach to each basic function (image, graphics, animation, video, sound, audio) to implement a metaverse system that exchanges multimedia data with each other, in the metaverse-era.

- **프로그래밍언어론 (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 various language constructs, 2) examining the design issues in these constructs, and 3) comparing design alternatives.

- **SW스타트업비즈니스 (SW Start-Up Business)**

실리콘벨리의 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 기반으로 한 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.

- **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.

- **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.

- **SW스타트업프로젝트 (SW Start-Up Project)**

산업사회가 고도화되면서 창업의 비중이 점차 높아지고 있는 추세이다. 오픈소스SW를 기반으로 창업아이템 선정, 사업기획, 제품샘플 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.

- **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.

- **실전기계학습 (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.

- **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.

- **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 introduces 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.

- **메타버스데이터처리 (Metaverse Data Processing)**

메타버스 시대에는 모든 데이터가 100% 압축된 상태로 서로에게 전달된다. 압축된 데이터 중 가장 큰 데이터는 메가 데이터인 비디오 데이터이다. 비디오 데이터는 2K-4K에서 8K로 매우 사실적인 느낌을 주며 해상도는 NK로 확장된다. 상상할 수 없을 정도로 많은 양의 데이터를 전송하고 처리하기 위해서는 가장 기본적인 압축을 익힐 필요가 있다. 본 과정은 이미지/오디오/비디오, 손실/무손실 비디오 압축 방법, JPEG, MPEG 등 국제 표준의 사례 연구 등 메타버스 데이터를 교환하기 위한 기본 알고리즘에 대한 개념적 연구를 수행한다.

In the Metaverse-era, all data is delivered to each other in a 100% compressed state. The largest data among the compressed data is video data as mega data. Video data goes from 2K-4K to 8K for ultra-realistic feel, and resolution is extended to NK. In order to transmit and process an unimaginably large amount of data, it is necessary to learn the most basic compression. This course conducts a conceptual study on the basic algorithms for exchanging metaverse data such as image/audio/video, loss/lossless video compression methods, the case studies of international standards such as JPEG, MPEG, etc.

- **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 디자인하는 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 다룰 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.

- **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.

- **빅데이터프로젝트 (Big Data Project)**

전통적으로 우리는 주로 RDBMS에서 구조화된 데이터로 작업해 왔지만, 현재에는 금융, 병원, 범죄, 날씨 등 다양한 분야에서 다양한 유형의 데이터들이 발생하고 있다. 수많은 데이터를 처리하기 위해 데이터 엔지니어 또는 과학자들은 빅데이터 솔루션의 큰 그림을 이해해야 한다. 빅데이터 솔루션에는 데이터 수집, 데이터 스토리지, 데이터 분석 및 데이터 시각화 등의 몇 가지 단계가 있다. 위의 단계를 이해하기 위해 학생들은 이 수업에서 빅데이터 프로젝트를 설계하고 다양한 관련 프레임워크를 관리하는 방법을 배우게 된다.

Traditionally, we have been working with structured data mainly from RDBMS, but in this century, at on of different type logs are coming from various fields, e.g., finance, hospital, crime, weather, etc. In order to handle the abundant logs, data engineers or scientists should understand a big picture of a Big Data solution. There are a few of steps in the Big Data solution : Data Acquisition, Data Storage, Data Analysis and Data Visualization. To understand the steps above, students will learn how to design bigdata projects and manage a variety of related frameworks.

- **클라우드프로젝트 (Cloud Project)**

인공지능의 보편화를 필두로, 4차 산업혁명이 시작 되었다. 4차 산업 혁명의 대표적인 기술에는 인공지능, 사물 인터넷, 빅데이터, 블록체인 등이 있다. 클라우드 컴퓨팅은 위의 기술들이 확장하기 위해서 뿌리가 되는 중요한 기술이다. 본 수업에서는 클라우드 컴퓨팅 개념을 시작으로, 산업체에서 활발하게 사용하고 있는 클라우드 컴퓨팅 서비스인 아마존 웹 서비스를 실습한다. 더불어 클라우드 연계 프로젝트를 위해서 IoT(사물 인터넷) 등을 활용하여 연계 프로젝트를 진행한다.

With the generalization of artificial intelligence, the 4th industrial revolution has begun. Representative technologies of the 4th industrial revolution include artificial intelligence, the Internet of Things, big data, and block chains. Cloud computing is an important technology that is the root of the above technologies to expand. In this class, starting with the concept of cloud computing, we practice Amazon Web Service, a cloud computing service that is actively used in industry. In addition, for cloud-linked projects, we use (IoT (Internet of Things) or AI etc) to carry out the linked project.

- **최신기술콜로키움1 (Latest Technology Colloquium 1)**

소프트웨어 기반의 융합 분야에 대한 최신 기술을 해당 분야 전문가를 초청하여 듣도록 한다. 트랙별로 학교 내부와 산업체 전문가

를 초청하게 되며, 학생들은 강연에 대한 요약과 함께, 본인의 진로와 연관된 의견을 보고서로 작성하여 제출한다. 성적은 출석과 보고서에 기반하여 Pass 혹은 Fail로 판단된다.

Invite experts from the field to hear the latest technology in the field of software-based convergence. Each track will invite school professionals and industry experts. Students will submit a summary of the lecture, along with a report on their career. The grade is judged to be Pass or Fail based on attendance and report.

- **AIoT디지털시스템 (AIoT Digital System)**

소프트웨어와 하드웨어를 포함하는 임베디드 시스템 환경에서 AI 기반 임베디드 응용에 대한 개발 개념과 이를 구현할 수 있는 FPGA 기반 임베디드 레퍼런스 보드 환경의 사용 및 구현 방법을 다룬다. 이 수업에서는 AI 알고리즘과 이를 이용한 응용 기술에 대해 배우고 이를 임베디드 환경내의 구현 방법을 다룬다.

This course provides the development concept for AI-based embedded applications in the embedded system environment including software and hardware, and how to use and implement the FPGA-based embedded reference board environment that can implement it. In this class, we learn about AI algorithms and applied technologies using them, and how to implement them in an embedded environment.

- **AIoT소프트웨어 (AIoT Software)**

컴퓨터 공학 기초 과목의 이해를 바탕으로 Embedded System에서의 AIoT 프로그램 개발을 경험한다. 아두이노 프로그래밍에서 시작하여, FreeRTOS를 활용한 RTOS를 다루고, 라즈베리파이에서 커널 프로그래밍을 포함한 임베디드 리눅스를 경험한다.

With a basic knowledge on computer engineering, this course gives an experience of AIoT on embedded system. In this course, we start with Arduino programming and develop RTOS programs on FreeRTOS.

- **컴퓨터비전 (Computer Vision)**

본 과목은 사람이 시각 정보를 이용하여 지능적인 판단과 행동을 하는 것과 마찬가지로, 컴퓨터(기계)도 사람과 같이 시각 정보를 획득, 처리하고 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.

- **블록체인 (Blockchain)**

Web3시대를 열어가게 하는 기반기술인 블록체인에 대해 알아본다. 비트코인 부터 시작된 블록체인의 역사와 이더리움을 거쳐 다양한 Web3 application이 어떤 식으로 진화했는지 앞으로 어떻게 발전해 나갈 수 있을지 알아본다. 또한 간단한 블록체인 프로그래밍을 통해 Web3 application개발을 시작해 본다.

Learn about blockchain, the underlying technology for the Web3 era. Learn about the history of blockchain, starting with Bitcoin, and how various Web3 applications have evolved through Ethereum, and how they will evolve in the future. You will also learn how to start developing Web3 applications through simple blockchain programming.

- **AI네트워킹 (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 AI기반 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.

- **AIoT네트워크 (AI Internet of Things Network)**

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

This course teaches how digital transformation is realized through three key concepts: the Internet of Things(IoT) and digitization and Artificial Intelligence(AI). The concept of what the AIoT is, what the AIoT does, and how AIoT digitalization affects the fourth industrial society are discussed from technical points of view.

- **모바일/웹서비스프로그래밍 (Mobile/Web Service Programming)**

오픈소스 프로그래밍 개발 환경에서 모바일 및 웹서비스 프로그래밍의 기본 개념을 배우고 여러 응용 프로그래밍을 구현함으로써 실제 업무에 적용 가능한 실무능력을 향상시킨다,

This course is designed for learning a fundamental concept of mobile programming and web-service programming based on the open source development environment.

- **모바일/웹서비스프로젝트 (Mobile/Web Service Project)**

모바일 프로그래밍의 응용 프로그래밍을 구현함으로써 실제 업무에 적용 가능한 실무능력을 향상시키고, 최신 적용가능 라이브러리 및 테크닉에 대해 학습한다. 또한 HTML/CSS를 기본으로 하여 javascript를 이용한 Node.js 등과 같은 각종 웹 프레임워크를 이용한 웹서비스 백엔드 및 프론트엔드를 개발하는 실전 프로젝트를 진행한다.

This course is designed for developing practical applications of mobile programming with the latest libraries and techniques. Also, this course performs practical projects to develop web service front-end and back-end using HTML/CSS, javascript, and the diverse web frameworks for Node.js.

- **소프트웨어보안 (Software Security)**

IT시장이 발달함에 따라 소프트웨어의 복잡성과 규모가 나날이 커지고 있다. 소프트웨어의 복잡도가 높아질수록 Use-After-Free 와 같은 메모리 오류, 권한제어등과 관련한 논리적인 취약점 등 다양한 보안 문제점들이 발생할 수 있는데, 본 과목에서는 소프트웨어의 개발프로세스 및 관리/운영과정을 포함하여 이러한 문제가 발생하는 원인 및 대응방안을 다룬다. 또한 수강생들의 기술적 이해도를 높이기 위하여 공격자의 입장에서 가상의 보안취약점을 다뤄보는 실습도 병행한다.

Software complexity and its development scale is constantly growing as the IT market expands. Software vulnerabilities such as Use-After-Free memory corruption, logic errors are more likely to exist as the software codebase becomes larger and sophisticated. Addressing such recent issues, this class covers a broad spectrum of security problems that could exist in software development and IT operations (DevOps); and how to mitigate them. In addition to theoretical discussion, the class provides various lab exercise regarding software vulnerabilities for practical/technical experiences.

- **웹보안 (Web Security)**

최근의 소프트웨어는 클라우드 및 웹 애플리케이션과 연동되는 형태로 개발되는 경우가 많다. 본 과목에서는 Flask/Django 등에 기반하는 웹 애플리케이션부터, Nginx/Apache 와 같은 웹서버, 및 방화벽 시스템들이 웹 브라우저와 함께 어떻게 상호작용하며 구동되는지 이해하고 그러한 시스템에서 존재할 수 있는 보안문제들을 다룬다. 특히 Server Side Request Forgery (SSRF) 와 같은 몇몇 웹 보안 취약점들은 서버들을 대상으로 하는 클라우드 서비스의 규모가 커짐에 따라 그 위험도가 더욱 높아지고 있는데, 이러한 공격 기술에 관한 실습도 병행한다.

Recent software development often involves interaction with web-application and cloud systems. This class covers security problems in web-applications such as Flask/Django, web servers such as Nginx/Apache and network firewalls

in terms of their interaction with client-side web browsers. In particular, Server Side Request Forgery (SSRF) is becoming a more serious threat than before due to the prevalence of cloud-based services. Addressing such recent software security trends, this class provides SSRF-related lab exercises.

• **독립심화학습1/2(컴퓨터공학과) (Individual In-Depth Study 1/2)**

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

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 특허화하고, 최종 출원하는 절차를 수행해 본다. 특허 작성을 희망하는 학생은 계획서와 실적서를 제출하고, 실질적인 작성 여부 등에 기반하여 지도교수로부터 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 지도교수로부터 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 지도교수로부터 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.

• **연구연수활동1/2 (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.

• **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.

• **게임프로그래밍입문 (Introduction to Game Programming)**

[수강 대상 : 프로그래밍 유경험자(Python/C/C++)] 간단한 GUI(Graphic User Interface) 환경의 기초적인 게임부터 오픈 소스를

활용한 고전적인 아케이드 게임, 플랫폼 게임 등에 대한 코드 레벨을 이해한 후, 직접 설계 및 개발하는 과정을 통하여 게임 개발에 대한 프로세스를 이해한다. 다양한 환경의 게임을 경험하기 위하여, 충돌 검출 및 반응, 게임 물리, 3D 그래픽, 게임 인공 지능 등에 대한 기초를 경험한다.

Understand the code level of basic game of simple GUI(Graphic User Interface) environment, classic arcade game, platform game using open source, and understand the process of game development through the process of designing and developing. In order to experience games in various environments, students will learn the basics of collision detection and reaction, game physics, 3D graphics, game artificial intelligence, and so on.

• 게임엔진기초 (Game Engine Basics)

[수강 대상 : 기초적인 프로그래밍(C++, C#, Python 등) 경험이 있는 자] 본 수업은 이론 수업과 실습 수업으로 이루어져 있다. 이론 수업의 목적은 게임 엔진의 전체적인 구조와, 이를 이해하기 위해 필요한 기초 이론들을 학습하는 것이 목표이다. 실습 수업의 목적은 상용 엔진을 실제로 사용하면서 엔진에 대한 이해를 심화하는 것이다.

This class comprises of theory part and a practice part. Through the theory part, students will understand the architecture of a game engine and the basic theories used in a game engine. Through the practice part, students will deepen their understanding of the game engine by actually using the commercial game engine.

• 마이크로서비스프로그래밍 (Microservice Programming)

데이터센터에서 마이크로서비스를 개발하고 운영하는 개발 방법론과 도구에 대한 이론을 이해하고 실습을 수행한다. 대표적인 기술로서 클라우드 컴퓨팅, 컨테이너/오케스트레이션 기술, 마이크로 서비스, DevOps에 대해서 이해하고, 실습을 통해서 직접 데이터센터의 소프트웨어 개발 환경을 구축하고, 시험/운영하는 능력을 함양한다.

Understand and practice theories on microservice development methodologies and tools used in data centers. As a representative technology, students will understand cloud computing, container/orchestration technology, microservices, DevOps, and build the ability to directly build and test/operate the software development environment of the data center.

• 회로와신호 (Circuits and Signals)

이 과목을 통해 학생들은 전자회로 기본소자의 동작원리와 기능을 익히고 기초회로 해석 능력을 습득한다. 또한 연속 및 이산 신호와 시스템의 수학적 표현기법과 변환기법을 배운다. 수업에서 다루는 개념은 전하, 전류, 전압, 저항, Ohm의 법칙, Kirchoff의 법칙, 직렬/병렬회로, RC/RL/RLC 회로, 연속 및 이산 신호의 Fourier 변환, Laplace/z 변환, 샘플링, 양자화, 필터 등을 포함한다.

In this course, students learn the basic elements and principles of electric circuits, mathematical representations on continuous and discrete signals and systems, and signal transforms. The course deals with electric charge, currents, voltages, resistance, Ohm's law, Kirchoff's law, series/parallel circuits, RC/RL/RLC circuits, Fourier transform of continuous and discrete signals, Laplace/z transform, sampling, quantization, and filters.

• 최신기술콜로키움2 (Latest Technology Colloquium 2)

소프트웨어 기반의 융합 분야에 대한 최신 기술을 해당 분야 전문가를 초청하여 들도록 한다. 트랙별로 학교 내부와 산업체 전문가를 초청하게 되며, 학생들은 강연에 대한 요약과 함께, 본인의 진로와 연관된 의견을 보고서로 작성하여 제출한다. 성적은 출석과 보고서에 기반하여 Pass 혹은 Fail로 판단된다.

Invite experts from the field to hear the latest technology in the field of software-based convergence. Each track will invite school professionals and industry experts. Students will submit a summary of the lecture, along with a report on their career. The grade is judged to be Pass or Fail based on attendance and report.

• 게임그래픽프로그래밍 (Game Graphic Programming)

[수강 대상 : C++가능자] 디지털 게임 제작에 있어서 가장 기본이 되는 게임 그래픽 개발 능력을 학습한다. 특히 그래픽 관련 선형대수(투영변환 등), 게임에 사용되는 물리학(이동, 회전, 관성, 충돌) 등의 이론적 지식을 활용하여 관련 라이브러리(OpenGL/DirectX,

GLSL/HLSL) 기반으로 실습을 진행한다.

In this course, students will learn game graphic development skills which are the most basic in digital game production. In particular, students will use the theoretical knowledge of linear algebra(projection transformation, etc.) related to graphics, and physics(movement, rotation, inertia, collision, etc.) used in games to practice based on related libraries(OpenGL/DirectX, GLSL/HLSL).

• 게임인터랙티브테크놀로지 (Game Interactive Technology)

[수강대상 : C++ 가능자] 게임에 활용되는 다양한 입출력 장치의 구동 및 활용에 대해서 학습하고 이를 확장하여 차세대 입출력 장치의 원리와 배경 이론에 대해서 학습한다. 기본적인 마우스 및 키보드와 같은 장치에서 시작하여 얼굴, 몸통, 손의 인식에 의한 입력 장치의 구동원리를 학습하고 인간-컴퓨터 상호작용에서 다루는 관련 이론 및 인터랙션 테크닉에 대해서 학습한다. 기존의 입출력 장치 외에 더 사실적인 게임 플레이의 경험을 제공하기 위해 도입되고 있는 인식 기술 및 관련 이론을 학습하고 이를 실제 게임에 적용해본다. 두번째로 시각, 청각, 촉각으로 대표되는 출력 장치들에 대해서 감각별 생체 기작 및 인지 이론을 학습하고 게임에 적용하며 이에 대한 평가를 수행하는 방법을 실습한다.

This course introduces the basic operation principles of input and output devices which extends toward the understanding of future I/O devices. Students will learn the fundamental theories of Human-Computer Interaction to design interaction techniques for computing and understand the application of physiological and perceptual backgrounds underneath the software and hardware design. In summary, this course is designed to learn how to provide an optimal experience using motion-based input devices utilizing various sensor technologies, immersive sensory output devices using wearable technology, and input/output devices.

• 가상/증강현실이론및실습 (Virtual and Augmented Reality Programming) (*게임콘텐츠)

[수강대상 : C++ 가능자] 디지털 게임 디자인 및 3D 그래픽스 그리고 게임엔진에 대한 이해를 바탕으로 AR(Augmented Reality) 이나 HMD(Head Mount Display) 기반의 VR(Virtual Reality) 콘텐츠 구현을 위한 이론적, 기술적 능력을 배양하고 이를 직접 제작한다.

Culture theoretical and technical skills for the AR(Augmented Reality) and HMD(Head Mount Display) based on VR (Virtual Reality) can be implemented directly and making it.

• 로봇프로그래밍 (Robot Programming)

로봇 운영체제(ROS)를 통해, 로봇의 하드웨어, 소프트웨어 플랫폼, 핵심 기술 애플리케이션 소프트웨어의 구조를 이해하고 직접 다뤄보도록 한다. 시뮬레이션 기반의 로봇 조작 알고리즘을 학습하고 SLAM을 통한 자율주행 기술의 이론과 실습을 다룬다.

In this course, students understand the structure of hardwares, software platforms, and core application softwares of future cars and robots based on Robot Operating System(ROS). Students learn and develop simulation-based robot manipulation algorithms and theory and practice of autonomous driving technology through SLAM.

• 3D데이터처리 (3D Data Processing)

자율 주행 자동차 및 지능 로봇에 활용되는 로봇 비전에 대해서 배운다. 카메라 및 삼차원 라이더등의 비전 센서의 원리 및 획득된 데이터를 처리하는 방식에 대해서 공부한다. 3차원 기하학을 기반으로 하여 영상들 또는 3차원 포인트 클라우드로부터 특징 추출 및 이를 활용한 3차원 복원 방식들에 대해서 다룬다.

In this course, we will deal with 3D robot vision used in autonomous vehicles and intelligent robots. This course will study the principles of vision sensors such as cameras and 3D lidar and how to process the data. Based on 3D geometry, we deal with feature extraction from images or 3D point clouds and 3D reconstruction methods using sensor data.

• 풀스택서비스프로그래밍 (Full-Stack Service Programming) (*공통)

한번 작성한 코드로 다양한 운영체제와 디바이스에서 실행되는 프로그램을 개발하는 크로스 플랫폼(Cross-Platform) 기술을 이용하

여 모바일 앱, 웹 서비스, 데스크탑 어플리케이션 및 서버 프로그램을 개발하는 방법을 배우는 과목이다. 이를 위해서 Dart 언어와 Flutter 프레임워크를 사용하며, 필요시 Adobe-XD를 통한 UI/UX 디자인 결과물을 Flutter 프레임워크와 연동하는 방법을 배우도록 한다.

This course is to learn how to develop mobile apps, web services, desktop applications, and server programs using cross-platform technology that develops programs that run on various operating systems and devices with code developed once. For this, Dart language and Flutter framework are used, and if necessary, learn how to link UI/UX design results through Adobe-XD with Flutter framework.

- **인공지능과게임프로그래밍 (AI and Game Programming)**

본 강의는 게임 프로그래밍을 통해 게임 요소, 자료 및 신호 분석, 인공지능에 대한 실질적인 프로그래밍 개념들을 소개하고 실습한다. 신호 및 영상 처리, 데이터 분석, 기계 학습, 게임 비전 등에 대한 기초 이론이 인공지능의 다양한 분야에 어떻게 활용되는지 공부하고 프로그래밍으로 구현 및 분석한다.

This course introduces and provides practical programming concepts for game components, data and signal analysis, and artificial intelligence using game programming. It explores fundamental theories of signal and image processing, analysis, machine learning, and game vision, focusing on their applications within diverse fields of artificial intelligence.

- **풀스택서비스네트워킹 (Full-Stack Service Networking) (*공통)**

전통적인 컴퓨터네트워크 기술을 학습한 후, 최신 네트워킹 기술을 익힌다. 전통적인 컴퓨터네트워크 기술은 OSI(Open System Interconnection) 7계층 구조에 기반하여, Ethernet, WLAN, TCP/IP 등 인터넷 중심의 기술을 학습한다. 최신 네트워킹 기술은 소프트웨어융합학과 트랙에 맞춰서, 실제 산업에서 사용하는 실용적인 최신 네트워킹 기술을 다룬다. 특히, 트랙별 캡스톤디자인 및 소프트웨어융합 캡스톤디자인 등 네트워킹 기술 개발이 필요한 학생들은 수업을 들으며 실전적인 네트워킹 기술에 대한 조연구 최신크 네트워킹 소프트웨어 기술에 대한 설명을 학습할 수 있다.

After learning traditional computer network skills, discover the latest networking skills. Traditional computer network technology is based on the OSI(Open System Interconnection) 7-layer structure and understands Internet-oriented technologies such as Ethernet, WLAN, and TCP/IP. In line with the software convergence department and track, the latest networking technology covers the latest practical networking technology used in real industry. In particular, students who need networking, such as capstone design and software convergence by track, can learn advice on practical networking technology and explanations on the latest networking software technology while taking classes.

- **자연언어처리 (Natural Language Processing) (*공통)**

이 과목은 자연언어처리(NLP)의 다양한 문제들을 다룬다. 따라서, 띄어쓰기, 형태소 분석 등과 같은 전통적인 자연언어처리 토픽 외에도 감성분석, 요약, 챗봇과 같은 응용 문제들도 다루고자 한다. 이 과목은 기계학습 기반의 자연언어처리 기법에 대하여 학생들이 이 심도 있는 이해를 하게 하여 추후 학생들이 자신만의 문제에 그 기법을 적용할 수 있도록 하는 것을 목표로 한다.

This course covers a wide range of tasks in natural language processing(NLP). As well as traditional and basic topics in NLP such as word spacing and morphological analysis, its coverage includes application-related topics such as sentiment analysis, summarization, and chat-bot. This course aims at providing in-depth understanding of recent machine learning-based NLP techniques to allow students to apply the techniques to their own language-related tasks.

- **강화학습 (Reinforcement Learning) (*로봇·비전)**

본 강의에서는 심층 신경망을 이용한 강화학습의 이론에 대해서 공부하고 실습을 통해 직접 구현해본다. 강화학습의 기본 개념, 벨만 방정식, 마르코프 결정 과정 등을 설명하고, 정책 함수 및 가치 함수 기반의 학습 방식에 대해서 공부한다. 시뮬레이션 기반으로 최신 심층 강화학습 알고리즘을 직접 구현해 본다.

In this course, we will study the theory of reinforcement learning using deep neural networks and implement it directly through practice. Basic concepts of reinforcement learning, Bellman's equation, Markov decision process, etc. are explained, and learning methods based on policy functions and value functions are studied. We will cover recent deep

reinforcement learning algorithms, and implement them on simulated environments.

- **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.

[별표6]

빅데이터 분석 마이크로디그리 이수 체계도

1. 마이크로디그리명(영문): 빅데이터 분석 마이크로디그리 (Bigdata Analysis Micro Degree)

2. 마이크로디그리 목표: 빅데이터 분석에 필요한 데이터마이닝 및 빅데이터 관리를 위한 프로그래밍 전문 인재 양성

3. 마이크로디그리 소개

- ① 빅데이터 분석을 위한 기본 개념 습득
- ② 빅데이터 마이닝과 머신러닝 기술 및 이러한 기술들을 위한 프로그래밍
- ③ 빅데이터의 시각화 기술 및 이를 위한 도구 사용법과 프로그래밍

4. 마이크로디그리 이수 역량과 자격

- ① 본인의 아이디어를 조기에 사업화/창업하고 싶은 소프트웨어 비전공자
- ② 빅데이터를 위한 프로그래밍에 관심이 있는 소프트웨어 전공자
- ③ 소프트웨어융합학과 '웹/파이선프로그래밍' 사전 수강 권장
- ④ 2학기 이상 이수한 재학생(편입생은 1학기 이상)에게 마이크로디그리의 신청자격을 부여하며, 마이크로디그리 이수를 희망하는 자는 학기별 소정의 기간에 신청 후 이수하면 된다. (단, 수업연한 초과자는 신청할 수 없음)
- ⑤ 마이크로디그리는 최대 3개까지 신청 및 이수할 수 있다. 단, 특정 학부(과) 소속학생의 신청이 제한될 수 있다.
- ⑥ 이수 중인 마이크로디그리를 포기하고자 하는 자는 학기별 소정의 기간에 마이크로디그리 포기 신청을 해야 한다.
- ⑦ 최종 이수 확정 된 마이크로디그리는 포기할 수 없다.
- ⑧ 마이크로디그리 미이수자 중 졸업요건을 충족한 자는 마이크로디그리를 위해 졸업유예를 할 수 없다.

5. 빅데이터 분석 마이크로디그리 이수학점

학부(과)명	학수번호	이수구분	교과목명	학점
컴퓨터공학부/소프트웨어융합학과	CSE434	전공선택	빅데이터프로그래밍	3
컴퓨터공학부	AI3004	전공선택	빅데이터마이닝	3
소프트웨어융합학과	SWCON425	전공선택	데이터사이언스및시각화	3
총계				9

6. 마이크로디그리 이수방법

- ① 마이크로디그리 신청자는 신청한 마이크로디그리의 이수체계에 따라 교과목을 이수하여야 한다.
- ② 마이크로디그리 신청 전에 해당 과목을 이수하였을 경우 이수학점으로 인정한다.
- ③ 마이크로디그리로 인정된 교과목은 전공 및 교양학점으로 중복 인정한다.
- ④ 이수 중인 마이크로디그리를 포기할 경우 기 이수한 교과목은 전공 및 교양, 또는 일반선택 학점으로 인정한다.
- ⑤ 마이크로디그리 이수 후 마이크로디그리 수료증을 발급받을 수 있으며, 성적증명서에 그 사실을 기재한다.

메타버스 마이크로디그리 이수 체계도

1. 마이크로디그리명(영문): 메타버스 마이크로디그리 (Metaverse Micro Degree)

2. 마이크로디그리 목표: 메타버스에 필요한 이론 및 영상데이터 관리 전문 인재 양성

3. 마이크로디그리 소개

- ① 메타버스 실현을 위한 비디오 데이터 압축 기술
- ② 메타버스 시스템 구현을 위한 이미지, 그래픽스, 애니메이션, 비디오, 사운드, 오디오를 포함하는 인간의 오감을 위한 멀티미디어 신호처리 기술
- ③ 메타버스의 거대한 데이터를 전달하기 위한 네트워크 기술

4. 마이크로디그리 이수 역량과 자격

- ① 본인의 아이디어를 조기에 사업화/창업하고 싶은 소프트웨어 비전공자
- ② 메타버스 산업에 관심이 있는 소프트웨어 전공자
- ③ 영상데이터 처리 또는 최신 네트워크 기술에 관심 있는 전공/비전공자
- ④ 컴퓨터공학과 '객체지향프로그래밍' 사전 수강 권장
- ⑤ 2학기 이상 이수한 재학생(편입생은 1학기 이상)에게 마이크로디그리의 신청자격을 부여하며, 마이크로디그리 이수를 희망하는 자는 학기별 소정의 기간에 신청 후 이수하면 된다. (단, 수업연한 초과자는 신청할 수 없음)
- ⑥ 마이크로디그리는 최대 3개까지 신청 및 이수할 수 있다. 단, 특정 학부(과) 소속학생의 신청이 제한될 수 있다.
- ⑦ 이수 중인 마이크로디그리를 포기하고자 하는 자는 학기별 소정의 기간에 마이크로디그리 포기 신청을 해야 한다.
- ⑧ 최종 이수 확정 된 마이크로디그리는 포기할 수 없다.
- ⑨ 마이크로디그리 미이수자 중 졸업요건을 충족한 자는 마이크로디그리를 위해 졸업유예를 할 수 없다.

5. 메타버스 마이크로디그리 이수학점

학부(과)명	학수번호	이수구분	교과목명	학점
컴퓨터공학부	CSE324	전공선택	메타버스시스템	3
컴퓨터공학부/소프트웨어융합학과	CSE430	전공선택	메타버스데이터처리	3
소프트웨어융합과	SWCON311	전공선택	게임그래픽프로그래밍	3
총계				9

6. 마이크로디그리 이수방법

- ① 마이크로디그리 신청자는 신청한 마이크로디그리의 이수체계에 따라 교과목을 이수하여야 한다.
- ② 마이크로디그리 신청 전에 해당 과목을 이수하였을 경우 이수학점으로 인정한다.
- ③ 마이크로디그리로 인정된 교과목은 전공 및 교양학점으로 중복 인정한다.
- ④ 이수 중인 마이크로디그리를 포기할 경우 기 이수한 교과목은 전공 및 교양, 또는 일반선택 학점으로 인정한다.
- ⑤ 마이크로디그리 이수 후 마이크로디그리 수료증을 발급받을 수 있으며, 성적증명서에 그 사실을 기재한다.

[별표7]

컴퓨터공학과 전공능력

1. 대내외 환경분석

구분	세부 구분		내용
외부	필수	사회 흐름	4차산업혁명의 중심 분야인 인공지능에 대한 사회적 관심과 관련 기술에 대한 수요가 점차적으로 늘고 있음
		산업 수요	국립학술원은 컴퓨터 공학에 대한 수요 증가에 따른 대학 기관의 대처 방안을 살펴본 보고서 발표. 미국에서 컴퓨터 공학 전공자에 대한 수요가 급격히 증가하여, 컴퓨터 공학 전공 졸업생의 배출 속도에 비해 훨씬 빠른 속도로 컴퓨터와 관련한 일자리가 발생. 20년 전 대비 컴퓨터 관련 직업의 수는 20배 가까이 증가해 관련 전공 학부 졸업자의 증가 속도를 2배 앞지름.
	선택	문헌 분석	
		타 대학 우수사례	
내부	학과(전공) 발전전략		<ul style="list-style-type: none"> - 코딩 능력을 중심으로 한 컴퓨터공학 기초 역량 교육 강화 - 글로벌 경쟁력을 갖춘 전문가 양성 - 학생에 대한 밀착지도를 통한 학생 컨설팅 강화
	재학생 역량분석		<ul style="list-style-type: none"> - 기본-발전-심층 소프트웨어 엔지니어링 커리큘럼을 통한 전문성 있는 인재 확충 - 산업체 관련 프로젝트 수행을 통한 실무능력 함양
	의견 수렴 및 요구 분석	재학생	<ul style="list-style-type: none"> • 방식: 인터뷰를 통한 사례 조사 • 조사대상: 학부재학생 (3학년, 4학년) • 시사점: 새로운 기술들이 많이 생성되고 있어 이를 반영할 수 있는 과목의 개설이 필요함.
		졸업생	<ul style="list-style-type: none"> • 방식: 인터뷰를 통한 사례 조사 • 조사대상: 학부졸업생 (직장인, 대학원생) • 시사점: 컴퓨터공학과 관련된 새로운 프로그래밍 언어, 기술들이 빠르게 생성되고 있으므로 교육과정을 좀 더 유연하고 적응적으로 개편하여 대응해야함.
		교수	<ul style="list-style-type: none"> • 방식: 인터뷰를 통한 사례 조사 • 조사대상: 학부 교수 • 시사점: 교과목의 다양성 확보 및 범위 확대를 위한 교원 충원이 필요함.
		산업체	<ul style="list-style-type: none"> • 방식: 인터뷰를 통한 사례 조사 • 조사대상: 협업 기업의 CEO 및 CTO • 시사점: 산업체에서 바로 적용할 수 있는 기술을 배양하는 것이 학생들의 문제해결 능력 신장 관점에서 유리할 것으로 생각됨.

2. 주요 요구 내용

<ul style="list-style-type: none"> - 학생 코딩 능력의 중요성 인식 - 컴퓨터공학 세부 분야에 대한 교과목 확대 - 산업체에서 활용 가능한 소프트웨어 응용분야에 대한 교육 확대 - 산업체 인재에 대한 겸직 교수제 도입

3. 컴퓨터공학과 시사점 도출

<ul style="list-style-type: none"> - 컴퓨터공학을 활용하는 여러 응용분야에 대한 과목이 확충 되어야 하며, 이를 수행하기 위한 우수 교수진 추가 확보 등 거고적 차원의 계획이 필요함
--

4. 컴퓨터공학과 교육목표 및 인재상

구분	세부내용		
학과(전공) 교육목표	첨단 IT 기술과 우리민족이 지니고 있는 문화적 저력을 접목시켜 새로운 기술문명의 패러다임을 창출하고자 한다.		
학과(전공) 인재상	학과 인재상	세부내용	본교 인재상과의 연계성
	소프트웨어 분야 핵심 개발 능력을 보유한 인재	융합분야 특화 SW 이론/실무 능력과 제4차 산업혁명의 흐름을 선도할 수 있는 인재 필요	비판적 지식탐구 인재
	실무적 전공교육과 후마니타스 교양교육을 통하여 미래산업을 주도할 SW중심의 T자형 인재	컴퓨터공학 이론을 기반으로 실제 세계에서 발생하는 여러 문제를 추상화하여 구현할 수 있는 능력을 갖춘 인재	주도적 혁신융합 인재
기초과학 및 윤리의식을 함양한 사회에서 요구되는 창의적 능력 배양함	기초과학을 및 윤리의식을 함양하여 문제해결 능력과 인품을 갖춘 사회적 가치추구 인재	사회적 가치추구 인재	

5. 컴퓨터공학과 전공능력

인재상	전공능력	전공능력의 정의
컴퓨터공학 및 수학(數學) 능력을 보유한 인재	소프트웨어 기본역량	자료구조, 데이터베이스, 운영체제 등의 기본적인 컴퓨터공학에 기본적인 지식을 습득하고 이를 활용할 수 있는 능력
실세계 문제를 인공지능 기술로 해결할 수 있는 실천 능력을 보유한 인재	추상화 능력	실세계 문제를 인공지능 문제로 추상화하여 표현할 수 있는 능력
	문제해결 능력	컴퓨터공학 기술을 사용하여 추상화된 실세계 문제를 모델링하여 구현할 수 있는 능력
인간에 대한 이해 및 윤리의식에 바탕을 둔 글로벌 경쟁력을 갖춘 인재	윤리의식 및 커뮤니케이션 능력	컴퓨터공학을 통한 자동화가 사회에 끼칠 수 있는 여러 윤리적인 문제를 이해하고 타인과의 효율적인 커뮤니케이션을 통해 본인의 문제해결을 도모할 수 있는 능력

6. 전공능력 제고를 위한 전공 교육과정 구성 및 체계도

가. 전공 교육과정 구성표

전공능력	학년	이수학기	교과목명
기초과학 및 수학	1	1	선형대수
기초과학 및 수학	1	1	미분적분학
기초과학 및 수학	1	2	미분방정식
기초과학 및 수학	2	1,2	확률및랜덤변수
기초과학 및 수학	2	1,2	이산구조
소프트웨어 기본역량	1	1,2	객체지향프로그래밍
추상화 능력	1	1,2	디자인적사고
소프트웨어 기본역량	1	1,2	웹/파이선프로그래밍
소프트웨어 기본역량	2	1,2	논리회로
소프트웨어 기본역량	2	1,2	컴퓨터구조
소프트웨어 기본역량	2	1,2	자료구조
소프트웨어 기본역량	2	1,2	오픈소스SW개발방법및도구

전공능력	학년	이수학기	교과목명
문제해결 능력	2	1,2	기계학습
소프트웨어 기본역량	3	1,2	운영체제
소프트웨어 기본역량	3	1,2	컴퓨터네트워크
소프트웨어 기본역량	3	1,2	알고리즘
소프트웨어 기본역량	3	1,2	데이터베이스
소프트웨어 기본역량	3	1,2	소프트웨어공학
문제해결 능력	4	1,2	졸업논문
문제해결 능력	4	1,2	졸업프로젝트
문제해결 능력	4	1,2	캡스톤디자인
추상화 능력	2	1	UI/UX프로그래밍
소프트웨어 기본역량	2	1	회로와신호
소프트웨어 기본역량	2	1,2	신호와시스템
소프트웨어 기본역량	2	1,2	마이크로서비스 프로그래밍
소프트웨어 기본역량	3	1	컴파일러
문제해결 능력	3	1	메타버스시스템
윤리의식 및 커뮤니케이션 능력	3	1	SW스타트업비즈니스
소프트웨어 기본역량	3	1	클라우드컴퓨팅
소프트웨어 기본역량	3	2	프로그래밍언어론
소프트웨어 기본역량	3	2	리눅스시스템프로그래밍
문제해결 능력	3	2	SW스타트업프로젝트
소프트웨어 기본역량	3	2	빅데이터프로그래밍
문제해결 능력	3	1,2	딥러닝
문제해결 능력	3	1,2	실전기계학습
소프트웨어 기본역량	3-4	2	로봇프로그래밍
문제해결 능력	4	1	정보보호
문제해결 능력	4	1	영상처리
문제해결 능력	4	1	컴퓨터그래픽스
문제해결 능력	4	1	빅데이터프로젝트
윤리의식 및 커뮤니케이션 능력	4	1	최신기술콜로키움1
문제해결 능력	4	1	AIoT소프트웨어
문제해결 능력	4	1	SI네트워킹
문제해결 능력	4	1	AIoT네트워크
윤리의식 및 커뮤니케이션 능력	2-4	2	최신기술콜로키움2
문제해결 능력	4	2	메타버스데이터처리
문제해결 능력	4	2	인간-컴퓨터상호작용
문제해결 능력	4	2	클라우드프로젝트
문제해결 능력	4	2	AIoT디지털시스템
문제해결 능력	4	2	컴퓨터비전
문제해결 능력	4	2	블록체인
문제해결 능력	4	2	모바일/웹서비스프로젝트

전공능력	학년	이수학기	교과목명
문제해결 능력	4	2	웹보안
문제해결 능력	2-4	1,2	단기현장실습
문제해결 능력	2-4	1,2	장기현장실습
문제해결 능력	2-4	1	연구연수활동1
문제해결 능력	2-4	2	연구연수활동2
문제해결 능력	3	1	독립심화학습1
문제해결 능력	3	2	독립심화학습2
문제해결 능력	2	2	게임프로그래밍입문
문제해결 능력	2	2	개인엔진기초
문제해결 능력	3	1	3D데이터처리
문제해결 능력	3	1	게임인터랙티브테크놀로지
문제해결 능력	3	1	게임그래픽프로그래밍
소프트웨어 기본역량	1	2	인공지능프로그래밍
문제해결 능력	3	1,2	고급딥러닝
문제해결 능력	3-4	1	빅데이터마이닝
문제해결 능력	3-4	1	인공지능과게임프로그래밍
문제해결 능력	3	2	가상/증강현실이론및실습
문제해결 능력	4	1	자연어처리
문제해결 능력	2-4	2	플스택서비스네트워킹
문제해결 능력	3-4	1	플스택서비스프로그래밍
문제해결 능력	4	1	강화학습

나. 권장 이수 체계도

		휴머니티스 교양	전공기초 (15학점)	전공필수 (45학점)	전공선택 (27학점-산학필수12학점 포함)
1학년	1학기	- 교양1 - 교양2 - 교양3	- 미분적분학 - 선형대수	- 웹/파이선프로그래밍	
	2학기	- 교양4 - 교양5	- 미분방정식 - 이산구조	- 디자인적사고 - 객체지향프로그래밍	
2학년	1학기	- 교양6 - 교양7	- 확률및랜덤변수	- 오픈소스SW개발방법및도구 - 논리회로	- 공통선택1
	2학기	- 교양8		- 기계학습 - 자료구조 - 컴퓨터구조	- 공통선택2 - 공통선택3
3학년	1학기	- 교양9		- 운영체제 - 컴퓨터네트워크 - 소프트웨어공학	- 분야선택1
	2학기	- 교양10		- 데이터베이스 - 알고리즘	- 분야선택2 - 분야선택3
4학년	1학기	- 교양11		- 캡스톤디자인	- 분야선택4
	2학기			- 졸업논문 - 졸업프로젝트	