소프트웨어융합학과 교육과정

학과소개

◩ 소프트웨어융합학과는 2017년도 설립되어 미래자동차·로봇트랙(Future Vehicle and Robot Track), 데이터사이언스트랙(Data Science Track), 게임콘텐츠트랙(Game Contents Track) 및 융합리더트랙(Convergence Leader Track)을 개설하며, 향후 사회와 학생의 수요에 맞춘 유연한 트랙 운영을 목표로 한다.

◩ 소프트웨어융합학과는 소프트웨어의 급격한 발전과 인공지능(AI: Artificial Intelligence)의 보편화로 신세계가 만들어지는 제4차 산업혁명이 시작된 세계적 추이를 반영하여 개설한다. 제4차 산업의 거대한 세계적 흐름을 선도하기 위하여, 신산업과 신학문을 선도하고, 미래 사회를 선도적으로 이끌어 나가는 인재 양성에 교육의 목표를 두고 있다. 소프트웨어융합학과는 소프트웨어를 기반으로 혁신적으로 발전할 융합 분야를 선정하고, 분야별로 전문화된 융합 전공 지식과 특화된 소프트웨어개발 능력을 교육함으로써, 목표한 융합 분야에서 바로 창업하거나 실무 투입이 가능한 글로벌 리더급 인재를 양성 한다.

◩ 소프트웨어융합학과(Department of Software Convergence)는 학부 졸업자에 공학사(Bachelor of Engineering)을 부여하며, 다음과 같이 학위명을 표기한다.
- 소프트웨어융합학과 (게임콘텐츠트랙), Bachelor of Software Convergence (Game Contents Track)
- 소프트웨어융합학과 (데이터사이언스트랙), Bachelor of Software Convergence (Data Science Track)
- 소프트웨어융합학과 (미래자동차·로봇트랙), Bachelor of Software Convergence (Future Vehicle and Robot Track)
- 소프트웨어융합학과 (융합리더트랙), Bachelor of Software Convergence (Convergence Leader Track)

◩ 소프트웨어융합학과는 다전공 졸업자에 학부 졸업자와 동일한 학위명(학과 및 트랙)을 표기한다

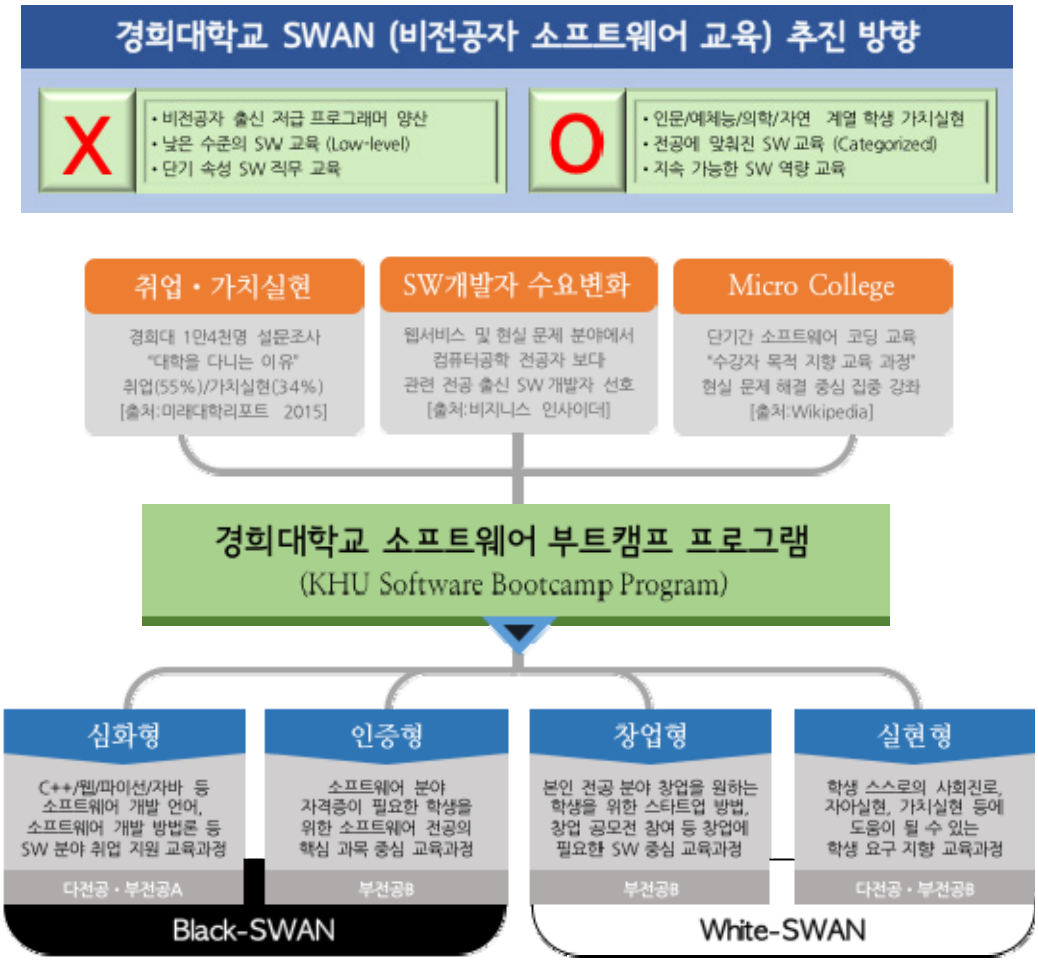
1. 교육목적

융합목표가 되는 분야에 대한 압축된 전공 지식과 해당 융합목표 분야에 특화된 소프트웨어 전문 지식 및 개발 능력을 토대로 신학문/신산업을 이끌어 나갈 인재 양성에 그 목적이 있다.

2. 교육목표

1) 소프트웨어융합학과는 융합이 기본이 되는 4차 산업분야를 선도할 수 있는 실천적 인재 양성을 목표로 하여, 졸업과 동시에 목표 산업 분야에서 바로 실무와 창업이 가능한 인력을 양성하는 것을 교육 목표로 함. 이를 위하여, 1) 수학적 지식과 2) 융합목표전공(Non-software, Domain Specific Knowledge)에 대한 전문 지식을 기반으로 하여, 3) 융합 분야에 특화된 소프트웨어의 이론적 지식과 실무적 능력을 확보한 전문 인력 양성을 목표로 함. 이를 위하여 트랙별로 특화된 인재상을 재정립하여 교육과정을 설계하고 운영하도록 함
2) [게임콘텐츠트랙] 게임에 대한 역사·철학·이론적 배경을 근거로, 시대가 바뀌어도, 문화·기술을 융합한 창의적 게임 소프트웨어(AI/VR/AR(Artificial Intelligence/Virtual Reality/Augmented Reality) Game, Serious Game, O2O(Online to Offline) Game 등)를 만들어 낼 수 있는 인력 양성
3) [데이터사이언스트랙] 수학과 공학을 배경으로, 데이터의 생성부터 폐기의 전 과정에 대한 통합 실무를 수행할 수 있으며, Data Life-Cycle의 전 과정(기획~폐기)에 대한 실전적 능력을 배양하는 교육
4) [미래자동차·로봇트랙] 자동차와 로봇에 대한 하드웨어와 소프트웨어를 통합적으로 설계/개발할 수 있으며, 새로운 혁신적 미래자동차와 로봇을 제안할 수 있는 교육
5) [융합리더트랙] 학생 스스로 미래를 설계하고 새로운 융합분야를 창조해 낼 수 있도록 하는 교육

6) 소프트웨어융합학과의 다전공/부전공 교육과정은 비전공자에 대한 가치실현, 전공에 부합하는 소프트웨어 교육, 그리고 지속 가능한 소프트웨어 역량 교육을 목표로 운영함 ([그림1] 참조)



[그림 1] 소프트웨어융합학과 다전공/부전공 교육 유형 및 목표

3. 학과별 교과목 수

학과명	구분	전공기초	전공필수	전공선택	전공과목
소프트웨어 융합학과	과목수	5	14	62	81
	학점수	15	37	174	226

※ 소프트웨어융합학과 교육과정 시행세칙에 명기된, 트랙 참여 타학과의 전공과목 포함임
※ 현장실습/장기현장실습 과목은 제외한 현황임

4. 소프트웨어융합학과 졸업 요건

1) 교육과정 기본구조표

학과	졸업 이수 학점	단일전공과정					다전공과정					부전공(심화형) 과정			부전공(인증형/창업형/실현형) 과정		
		전공학점				타 전공 인정 학점	전공학점				타 전공 인정 학점						
		전공 기초	전공 필수	전공 선택	계		전공 기초	전공 필수	전공 선택	계		전공 필수	전공 선택	계	전공 필수	전공 선택	계
소프트웨어융합학과	130	15	37	36	88	-	15	21	24	60	-	18	12	30	12	9	21

※ 교양이수는 교양교육과정 이수구조를 따름
※ 트랙별 세부적인 전공이수 및 타전공 인정학점 등은 소프트웨어융합학과 교육과정 시행세칙에 따름

2) 졸업논문

소프트웨어융합학과의 '소프트웨어융합캡스톤디자인'을 이수하는 것으로 경희대학교 졸업을 위한 "졸업논문" 합격으로 인정한다. 단, '졸업논문(소프트웨어융합)'을 필히 수강신청 해야 한다.

3) 졸업능력인증제

소프트웨어융합대학 졸업능력 인증제를 따른다.

소프트웨어융합학과 교육과정 시행세칙

제 1 장 총 칙

제1조(학과설치목적) ① 소프트웨어융합학과는 소프트웨어의 급격한 발전과 인공지능(AI)의 보편화로 신세계가 만들어지는 제4차 산업혁명이 시작된 현재의 세계적인 추이를 반영하여 개설한다. 제4차 산업의 거대한 세계적 흐름을 선도하기 위하여, 신산업과 신학문을 선도하며, 미래 사회를 선도적으로 이끌어 나가는 인재 양성에 교육의 목표를 두고 있다. 소프트웨어융합학과는 소프트웨어를 기반으로 혁신적으로 발전할 융합 분야를 선정하고, 분야별로 전문화된 융합 전공 지식과 특화된 소프트웨어개발 능력을 교육함으로써, 목표한 융합 분야에서 바로 창업하거나 실무 투입이 가능한 글로벌 리더 인재를 양성 한다.

② 소프트웨어융합학과에는 미래자동차·로봇트랙, 데이터사이언스트랙, 게임콘텐츠트랙 및 융합리더트랙을 개설하여, 향후 사회와 학생의 수요에 맞춘 유연한 트랙 운영을 목표로 한다.

③ 게임콘텐츠트랙은 게임에 대한 역사·철학·이론적 배경을 근거로, 시대가 바뀌어도, 문화·기술을 융합한 (기존에 없던) 창의적 게임 소프트웨어(AI/VR/AR Game, Serious Game, O2O Game 등)를 만들어 낼 수 있는 인력 양성을 하는 교육이다. 데이터사이언스트랙은 수학과 공학을 배경으로, 데이터의 생성부터 폐기의 전 과정에 대한 통합 실무를 수행할 수 있으며, Data Life-Cycle의 전 과정(기획~폐기)에 대한 실전적 능력을 배양하는 교육이다. 미래자동차·로봇트랙은 국내 최초로 자동차와 로봇에 대한 하드웨어와 소프트웨어를 통합적으로 설계/개발할 수 있으며, 새로운 혁신적 미래자동차와 로봇을 제안할 수 있는 교육이다. 융합리더트랙은 학생 스스로 새로운 융합분야를 개척할 수 있도록 하는 교육이다.

제2조(일반원칙) ① 소프트웨어융합학과를 단일전공, 다전공, 부전공으로 이수하고자 하는 학생은 이 시행세칙에서 정하는 바에 따라 교과목을 이수해야 한다.

② 교과목의 선택은 지도교수와 상의하여 결정한다.

③ 모든 교과목은 [별표1] 교육과정 편성표에 제시된 이수학년과 개설학기에 준해 이수할 것을 권장한다.

④ 전공과목의 선수과목지정은 [별표5]와 같으며, 선·후수과목의 체계를 준수하여 이수하여야 한다. 선·후수과목의 체계는 전산시스템에 반영되어 있으며 수강신청 시 자동으로 적용된다. 단, 교과목 담당교수가 인터뷰를 통하여 선수과목의 필요여부를 판단하여 선수과목 미이수 학생에 대하여 수강을 허용할 수 있다.

제 2 장 교양과정

제3조(교양과목 이수) ① 교양과목은 본 대학교 교양과정 기본구조표에서 정한 소정의 학점을 취득하여야 한다.

제 3 장 전공과정

제4조(전공 및 트랙과목 이수) ① 소프트웨어융합학과 단일전공과정 졸업을 위해서는 [표1]에 명시된 전공기초, 전공필수(융합필수, 소프트웨어필수, 산학필수 포함), 전공선택 학점을 이수하여야 한다.

② 게임콘텐츠 트랙을 이수하고자 하는 학생은 [표1]의 전공선택 과목에서, 게임콘텐츠 트랙을 위하여 개설된 '트랙필수' 및 '트랙선택' 교과목을 최소 27학점(9과목) 이수하여야 한다. 이중 게임콘텐츠 트랙을 위하여 개설된 '트랙필수'는 반드시 모두 이수하여야 한다.

③ 데이터사이언스 트랙을 이수하고자 하는 학생은 [표1]의 전공선택 과목에서, 데이터사이언스 트랙을 위하여 개설된 '트랙필수' 및 '트랙선택' 교과목을 최소 27학점(9과목) 이수하여야 한다. 이중 데이터사이언스 트랙을 위하여 개설된 '트랙필수'는 반드시 모두 이수하여야 한다.

④ 미래자동차·로봇 트랙을 이수하고자 하는 학생은 [표1]의 전공선택 과목에서, 미래자동차·로봇 트랙을 위하여 개설된 '트랙필수' 및 '트랙선택' 교과목을 최소 27학점(9과목) 이수하여야 한다. 이중 미래자동차·로봇 트랙을 위하여 개설된 '트랙필수'는 반드시 모두 이수하여야 한다.

⑤ 융합리더 트랙을 이수하고자 하는 학생은 [표1]의 전공선택 과목에서, 최소 36학점(12과목) 이수하여야 한다.

⑥ 학년별/학기별 교과목 편성은 [별표1]을 참조한다.

⑦ 소프트웨어융합학과를 다전공과정으로 이수하고자 하는 학생은 [표2]에 명시된 전공기초, 전공필수, 전공선택 학점을 이수하여야 하며, [표3]에 명시된 타학과 과목들을 소프트웨어융합학과 전공과목으로 대체 인정 받을 수 있다. 단일전공과정을 이수하는 학생과 동일하게 개설된 트랙 중 하나를 선택하여, 해당 트랙의 트랙 필수와 트랙 선택 과목을 이수하여야 한다.

⑧ 소프트웨어융합학과를 부전공과정으로 이수하고자 하는 학생은 [표4]에 명시된 전공필수, 전공선택 학점을 이수하여야 한다. 부전공과정을 이수하고자 하는 학생은 본인이 희망하는 유형을 먼저 정의하고, 희망하는 유형에 맞춰서 [표4]의 유형별 전공과목을 이수하여야 한다.

[표1] 단일전공 전공과목 편성표

구분	교과목명			과목수
전공기초 (15)	수학		미분적분학, 선형대수, 미분방정식, 확률 및 랜덤변수(EE)	5
	물리		물리학 및 실험 1	
전공필수 (37)	융합필수		소프트웨어융합개론, 디자인적사고, 최신기술콜로키움 1(CSE), 최신기술콜로키움 2, 소프트웨어융합캡스톤디자인, 졸업논문(소프트웨어융합)	14
	소프트웨어필수		객체지향프로그래밍(CSE), 웹/파이선프로그래밍, 소프트웨어개발방법 및 도구, 자료구조(CSE), 알고리즘분석(CSE), 운영체제(CSE), 데이터베이스(CSE), 소프트웨어공학(CSE)	
전공선택 (36)	데이터사이언스 트랙	트랙 필수	응용통계학 (통계학2)(IE), 데이터센터프로그래밍, 고객관계관리(IE), 데이터마이닝(IE), 데이터분석캡스톤디자인	14
		트랙 선택	경영과학1(기본)(IE), 경영과학 2(심화)(IE), 의사결정론(IE), 고급데이터마이닝, 금융공학(IE), 프로세스마이닝, 서비스데이터사이언스(IE), 금융데이터분석, 데이터사이언스 및 시각화	
	게임콘텐츠 트랙	트랙 필수	스토리텔링(DC), 게임 프로그래밍 입문, 게임 UI/UX 디자인, 게임학(DC), 게임그래픽 프로그래밍, 게임엔진프로그래밍, 게임디자인(DC)	14
		트랙 선택	3D 모델링(DC), 인터랙션 디자인(DC), 사운드 디자인(DC), 3D 애니메이션(DC), 체감형기술 이론 및 실습, 가상/증강 현실 이론 및 실습, 모션그래픽스(DC)	
	미래자동차·로봇 트랙	트랙 필수	물리학 및 실험 2(APHY), 동역학(ME), 전기전자회로(ME), 신호와시스템(EE), 미래자동차·로봇프로그래밍, 메카트로닉스(ME)	17
		트랙 선택	계측공학(ME), 자동제어(ME), 혁신적미래자동차로봇디자인, 시스템동역학(ME), 미래자동차·로봇캡스톤디자인, 로봇제어공학(EE), 로봇공학(ME), IoT소프트웨어(CSE), 영상처리(CSE), 디지털신호처리(EE), 디지털통신1(EE)	
	융합리더 트랙	트랙 필수	※ 전공 선택에서 자유롭게 수강함 (최소 12과목 - 36학점 이상)	0
		트랙 선택		
	공통 선택		이산구조(CSE), 시스템분석 및 설계(CSE), 컴퓨터네트워크(CSE), 컴퓨터그래픽스(CSE), 소프트웨어스타트업비지니스(CSE), Java프로그래밍(CSE), 인공지능(CSE), 정보보호(CSE) 독립심화학습 1/2, 융합연구 1/2/3/4/5/6, 현장실습/장기현장실습, 머신러닝(CSE)	18
	산학필수(10)		최신기술콜로키움 1(CSE), 최신기술콜로키움 2, 소프트웨어융합캡스톤디자인, 각 트랙별 캡스톤디자인 과목 (데이터분석캡스톤디자인, 게임디자인(DC), 미래자동차·로봇캡스톤디자인 중 택 1), 현장실습/장기현장실습	8

※ 졸업논문(0학점, P/N평가), 최신기술콜로키움 1/최신기술콜로키움 2(2학점, P/N평가), 독립심화학습 1/2(3학점, P/N평가), 융합연구 1/2/3/4/5/6(1학점, P/N평가), 사운드디자인(2학점, 등급평가)이며, 상기 과목을 제외한 모든 과목은 3학점, 등급평가임

※ 과목명에 학수번호 코드가 있는 과목은 소프트웨어융합학과의 트랙에 참여하는 타학과 과목임
(CSE: 컴퓨터공학과, EE: 전자공학과, DC: 디지털콘텐츠학과, IE: 산업경영공학과, ME: 기계공학과)

※ 산학필수에 대한 사항은 본 교육과정의 부칙 제3조(산학필수)에 따름

[표2] 다전공 전공과목 편성표

구분	교과목명			과목수
전공 기초 (15)	수학		미분적분학, 선형대수, 미분방정식, 확률 및 랜덤변수(EE)	5
	물리		물리학 및 실험 1	
전공 필수 (21)	융합필수		디자인적 사고, 소프트웨어융합캡스톤디자인, 졸업논문(소프트웨어융합)	8
	소프트웨어필수		객체지향프로그래밍(CSE), 웹/파이선 프로그래밍, 소프트웨어개발방법 및 도구, 자료구조(CSE), 알고리즘분석(CSE)	
전공 선택 (24)	데이터사이언스 트랙	트랙 필수	응용통계학 (통계학2)(IE), 데이터센터 프로그래밍, 고객관계관리(IE), 데이터마이닝(IE), 데이터분석캡스톤디자인	14
		트랙 선택	경영과학 1 (기본)(IE), 경영과학 2 (심화)(IE), 의사결정론(IE), 고급데이터마이닝, 금융공학(IE), 프로세스마이닝, 서비스데이터사이언스(IE), 금융데이터분석, 데이터사이언스 및 시각화	
	게임콘텐츠 트랙	트랙 필수	스토리텔링(DC), 게임 프로그래밍 입문, 게임 UI/UX 디자인, 게임학(DC), 게임그래픽 프로그래밍, 게임 엔진 프로그래밍, 게임디자인(DC)	14
		트랙 선택	3D 모델링(DC), 인터랙션 디자인(DC), 사운드 디자인(DC), 3D 애니메이션(DC), 체감형기술이론 및 실습, 가상/증강 현실이론 및 실습, 모션그래픽스(DC)	
	미래자동차·로봇 트랙	트랙 필수	물리학 및 실험 2(APHY), 동역학(ME), 전기전자회로(ME), 신호와시스템(EE), 미래자동차·로봇프로그래밍, 메카트로닉스(ME)	17
		트랙 선택	계측공학(ME), 자동제어(ME), 혁신적 미래자동차로봇 디자인, 시스템동역학(ME), 미래자동차·로봇 캡스톤디자인, 로봇제어공학(EE), 로봇공학(ME), IoT소프트웨어(CSE), 영상처리(CSE), 디지털신호처리(EE), 디지털통신1(EE)	
	융합리더 트랙	트랙 필수	※ 전공 선택에서 자유롭게 수강함 (최소 8과목 - 24학점 이상)	0
		트랙 선택		

※ 졸업논문(0학점, P/N평가), 최신기술콜로키움 1/최신기술콜로키움 2(2학점, P/N평가), 독립심화학습 1/2(3학점, P/N평가), 융합연구 1/2/3/4/5/6(1학점, P/N평가), 사운드디자인(2학점, 등급평가)이며, 상기 과목을 제외한 모든 과목은 3학점, 등급평가임
※ 과목명에 학수번호 코드가 있는 과목은 소프트웨어융합학과의 트랙에 참여하는 타학과 과목임
(CSE: 컴퓨터공학과, EE: 전자공학과, DC: 디지털콘텐츠학과, IE: 산업경영공학과, ME: 기계공학과)

[표3] 대체인정과목 일람표

순번	대체인정 교과과정				타학과 교과과정			비고
	이수구분	학수번호	교과목명	학점	학수번호	교과목명	학점	
1	전공기초	AMTH1009	미분적분학	3	AMTH1002	미분적분학 1	3	
					MATH1101	미적분학 및 연습 1	3	서울캠퍼스
2	전공기초	AMTH1001	미분방정식	3	IE203	공학수학 1	3	
					ME113	공학수학 1	3	
3	전공기초	AMTH1004	선형대수	3	IE204	공학수학 2	3	
					ME202	공학수학 2	3	
4	전공기초	APHY1002	물리학 및 실험 1	3	APHY1000	물리학 1	3	
					APHY1004	일반물리	3	
					PHYS1101	물리학 및 실험 1	3	서울캠퍼스
5	전공선택	APHY1003	물리학 및 실험 2	3	APHY1001	물리학 2	3	
					PHYS1102	물리학 및 실험 2	3	서울캠퍼스
6	전공선택	CSE440	IoT소프트웨어	3	CSE425	임베디드소프트웨어	3	
7	전공선택	ME380	시스템동역학	3	ME373	시스템 모델링	3	

1) 우측 타학과 과목 수강시에 좌측 소프트웨어융합학과 과목으로 인정함
2) 소프트웨어융합학과를 다전공 과정으로 이수하고자 하는 학생 대상
3) 소프트웨어융합학과를 단일전공 과정으로 이수하고자 하는 전과생의 경우, 1번 항목을 제외하고 인정함

[표4] 부전공 전공과목 편성표

부전공 유형	심화형	인증형	창업형	실현형
	부전공(심화형)	부전공(인증형/창업형/실현형)		
전공 필수	객체지향프로그래밍(CSE), 웹/파이선프로그래밍, 디자인적사고, 소프트웨어개발방법 및 도구, 자료구조(CSE), 알고리즘분석(CSE)	객체지향프로그래밍(CSE), 자료구조(CSE), 알고리즘분석(CSE), 운영체제(CSE)	객체지향프로그래밍(CSE), 웹/파이선프로그래밍, 디자인적사고, 소프트웨어개발방법 및 도구	객체지향프로그래밍(CSE), 웹/파이선프로그래밍, 디자인적사고, 소프트웨어개발방법 및 도구
	6과목 (18학점 이수)	4과목 (12학점 이수)	4과목 (12학점 이수)	4과목 (12학점 이수)
전공 선택	※ [표2]의 전공선택 과목 중 12학점 이상 이수	이산구조(CSE), 데이터베이스(CSE), 소프트웨어공학(CSE), 컴퓨터네트워크(CSE), Java프로그래밍(CSE)	데이터센터프로그래밍, SW스타트업비지니스(CSE), 융합연구 1,2,3,4,5,6 (융합연구에서는 3학점만 가능)	※ [표1]의 전공선택 과목 중 9학점 이상 이수
	- (12학점 이수)	5과목 (9학점 이수)	8과목 (9학점 이수)	- (9학점 이수)

※ 졸업논문(0학점, P/N평가), 최신기술콜로키움 1/최신기술콜로키움 2(2학점, P/N평가), 독립심화학습 1/2(3학점, P/N평가), 융합연구 1/2/3/4/5/6(1학점, P/N평가), 사운드디자인(2학점, 등급평가)이며, 상기 과목을 제외한 모든 과목은 3학점, 등급평가임
※ 과목명에 학수번호 코드가 있는 과목은 소프트웨어융합학과의 트랙에 참여하는 타학과 과목임
(CSE: 컴퓨터공학과, EE: 전자공학과, DC: 디지털콘텐츠학과, IE: 산업경영공학과, ME: 기계공학과)

제5조(타전공과목 이수) 타전공과목의 이수는 [표1] 단일전공 전공과목 편성표와 [별표1] 교육과정 편성표에 명시된 과목에 대해서 전공 학점으로 인정한다.

제6조(대학원 과목의 이수요건과 인정과목) ① 3학년까지의 평균 평점이 3.5 이상인 학생은 대학원 소프트웨어융합학과 주임교수의 승인을 받아 학과생의 이수가 허용된 대학원 교과목을 통산 6학점까지 이수할 수 있으며, 그 취득 학점은 전공선택 학점으로 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 절차 거쳐 6학점이내에서 대학원 진학 시에 대학원 학점으로 인정받을 수 있다.

제 4 장 졸업이수요건

제7조(졸업이수학점) 소프트웨어융합학과의 최저 졸업이수학점은 130학점이다.

제8조(전공 및 트랙이수학점) ① 단일전공과정 : 소프트웨어융합학과의 학생으로서 단일전공자는 전공기초 15학점, 전공필수(융합필수, 소프트웨어필수, 산학필수 포함) 37학점, 전공선택 36학점을 포함하여 전공학점 88학점 이상을 이수하여야 하며, 소속한 트랙에 따라 제4조의 ②항, ③항, ④항, ⑤항 중 본인의 트랙에 해당하는 사항을 반드시 이수하여야 한다.

② 다전공과정 : 소프트웨어융합학과 학생으로서 타전공을 다전공으로 이수하거나 타학과 학생으로서 소프트웨어융합전공을 다전공과정으로 이수하는 학생은 전공기초 15학점, 전공필수 21학점, 전공선택 24학점을 포함하여 전공학점 60학점 이상 이수하여야 한다.

③ 부전공과정 : 타학과 학생의 소프트웨어융합전공 부전공과정 이수는 제4조의 ⑧항을 따른다.

제9조(편입생 전공이수학점) ① 일반편입생은 전적대학에서 이수한 학점 중 본교 학점인정심사에서 인정받은 학점을 제외한 나머지 학점을 추가로 이수하여야 한다.

② 학사편입생은 본교 학점인정심사에 의거 전공기초과목은 인정할 수 있으나, 전공필수 및 전공선택 학점은 인정하지 않는다.

제10조(영어강좌 이수학점) 전공과목 중에서 영어강좌를 3과목 이상, 편입생의 경우에는 1과목 이수하여 졸업요건을 충족하여야 하며, 졸업능력인증제를 따른다.

제11조(졸업논문) 소프트웨어융합학과의 '소프트웨어융합캡스톤디자인(SWCON401)' 교과목을 이수하는 것으로 경희대학교 졸업을 위한 "졸업논문" 합격으로 인정한다. 단, '졸업논문(소프트웨어융합)'을 필히 수강신청 해야 한다.

제12조(SW교육 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

제13조(졸업논문 제출의무의 대상) 소프트웨어융합학과를 졸업하기 위해서는, 제11조에 의거하여 '소프트웨어융합캡스톤디자인(SWCON401)' 교과목을 이수한 후, 지도교수가 지정하는 시기에 졸업논문을 제출하여야 한다 (다전공자 포함). 단, 소프트웨어융합을 부전공하는 경우에는 제출의무를 적용하지 아니한다.

제14조(졸업논문 지도교수의 지정) 7학기 이수 중에 전공지도교수에게 "졸업논문계획서"를 제출하고 졸업논문 지도교수를 지정받아야 한다.

제15조(졸업논문 제출자격의 부여) 졸업논문 제출자격은 6학기를 이수한 후 소프트웨어융합학과에서 개최하는 "졸업논문발표회"에서 1회 이상 발표한 자에게만 부여한다.

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

제17조(최초 트랙의 신청) ① 소프트웨어융합학과를 단일전공과정으로 이수하는 학생은 1학년 2학기에, 본인이 참여를 희망하는 트랙을 신청하여야 한다. 신청은 소프트웨어융합학과에서 지정하는 소정의 기간에 신청한다. 신청한 사항은 소프트웨어융합학과 교수회의를 통하여 심사 후 트랙 배정을 실시하는 것으로 한다. 융합리더트랙을 신청하고자 하는 학생은 별도의 "융합리더 학업계획서"를 작성하여 제출한다. 소프트웨어융합학과 교수회의를 통하여 융합리더트랙 신청 학생을 심의 후 결정한다. 융합리더트랙 신청학생에 대한 학업계획은 교수회의의 권고를 통하여 조정가능하다. 융합리더트랙의 인원은 매년 교수회의를 통하여 결정하도록 한다.

② 소프트웨어융합학과를 다전공 과정으로 이수하고자 하는 학생은 다전공 신청기간에 온라인 신청과 별개로 오프라인으로 트랙 신청에 대한 양식에 맞춰 트랙 신청 이유를 제시하며, 이를 소프트웨어융합학과 교수회의에서 심의하여 트랙 참여를 결정한다.

③ 소프트웨어융합학과를 부전공과정으로 이수하고자 하는 학생은 본인이 희망하는 유형(심화형의 경우는 트랙 포함)을 선택하여 교육과정을 이수하고, 졸업예정학기에 행정실에 부전공을 신청한다.

④ 소프트웨어융합학과를 전과 혹은 편입으로 이수하고자 하는 학생은 전과/편입 시점에 트랙 신청에 대한 양식에 맞춰서 트랙 신청 이유를 제시하며, 이를 소프트웨어융합학과 교수회의에서 심의하여 트랙 참여를 결정한다.

제18조(트랙의 변경) 소프트웨어융합학과 학생의 트랙 변경은 원칙적으로 불가능 하다. 불가피하게 트랙을 변경하고자 하는 경우는, 트랙 변경에 대한 사유를 지도교수에게 제출하고, 이에 대한 소프트웨어융합학과 교수회의를 통하여 결정한다.

제19조(복수 트랙의 이수) 소프트웨어융합학과 학생이 본인이 최초 선택한 트랙 외에 소프트웨어융합학과가 개설한 다른 트랙의 이수 조건을 만족한 경우는 복수 트랙에 대한 졸업 자격을 부여한다. 본인이 최초 선택한 트랙외의 추가 트랙에 대한 이수 여부는 제4조의 ②항, ③항, ④항, ⑤항에 따른다.

제20조(보칙) 본 내규에 정하지 않는 사항은 소프트웨어융합학과회의 의결에 따른다.

부 칙

제1조(시행일) 본 시행세칙은 2019년 3월 1일부터 시행한다.

[별표1] 교육과정 편성표
[별표2] 게임콘텐츠트랙 교육과정 이수체계도
[별표3] 데이터사이언스트랙 교육과정 이수체계도
[별표4] 미래자동차·로봇트랙 교육과정 이수체계도
[별표5] 선수과정 지정표
[별표6] 2017학년도 교육과정 경과조치
[별표7] 2018학년도 교육과정 경과조치
[별표8] 교과목 해설

제2조(경과조치) ① 2017년 입학생 중 미래자동차·로봇 트랙을 선택한 경우는 미분적분학 1(AMTH1002)과 미분적분학 2(AMTH1003)를 수강하는 것을 원칙으로 한다. 2017년 입학생 중 여타 트랙을 선택한 경우는 미분적분학 1(AMTH1002)을 수강하는 것을 원칙으로 한다.

② 소프트웨어융합학과 단일전공과정을 이수하고자 하는 전과생이 미분적분학1(AMTH1002)를 수강한 경우, 미분적분학 2(AMTH1003) 또는 미분적분학(AMTH1009)를 수강해야 한다. 이 경우, 미분적분학1(AMTH1002)은 전공기초로 인정하고, 미분적분학2(AMTH1003) 또는 미분적분학(AMTH1009) 중 1개 과목을 전공선택으로 인정한다. 소프트웨어융합학과 단일전공과정을 이수하고자 하는 전과생은 [표3]에 명시된 타학과 과목들([표3]의 1번 항목은 제외)을 소프트웨어융합학과 전공과목으로 대체인정 받을 수 있다. 타학과 교과목 대체인정 시, 중복 인정하지 않는다. 본 내용은 2019학년도 교육과정부터 포함되며, 2018학년도 전과생부터 경과조치한다.

③ 소프트웨어융합학과 다전공과정을 이수하고자 하는 학생은 [표3]에 명시된 타학과 과목들을 소프트웨어융합학과 전공과목으로 대체인정 받을 수 있다. 다전공자의 타학과 교과목 대체인정 시, 중복 인정 하지 않는다. 본 내용은 2019학년도 교육과정부터 포함되며, 2018학년도 다전공자부터 경과조치한다.

④ 2017년 입학생(단일전공, 다전공, 편입생, 전과생 모두 포함)은 [별표5]에 명시된 2018학년도 선수과목 지정표를 따른다.

⑤ 2017년 입학생(단일전공, 다전공, 편입생, 전과생 모두 포함)은 [별표6]에 명시된 신규개설 및 대체된 교육과정을 따른다.

⑥ 2018년 입학생(단일전공, 다전공, 편입생, 전과생 모두 포함)은 [별표7]에 명시된 신규개설 및 대체된 교육과정을 따른다.

제3조(산학필수) 산학필수는 최신기술 콜로키움 1(CSE438, 2학점), 최신기술 콜로키움 2(SWCON302, 2학점), 소프트웨어융합 캡스톤디자인(SWCON401, 3학점)과 각 트랙별 캡스톤디자인 과목 3학점(데이터분석 캡스톤디자인(SWCON321), 게임디자인(DC318), 미래자동차·로봇 캡스톤디자인(SWCON332) 중 1과목), 현장실습/장기현장실습을 포함하여 총 10학점 이상을 이수하여야 한다.

[별표1]

소프트웨어융합학과 교육과정 편성표

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		PF 평가	비고
					이론	실기	실습	설계		1학기	2학기		
1	전공 기초	미분적분학	AMTH1009	3	3				1	O			
2		물리학 및 실험 1	APHY1002	3	2		2		1	O			
3		선형대수	AMTH1004	3	3				1	O			
4		미분방정식	AMTH1001	3	3				1		O		
5		확률 및 랜덤변수	EE211	3	3				2/3	O	O		
6	전공 필수	소프트웨어융합개론	SWCON101	3	3				1	O	◎		
7		객체지향프로그래밍	CSE207	3	2		2		2	◎	O		
8		디자인적사고	SWCON103	3	2		2		1	O	◎		
9		웹/파이선프로그래밍	SWCON104	3	2		2		1	◎	O		
10		소프트웨어개발방법 및 도구	SWCON201	3	3				2	◎	O		
11		자료구조	CSE204	3	2		2		2	O	◎		
12		알고리즘분석	CSE304	3	3				3	O	O		
13		운영체제#1	CSE301	3	3				3	O	O		
14		데이터베이스#2	CSE305	3	1		2	1	3	O	O		
15		소프트웨어공학#3	CSE327	3	3				3		O		
16		최신기술콜로키움 1	CSE438	2	2				2-4	O		P/F	
17		최신기술콜로키움 2	SWCON302	2	2				2-4		O	P/F	
18		소프트웨어융합 캡스톤디자인	SWCON401	3	1			3	4	O	O		포
19		졸업논문 (소프트웨어융합)	SWCON402	0					4	O	O	P/F	포
20	전공 선택 (데이터 사이언스 트랙)	응용통계학(통계학 2)	IE208	3	2			1	2		O		
21		데이터센터프로그래밍	SWCON221	3	2		2		2	O	◎		
22		데이터마이닝	IE306	3	3				3	O			
23		데이터분석 캡스톤디자인	SWCON321	3	1			2	3	O	O		포
24		고객관계관리	IE202	3	3				2		O		
25		경영과학 1(기본)	IE301	3	2			1	3	O			
26		경영과학 2(심화)	IE302	3	3				3		O		
27		의사결정론	IE316	3	3				3	O			
28		고급데이터마이닝	SWCON322	3	3				3		O		
29		금융공학	IE414	3	3				4	O			
30		프로세스마이닝	SWCON423	3	3				4	O			
31		서비스데이터사이언스	IE419	3	3				4	O			
32		금융데이터분석	SWCON424	3	2		2		4		O		
33		데이터사이언스 및 시각화	SWCON425	3	2		2		4		O		

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

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		PF 평가	비고
					이론	실기	실습	설계		1학기	2학기		
34	전공 선택 (게임 콘텐츠 트랙)	스토리텔링	DC205	3	2	2			2	O			
35		게임프로그래밍입문	SWCON211	3	2		2		2	O	◎		포
36		게임UI/UX디자인	SWCON212	3	2		2		2		O		
37		게임학	DC304	3	2	2			3	O			
38		게임그래픽프로그래밍	SWCON311	3	2		2		3	O	O		포
39		게임엔진프로그래밍	SWCON314	3	2		2		3	O	O		
40		게임디자인	DC318	3		2		2	3		O		포
41		3D 모델링	DC201	3	2	2			2	O			
42		인터랙션 디자인	DC203	3	2	2			2		O		
43		사운드 디자인	DC207	2		4			2	O			
44		3D 애니메이션	DC211	3	2	2			2		O		
45		체감형기술이론 및 실습	SWCON312	3	2		2		3	O	O		
46		가상/증강현실이론 및 실습	SWCON313	3	2		2		3		O		
47		모션그래픽스	DC302	3	2	2			3	O			
48	전공 선택 (미래 자동차·로봇 트랙)	물리학 및 실험 2	APHY1003	3	2		2		1		O		
49		동역학	ME271	3	3				2	O			
50		전기전자회로	ME275	3	3				2/3	O	O		
51		신호와시스템	EE210	3	3				2/3	O	O		
52		미래자동차·로봇 프로그래밍	SWCON331	3	2		2		3	O	O		
53		메카트로닉스	ME375	3	2			1	3	O			
54		계측공학	ME276	3	3				2		O		
55		혁신적미래자동차·로봇디자인	SWCON231	3	2		2		2		O		
56		시스템동역학	ME380	3	2			1	3	O			
57		자동제어	ME376	3	2			1	3		O		
58		미래자동차·로봇 캡스톤디자인	SWCON332	3	1			2	3		O		포
59		디지털신호처리	EE342	3	2			1	3	O	O		
60		디지털통신 1	EE341	3	3				3	O	O		
61		로봇공학	ME475	3	2			1	4	O			
62		IoT소프트웨어	CSE440	3	1		2	1	4	O			
63		영상처리	CSE426	3	2			1	4	O			
64		로봇제어공학	EE461	3	3				4		O		
65	전공 선택	이산구조	CSE201	3	3				2	O	O		
66		시스템분석 및 설계#4	CSE303	3	2			1	3	O	O		
67		컴퓨터네트워크#5	CSE302	3	3				3	O	O		
68		컴퓨터그래픽스#6	CSE428	3	2		2		4	O			
69		SW스타트업비지니스#7	CSE330	3	3				3		O		
70		Java프로그래밍#8	CSE329	3	1		2	1	3	O			

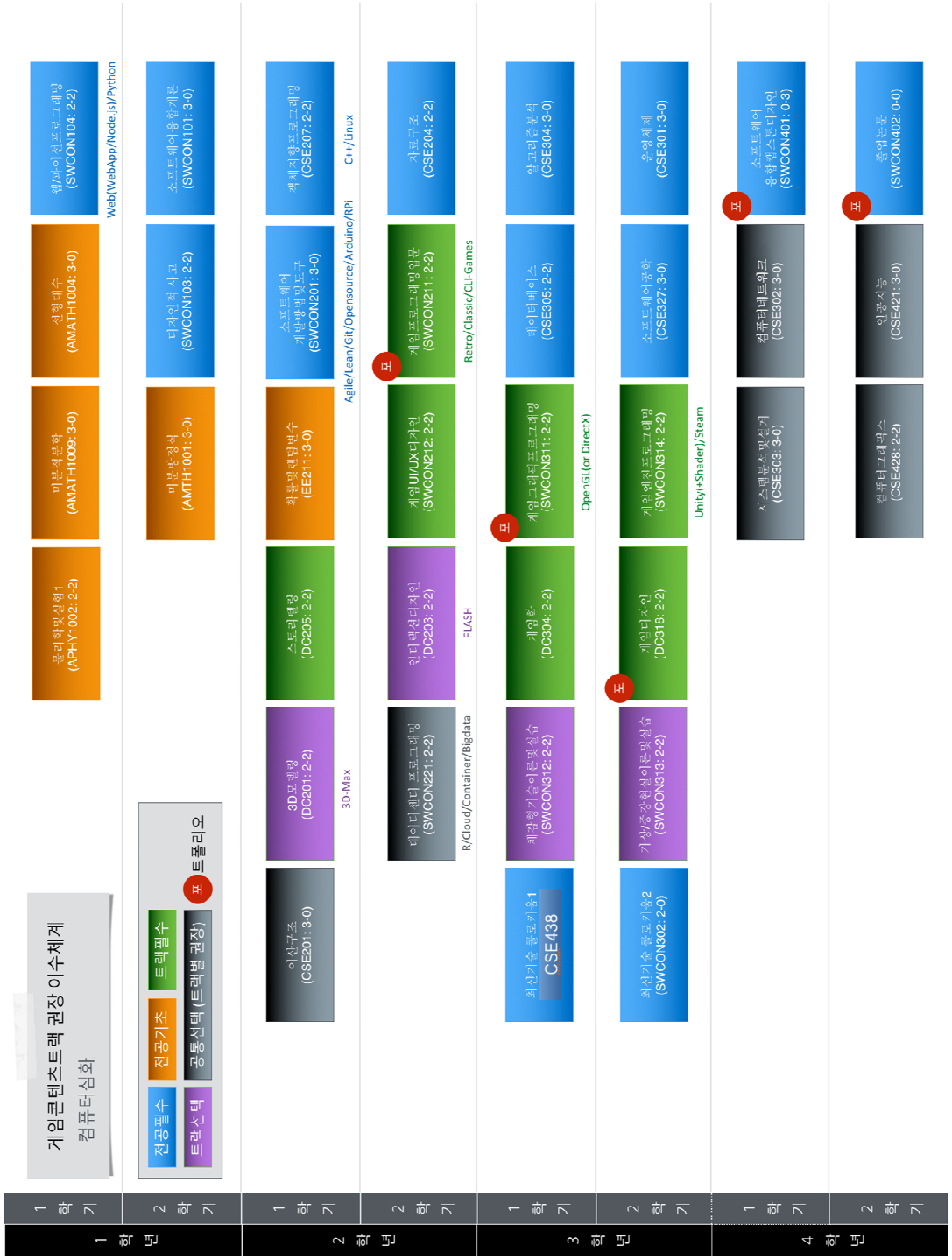
※ 비고의 '포'는 포트폴리오 교과목으로서, 반드시 교과목내 결과물을 개발하고, 학부 과정 동안 개인 실적으로서 관리해야 함
※ #4~#11은 컴퓨터공학과 과목 혹은 2019년 자체 개설을 고려함

순번	이수 구분	교과목명	학수번호	학점	시간				이수 학년	개설학기		PF 평가	비고
					이론	실기	실습	설계		1학기	2학기		
71	전공 선택	인공지능#9	CSE421	3	3				4	O			
72		정보보호#10	CSE423	3	2			1	4	O			
73		독립심화학습 1(소프트웨어융합학과)	SWCON363	3			3		3	O		P/F	
74		독립심화학습 2(소프트웨어융합학과)	SWCON364	3			3		3		O	P/F	
75		융합연구 1	SWCON241	1			2		2	O		P/F	
76		융합연구 2	SWCON242	1			2		2		O	P/F	
77		융합연구 3	SWCON341	1			2		3	O		P/F	
78		융합연구 4	SWCON342	1			2		3		O	P/F	
79		융합연구 5	SWCON441	1			2		4	O		P/F	
80		융합연구 6	SWCON442	1			2		4		O	P/F	
81		현장실습		※3)					2-4				
82		장기현장실습		※3)					2-4				
83		머신러닝#11	CSE331	3	2		2		3		O		

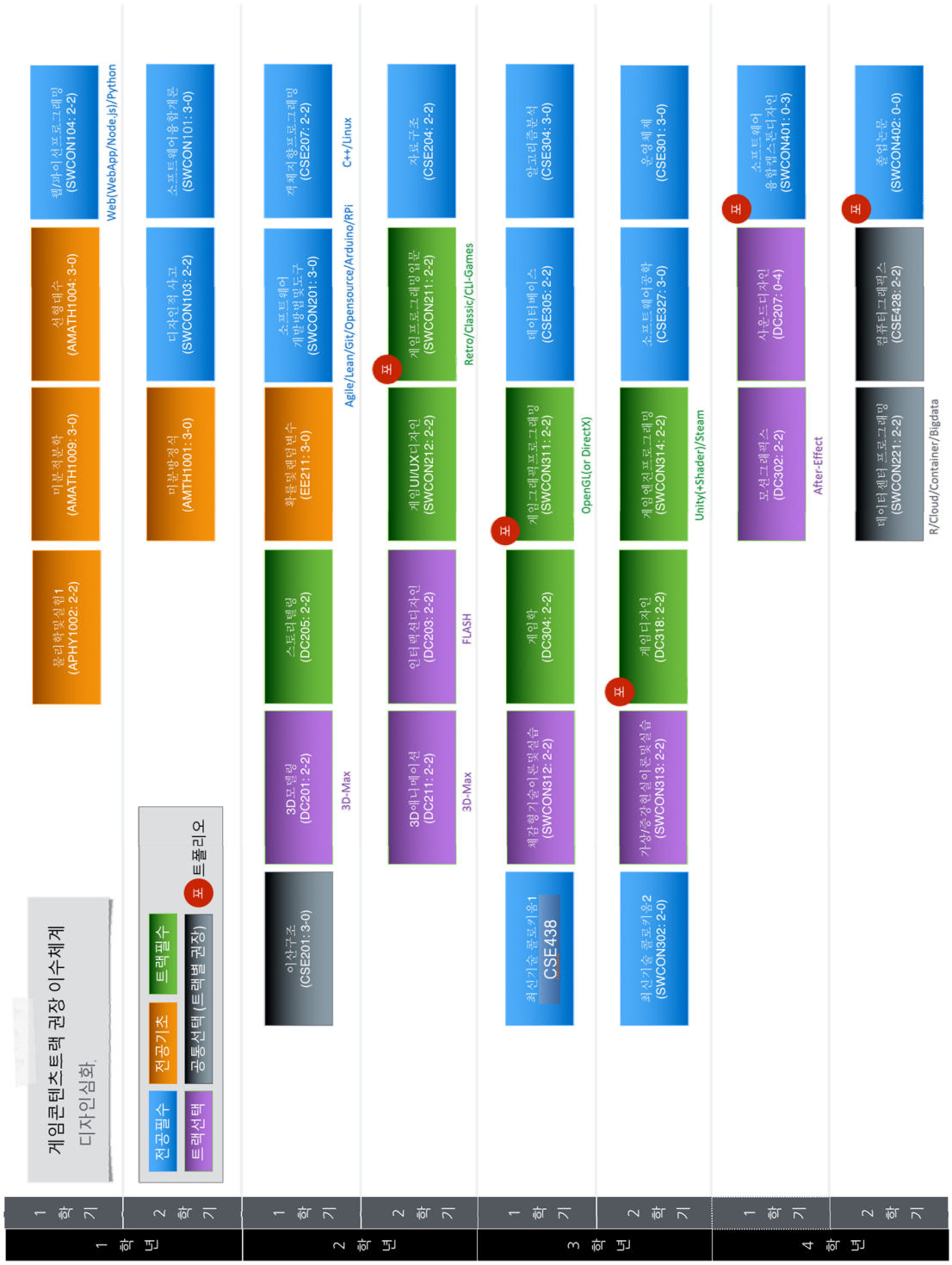
※ 1)독립심화학습 1/2(소프트웨어융합학과): 논문 작성 및 제출, 특허 출원, 소프트웨어 도서 출간, 첨단 기술 백서 출간 등을 설계 및 진행
※ 2)융합연구 1/2/3/4/5/6 : 전공과목 멘토링, 외국인 학생 프렌드십, 프로그래밍 재능기부, 오픈소스 기여, 프로그램 앱스토어 등록 등을 설계 및 진행하거나 소프트웨어 공모전, 창업 공모전, 프로그래밍 경진대회, 학술 공모전 등을 참가 및 출품
※ 3)현장실습/장기현장실습은 각 활동별로 3학점, 6학점, 12학점을 산학필수 학점으로 이수함

[별표2]

소프트웨어융합학과 게임콘텐츠트랙 교육과정 이수체계도 (컴퓨터심화과정)

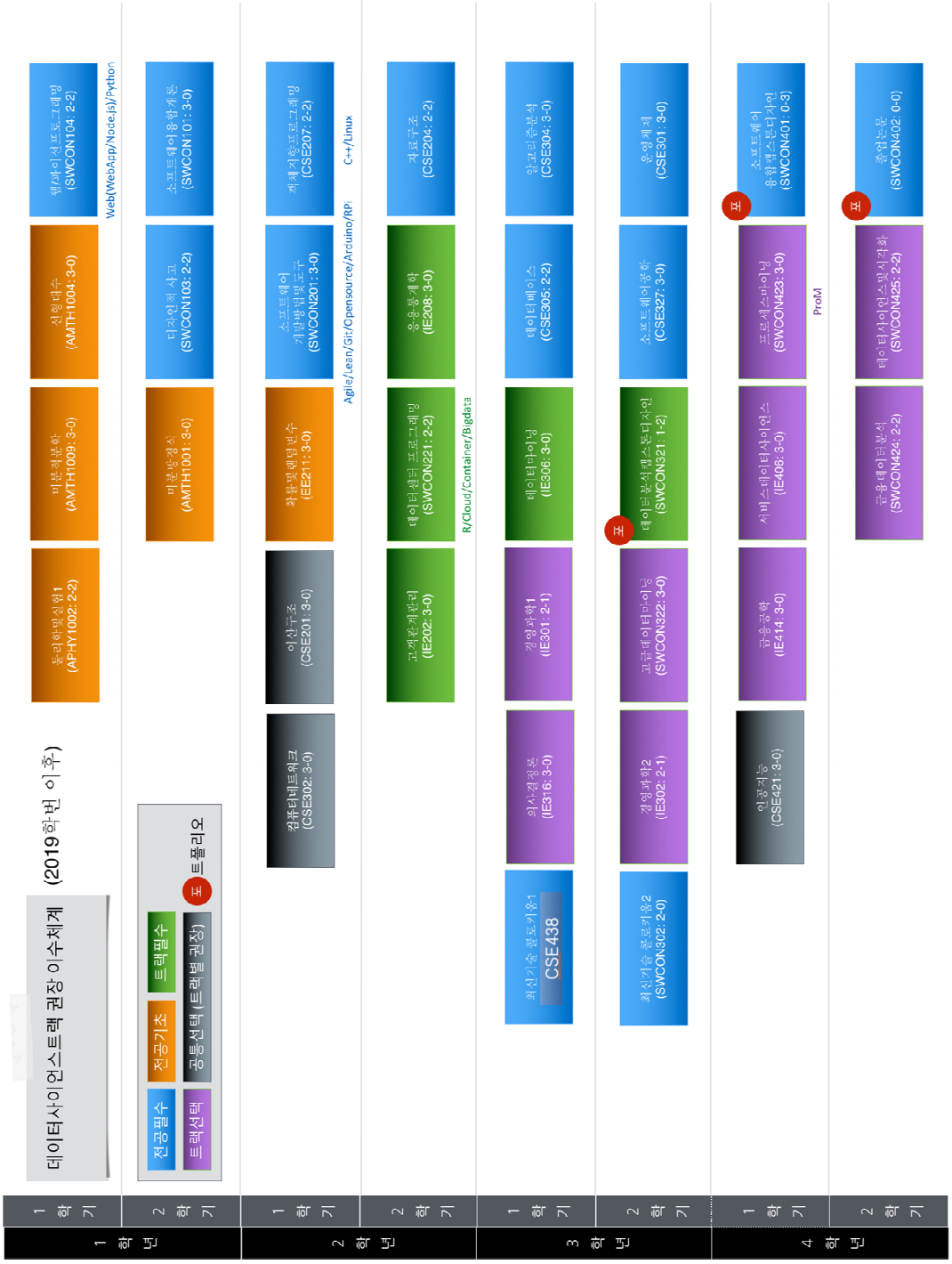


소프트웨어융합학과 게임콘텐츠트랙 교육과정 이수체계도 (디자인심화과정)

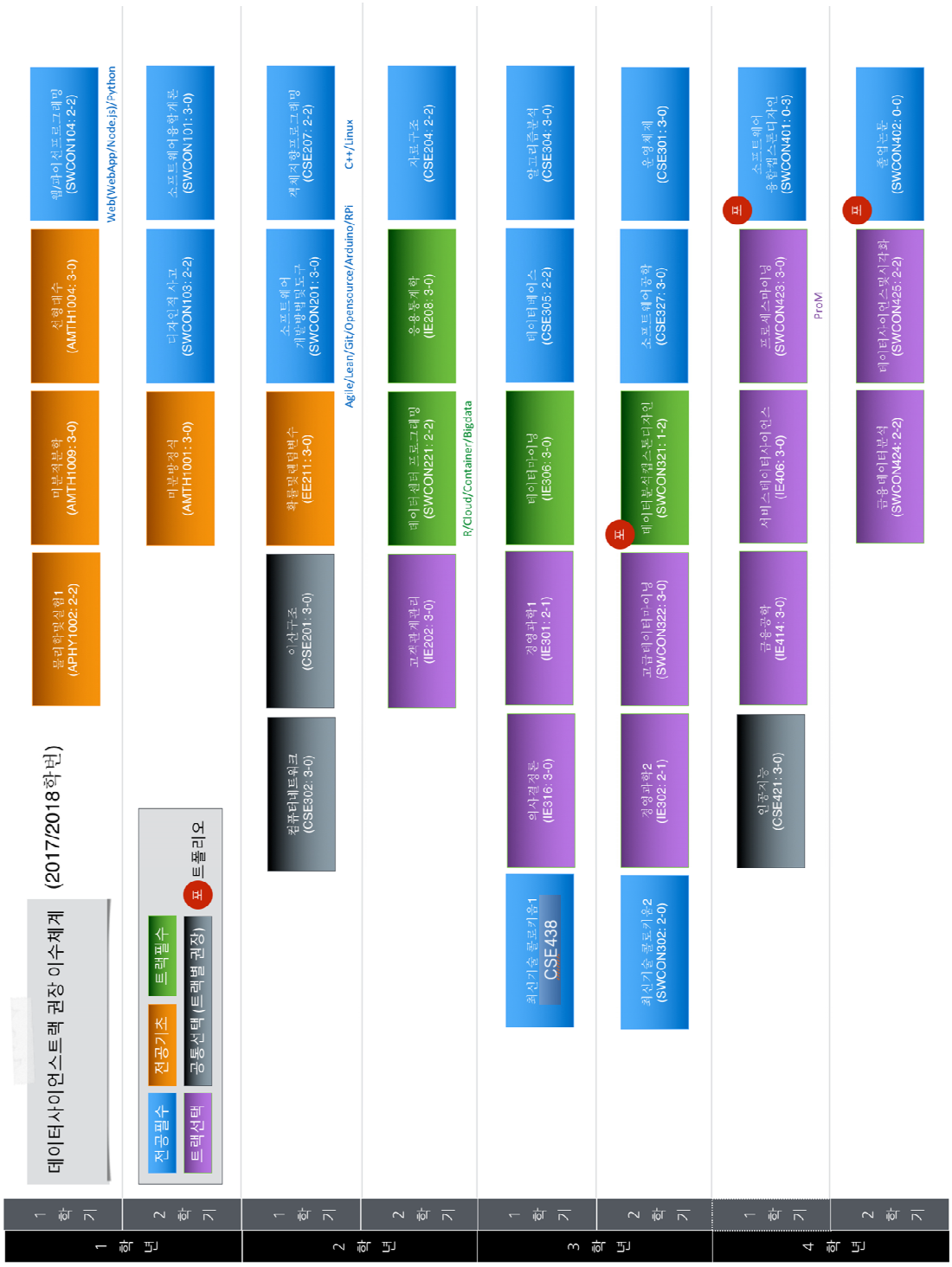


[별표3]

소프트웨어융합학과 데이터사이언스트랙 교육과정 이수체계도 (2019학번 이후)

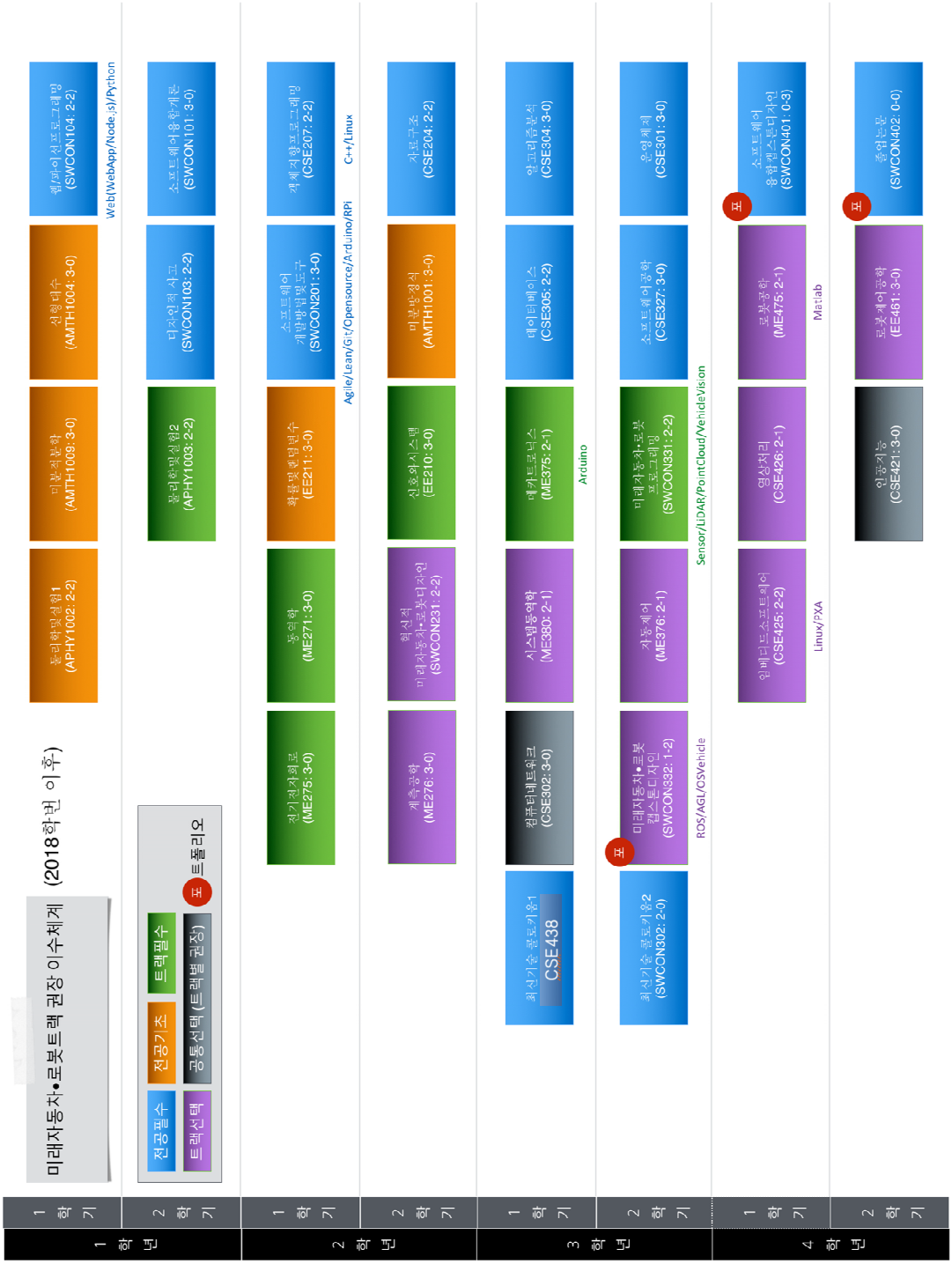


소프트웨어융합학과 데이터사이언스트랙 교육과정 이수체계도 (2018학번 이전)

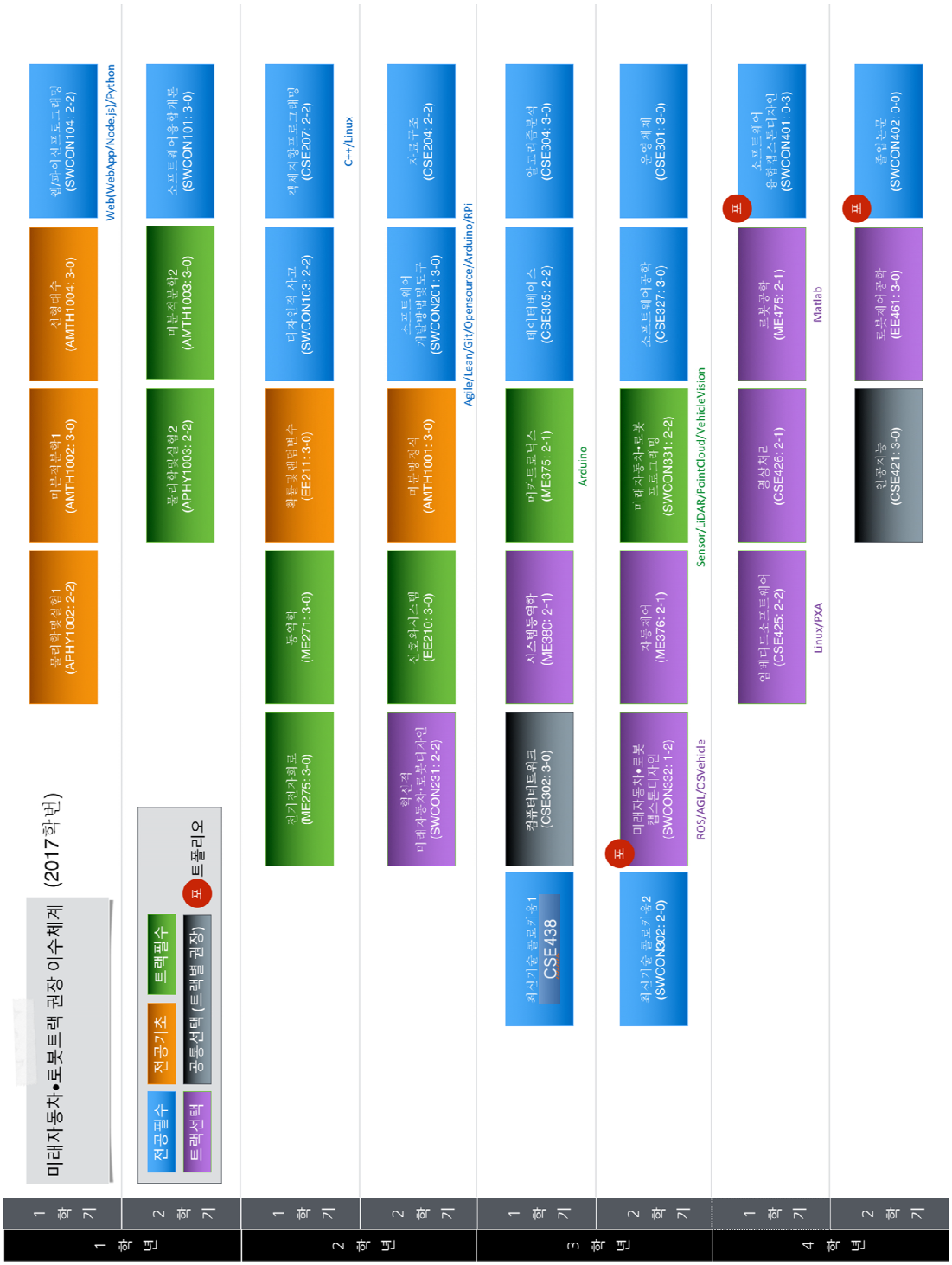


[별표4]

소프트웨어융합학과 미래자동차·로봇트랙 교육과정 이수체계도 (2018학번 이후)



소프트웨어융합학과 미래자동차·로봇트랙 교육과정 이수체계도 (2017학번)



[별표5]

가. 소프트웨어융합학과 선수과목 지정표 (2017/2018학번)

순번	전공명	교과목명(후수과목)			선수과목			비고
		학수번호	교과목명	학점	학수번호	교과목명	학점	
1	소프트웨어융합	CSE204	자료구조	3	CSE202	고급객체지향프로그래밍	3	
					CSE207	객체지향프로그래밍	3	
2		CSE304	알고리즘분석	3	CSE204	자료구조	3	
3		DC211	3D 애니메이션	3	DC201	3D 모델링	3	
4		APHY1003	물리학 및 실험 2	3	APHY1002	물리학 및 실험 1	3	
5		EE210	신호와시스템	3	AMTH1009	미분적분학	3	
					AMTH1003	미분적분학 2	3	
6		EE342	디지털신호처리	3	EE210	신호와시스템	3	
7		EE341	디지털 통신 1	3	EE210	신호와시스템	3	모두 수강
					EE211	확률 및 랜덤변수	3	
8		EE461	로봇제어공학	3	EE363	자동제어	3	
					ME376	자동제어	3	
9		IE208	응용통계학 (통계학2)	3	IE207	실험통계학 (통계학1)	3	
					EE211	확률 및 랜덤변수	3	
10		IE302	경영과학2 (심화)	3	IE301	경영과학 1 (기본)	3	
11		IE414	금융공학	3	IE201	경제성공학	3	
					EE211	확률 및 랜덤변수	3	
12		IE316	의사결정론	3	IE207	실험통계학 (통계학 1)	3	
					EE211	확률 및 랜덤변수	3	
13		ME375	메카트로닉스	3	ME275	전기전자회로	3	모두 수강
					ME271	동역학	3	
14		ME376	자동제어	3	ME375	메카트로닉스	3	
15		ME475	로봇공학	3	ME271	동역학	3	
16		SWCON424	금융데이터분석	3	IE312	금융공학	3	
17		SWCON402	졸업논문 (소프트웨어융합)	3	SWCON401	소프트웨어융합 캡스톤디자인	3	

※ 우측 선수과목 수강시에 좌측 후수과목 수강을 허용함
※ 교과목 담당교수가 수강 희망 학생과의 인터뷰를 통하여 선수과목 수준의 전공 이해 능력을 확인한 후, 선수과목의 필요가 없다고 판단한 경우에는, 선수과목 미이수 학생에 대해서도 해당 과목의 수강을 허용할 수 있음
※ 선수과목에 복수의 과목이 있는 경우는, 두 과목 중 한 과목을 수강하는 것을 의미함

나. 소프트웨어융합학과 선수과목 지정표 (2019학번 이후)

순번	전공명	교과목명(후수과목)			선수과목			비고
		학수번호	교과목명	학점	학수번호	교과목명	학점	
1	소프트웨어융합	CSE204	자료구조	3	CSE202	고급객체지향프로그래밍	3	
					CSE207	객체지향프로그래밍	3	
2		CSE304	알고리즘분석	3	CSE204	자료구조	3	
3		DC211	3D 애니메이션	3	DC201	3D 모델링	3	
4		APHY1003	물리학 및 실험 2	3	APHY1002	물리학 및 실험 1	3	
5		EE210	신호와시스템	3	AMTH1009	미분적분학	3	
					AMTH1003	미분적분학 2	3	
6		EE342	디지털신호처리	3	EE210	신호와시스템	3	
7		EE341	디지털 통신 1	3	EE210	신호와시스템	3	모두수강
					EE211	확률 및 랜덤변수	3	
8		EE461	로봇제어공학	3	EE363	자동제어	3	
					ME376	자동제어	3	
9		IE208	응용통계학 (통계학 2)	3	IE207	실험통계학 (통계학 1)	3	
					EE211	확률 및 랜덤변수	3	
10		IE302	경영과학 2 (심화)	3	IE301	경영과학1 (기본)	3	
11		IE414	금융공학	3	IE201	경제성공학	3	
					EE211	확률 및 랜덤변수	3	
12		IE316	의사결정론	3	IE207	실험통계학 (통계학 1)	3	
					EE211	확률 및 랜덤변수	3	
13		ME375	메카트로닉스	3	ME275	전기전자회로	3	모두수강
					ME271	동역학	3	
14		ME376	자동제어	3	ME375	메카트로닉스	3	
15		ME475	로봇공학	3	ME271	동역학	3	
16		SWCON424	금융데이터분석	3	IE312	금융공학	3	
17		SWCON402	졸업논문 (소프트웨어융합)	3	SWCON401	소프트웨어융합 캡스톤디자인	3	
18		IE202	고객관계관리	3	SWCON104	웹파이선프로그래밍	3	
19		IE306	데이터마이닝	3	IE202	고객관계관리	3	
20		IE419	서비스데이터사이언스	3	IE306	데이터마이닝	3	

※ 1~17은 2018학번 이상과 동일하며, 18~20사항을 2019년에 추가함
※ 우측 선수과목 수강시에 좌측 후수과목 수강을 허용함
※ 교과목 담당교수가 수강 희망 학생과의 인터뷰를 통하여 선수과목 수준의 전공 이해 능력을 확인한 후, 선수과목의 필요가 없다고 판단한 경우에는, 선수과목 미이수 학생에 대해서도 해당 과목의 수강을 허용할 수 있음
※ 선수과목에 복수의 과목이 있는 경우는, 두 과목 중 한 과목을 수강하는 것을 의미함

[별표6]

2017학년도 교육과정 경과조치

순번	신규개설 및 대체인정 교과목				기존 교과목 (폐지 및 삭제)			비고
	이수구분	학수번호	교과목명	학점	학수번호	교과목명	학점	
1	-	-	-	-	EE202	회로이론	3	교육과정에서 삭제됨
2	전공선택	AMTH1003	미분적분학 2	3	-	-	-	미래자동차·로봇 트랙의 트랙필수 과목으로 추가됨
3	전공선택	APHY1003	물리학 및 실험 2	3	-	-	-	미래자동차·로봇 트랙의 트랙필수 과목으로 추가됨
4	전공선택	SWCON363	독립심화학습 1	3	-	-	-	공통선택 과목 신규개설
5	전공선택	SWCON364	독립심화학습 2	3	-	-	-	공통선택 과목 신규개설
6	전공선택		현장실습	※1)	-	-	-	공통선택 과목 신규개설
7	전공선택		장기현장실습	※1)	-	-	-	공통선택 과목 신규개설
8	전공선택	CSE440	IoT소프트웨어	3	CSE425	임베디드소프트웨어	3	과목명/과목코드 변경됨
9	전공선택	ME380	시스템동역학	3	ME373	시스템모델링	3	과목명/과목코드 변경됨
10	전공선택	SWCON241	융합연구 1	1	SWCON241	전공과목멘토링 1	1	과목명/과목코드 변경됨
					SWCON242	전공과목멘토링 2	1	과목명/과목코드 변경됨
					SWCON345	전공과목멘토링 3	1	과목명/과목코드 변경됨
11	전공선택	SWCON242	융합연구 2	1	SWCON243	외국인학생프렌드십	1	과목명/과목코드 변경됨
12	전공선택	SWCON341	융합연구 3	1	SWCON244	프로그래밍재능기부 1	1	과목명/과목코드 변경됨
					SWCON245	프로그래밍재능기부 2	1	과목명/과목코드 변경됨
					SWCON346	프로그래밍재능기부 3	1	과목명/과목코드 변경됨
13	전공선택	SWCON342	융합연구 4	1	SWCON341	소프트웨어공모전	1	과목명/과목코드 변경됨
					SWCON342	창업공모전	1	과목명/과목코드 변경됨
					SWCON343	프로그래밍경진대회	1	과목명/과목코드 변경됨
14	전공선택	SWCON441	융합연구 5	1	SWCON344	특허출원	1	과목명/과목코드 변경됨
					SWCON347	소프트웨어도서출간	1	과목명/과목코드 변경됨
					SWCON348	첨단기술백서출간	1	과목명/과목코드 변경됨
15	전공선택	SWCON442	융합연구 6	1	SWCON441	오픈소스기여	1	과목명/과목코드 변경됨
					SWCON442	학술공모전	1	과목명/과목코드 변경됨
					SWCON443	프로그램앱스토어등록	1	과목명/과목코드 변경됨
16	전공선택	DC318	게임디자인	3	DC314	캡스톤디자인기반 게임디자인	3	과목명/과목코드 변경됨
17	전공선택	CSE331	머신러닝	3	-	-	3	공통선택 과목 신규개설

※ 1) 현장실습/장기현장실습은 각 활동별로 3학점, 6학점, 12학점을 산학필수 학점으로 이수함

[별표7]

2018학년도 교육과정 경과조치

순번	신규개설 및 대체인정 교과목				기존 교과목 (폐지 및 삭제)			비고
	이수구분	학수번호	교과목명	학점	학수번호	교과목명	학점	
1	전공선택	DC318	게임디자인	3	DC314	캡스톤디자인기반 게임디자인	3	과목명/과목코드 변경됨
2	전공선택	CSE331	머신러닝	3	-	-	3	공통선택 과목 신규개설

[별표8]

소프트웨어융합학과 교과목 해설

[전공 필수 - 융합 필수]

• **소프트웨어융합개론 (Introduction to Software Convergence Engineering)**

소프트웨어융합에 대한 기초적인 지식을 트랙별 개론 및 콜로키움 형태로 학습한다. 먼저 소프트웨어에 대한 기초적인 배경을 습득한 후, 소프트웨어융합학과를 구성하는 3가지 트랙에 대한 역사와 기초지식 이론적 지식을 학습한다. 트랙에 참여하는 전임/겸직 교수님들과 관련 기업체 초청강사님을 통한 트랙별 콜로키움을 수행하여, 트랙에 대한 산업 및 학술에 대한 최신 현황을 학습하여, 추후 트랙의 선택에 대한 실무적 배경을 제공한다. 특별히 게임콘텐츠 트랙에 대해서는 게임에 대한 역사를 통한 해당 분야 이해와 함께 보드 게임 개발을 통한 게임의 이해를 수행한다.

Learn basic knowledge of software convergence by track and colloquium type. After acquiring basic background of software, students will learn the history and basic knowledge of the three tracks that make up the software convergence course. Perform track-by-track colloquiums with full-time / part-time professors participating in the track and invited lecturers from related companies to learn the current status of industry and academic track and provide a practical background for future track selection. Especially, for the game content track, we understand the field through game history and understand the game through board game development.

• **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.

• **최신기술콜로키움 1 (Latest Technology Colloquium 1)**

소프트웨어 기반의 융합 분야에 대한 최신 기술을 해당 분야 전문가를 초청하여 듣도록 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.

• **최신기술콜로키움 2 (Latest Technology Colloquium 2)**

소프트웨어 기반의 융합 분야에 대한 최신 기술을 해당 분야 전문가를 초청하여 듣도록 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.

• **소프트웨어융합캡스톤디자인 (Capstone Design in Software Convergence Engineering)**

소프트웨어융합전문프로그램의 졸업을 위한 합격여부를 결정한다. 트랙별 주제에 부합하는 소프트웨어의 개발, 졸업 논문 작성 및 심사 발표를 수행한다. 결과물은 개인의 공식 포트폴리오로 관리하도록 한다.

Decide whether to pass for graduation from the Software Convergence Program. Develop software that meets

track-specific themes, write graduation theses, and present audition announcements. The results should be managed by an individual's official portfolio.

• **졸업논문(소프트웨어융합) (Graduation Thesis:Software Convergence)**
소프트웨어융합학과는 '소프트웨어융합캡스톤디자인'을 이수하고, 결과물을 소프트웨어융합학과가 제시한 양식의 연구논문 형태로 제출하는 것으로 "졸업논문" 합격 여부를 결정한다.
The Software Convergence Department decides whether or not to pass the "Graduation Thesis" by completing the "Software Convergence Capstone Design" and submitting the results in the form of research papers presented by the Software Convergence Department.

[전공 필수 - 소프트웨어 필수]

• **객체지향프로그래밍 (Object Oriented Programming)**
기초적인 C++ 프로그래밍을 익히는 것이다. 이를 위해, C++ 프로그램의 기본적인 구조, 데이터 형, 변수, 함수, 분기문, 반복문, 재귀 프로그래밍, 문자 입출력, 배열, 포인터, Class, 유전의 법칙 등 고급 C++ 프로그래밍을 위한 기초를 배운다. 교재는 많은 예제 프로그램을 포함하여, 초보자도 쉽게 프로그래밍에 친숙해질 수 있고, 이론과 실습을 병행함으로써 컴퓨터 공학을 비롯한 소프트웨어융합대학에서 필요한 기초적인 프로그래밍 능력을 배양한다.
Learning basic C ++ programming. To do this, you will learn the fundamentals of advanced C ++ programming, including the basic structure of a C ++ program, data types, variables, functions, branch statements, looping statements, recursive programming, character input and output, arrays, pointers, classes and rules of inheritance. The textbook includes a lot of example programs, it is easy for beginners to become familiar with programming, and the theory and practice are combined to cultivate the basic programming ability necessary for computer science and other electronic information colleges.

• **웹/파이선프로그래밍 (Web and Python Programming)**
웹 프로그래밍과 파이선 프로그래밍의 기초적인 내용을 배우도록 한다. 웹 프로그래밍은 HTML5/CSS3/Javascript를 사용하는 WebApp을 개발함으로서, 클라이언트 개발을 가능하게 한다. 아울러 Node.js를 통한 서버 프로그래밍까지 할 수 있도록 한다. 파이선은 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.

• **소프트웨어개발방법 및 도구 (Software Development Methods and Tools)**
최근에 가장 많이 쓰이는 소프트웨어 개발 방법과 도구에 대한 이해와 실습을 수행한다. 소프트웨어 개발방법론으로는 Agile 개발 방법론과 Lean 개발 방법론에 대해서 학습한다. 소프트웨어 개발 관리를 위해서는 Git 도구에 대해서 학습한다. 그리고 오픈소스 라이센싱에 대한 이해를 갖는다. 마지막으로 대표적인 physical programming인 Arduino와 RaspberryPi에 대해서 배우도록 한다.
This course deals with understanding and practicing the most frequently used software development methods and tools. As a software development methodology, we learn about agile development methodology and Lean development methodology. Learn about Git tools for software development management. And have an understanding of open source licensing. Finally, learn about typical physical programming Arduino and RaspberryPi.

• **자료구조 (Data Structure)**
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.

Learn data structures such as data abstraction, arrays, lists, stacks, queues, trees, graphs, and algorithms that can leverage such data structures. This course allows students to expand their knowledge of computer science and improve their programming skills.

• **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.

Students will have a basic understanding of algorithms and learn the types of representative algorithms. We classify algorithm methods into divide-and-conquer, dynamic programming, greedy algorithms, and branch-and-bound. We also look at basic complexity issues. After completing this course, students develop the ability to find solutions to new problems.

• **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, 스케쥴링, 교착, 메모리관리, 가상메모리관리, 파일 시스템, 디스크 스케쥴링, 정보공유, 보호/보안 및 분산운영체제와 같은 주제를 중점적으로 학습한다.

The operating system monitors the execution of user programs and various resource allocations such as peripherals and memory space. This course introduces the concept of multiprogramming, time-sharing, and asynchronous processors. Topics include synchronization, scheduling, deadlock, memory management, virtual memory management, file system, disk scheduling, information sharing, protection / security and distributed operating systems.

• **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 introduces the theory of basic components of database system, introduces database design theory based on ER-model and relational data model, and theory that implements database management system such as concurrency control, recovery, and transaction management.

• **소프트웨어공학 (Software Engineering)**

소프트웨어 공학 분야는 프로그램이 방대하고, 오랜 기간 동안 많은 프로그래머들이 참여하는 경우 발생되는 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 vast and involve a lot of programmers for a long time. Topics covered in this course include design and configuration of programming projects, test and program reliability, nature and causes of software costs, coordination among various programmers, and user-friendly interface design and documentation.

[전공 선택 - 데이터사이언스 트랙 - 트랙 필수]

• **응용통계학 (Applied Statistics)**

통계학이론 중에서 확률통계이론의 응용력을 확대할 수 있는 기법과 확장된 이론을 체득할 수 있도록 한다. 다루어지는 내용은 시료분포, 추정, 검정, 중선형 및 곡선회귀, 직교다항식, 샘플링방법, 요인배치법, 교락법, 직교배열법, 파라메터 및 허용차 설계 방법 등이다.

Students will be able to learn techniques and techniques for expanding the application of probability statistical theory in statistical theory. The topics covered include sample distribution, estimation, calibration, midline and curve regression, orthogonal polynomials, sampling methods, factoring, delineation, orthogonal arraying, and parameter and tolerance design methods.

- **데이터센터프로그래밍 (Datacenter Programming)**

데이터센터와 서버 프로그래밍에 대한 이론을 이해하고 실습을 수행한다. 대표적인 기술로서 클라우드 컴퓨팅(OpenStack 등), 콘테이너 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.

- **데이터마이닝 (Data Mining)**

데이터 마이닝이란 대량의 데이터에서 의미 있는 패턴과 규칙을 발견하기 위해 자동적인 또는 준자동적인 방법에 의해 데이터를 조사하고 분석하는 절차이다. 본 과목은 데이터 마이닝의 기초적인 개념들과 그 적용법들을 제공한다. 주요 논제로 decision trees, classification, association, clustering, statistical modeling, Bayesian classification, k-nearest neighbors, CART 등을 다룬다.
Data mining is the process of examining and analyzing data by automatic or semi-automatic methods to find meaningful patterns and rules in large amounts of data. This course provides basic concepts of data mining and its applications. Topics include decision trees, classification, association, clustering, statistical modeling, Bayesian classification, k-nearest neighbors, and CART.

- **데이터분석캡스톤디자인 (Data Analysis Capstone Design)**

데이타 분석의 전주기적인 과정을 실질적인 데이터를 통해서, 이해하고 실습하도록 한다.
Students will understand and practice the whole process of data analysis through practical data.

[전공 선택 - 데이터사이언스 트랙 - 트랙 선택]

- **고객관계관리 (Customer Relationship Management)**

CRM은 고객정보를 이용해서 고객과의 관계를 유지, 확대, 개선시킴으로써 고객의 만족과 충성도를 제고하고, 기업 및 조직의 지속적인 운영, 확장, 발전을 추구하는 고객관련 제반 프로세스 및 활동을 연구하는 학문이다. 따라서 이에 따른 내용은 분석적 CRM, 운영적 CRM, 활용적 CRM, 고객정보, 데이터베이스 마케팅, 애프터 마케팅, 관계마케팅 등이 다루어지게 된다.
CRM is the study of customer-related processes and activities that enhance customer satisfaction and loyalty by maintaining, expanding, and improving relationships with customers using customer information, and pursuing continuous operation, expansion and development of companies and organizations to be. Therefore, analytical CRM, operational CRM, applied CRM, customer information, database marketing, aftermarket, and relationship marketing are covered.

- **경영과학 1 (Management Science 1)**

계량적 방법을 통하여 어떻게 최선의 의사결정을 내릴 수 있는지 수강자들에게 관련된 이론을 체계적으로 소개하고 이를 현실사회의 문제에 실제로 적용할 수 있도록 훈련시킨다. 선형계획법 및 그 응용분야가 주로 다루어진다.
An introduction to deterministic models in operations research with special emphasis on linear programming. Topics include simplex algorithm, transportation and assignment algorithms and their engineering applications.

• **경영과학 2 (Management Science 2)**

계량적 방법을 통하여 어떻게 최선의 의사결정을 내릴 수 있는지 수강자들에게 관련된 이론을 체계적으로 소개하고 이를 현실사회의 문제에 실제로 적용할 수 있도록 훈련시킨다. 주요 논제는 네트워크이론, 동적계획법, 정수계획법, 게임이론, 의사결정이론, 예측이론 등이다.

How to make the best decision through quantitative method Introduce the theories related to the learners systematically and train them to apply them to the problems of real society. Linear programming and its application areas are mainly covered.

• **의사결정론 (Decision Analysis)**

복잡한 의사결정 문제를 체계적으로 설계하고 분석하는 데 도움이 되는 기법들을 소개한다. 주요 주제는 대안개발, 불확실성 분석, 대안평가 및 선택기법 등이며 의사결정나무, 영향도, AHP, DEA 등의 의사결정 분석기법도 논의된다.

An introduction to basic techniques for design and analysis of complex decision making problems. Topics include development of alternatives, uncertainty anaysis, evaluation and selection of alternatives. Various techniques such as decision tree analysis, influence diagram, AHP as well as DEA will be discussed.

• **고급데이터마이닝 (Advanced Data Mining)**

기초 데이터마이닝 이후의 고급 과정에 대해서 다룬다. 학생들은 고급 과정에 해당하는 이론 내용과 함께 Case-Study를 통한 실제 사례의 이해와 실제 문제의 해결을 직접 해보도록 한다.

This course deals with advanced processes after basic data mining. Students will be able to understand the actual cases through the case-study, as well as the theoretical content corresponding to the advanced course, and solve the actual problems directly.

• **금융공학 (Financial Engineering)**

미래의 금융환경에 대한 불확실성의 증대 속에, 투자자의 다양한 투자전략수립, 고객의 니즈에 맞는 신상품의 지속적인 개발, 금융자산에 대한 위험관리에 관한 수학적, 공학적 이론에 관한 학습을 한다.

The goal of this course is to develop leading-edge skills and provide new information on financial engineering. Topics such as deterministic cash flow analysis, single-period random cash flow analysis, and derivative securities will be discussed.

• **프로세스마이닝 (Process Mining)**

프로세스 마이닝은 누적된 로그 데이터로부터 프로세스를 도출하고, 분석 및 개선하기 위한 기법이다. 제조, 서비스, 인터넷 등의 정보시스템의 다양한 데이터를 바탕으로 비즈니스 운영 과정을 분석하는 방법을 학습한다.

Process mining is a technology for discovering, analyzing, and improving process from historical log data. Students are taught how to analyze operational procedures based on various data that have been accumulated in manufacturing, service, and internet information systems.

• **서비스데이터사이언스 (Service Data Science)**

서비스 경영은 서비스 사회의 서비스 경제를 유지하고 창출하는 경영전략을 수립하고 이행하는 학문이라 할 수 있다. 이에 따른 주요 내용은 서비스 마케팅, 확장된 서비스 마케팅, 서비스 생산성, 서비스 수행도, 서비스 품질, 서비스 관계마케팅, 서비스 인터넷 마케팅 등을 다루게 된다.

Service Management is a disciplined business strategy to create and sustaining service economy for service society. Major topics include service marketing, service mixed marketing, service productivity, service performance, service quality, service relationship marketing, service internet marketing.

• **금융데이터분석 (Financial Data Analysis)**

금융공학의 다양한 기법을 기반으로 금융시장의 데이터를 분석하는 방법을 익힌다. 주식시장 데이터와 경제지표 데이터 등의 분석을 위해 사용되는 MS Excel, R, Python, Matlab을 중점적으로 학습하여 다양한 금융데이터 분석모델을 구현하게 된다.

This course introduces students to analyzing financial data based on various financial engineering models. Students will learn how to implement various techniques for analyzing stock market data and economic data using MS Excel, R, Python, and Matlab.

• **데이터사이언스 및 시각화 (Data Science and Visualization)**

데이터 분석 결과를 쉽게 이해할 수 있도록 시각적으로 표현하고 전달하는 방법론에 대해서 학습한다. 특히 빅데이터의 시각화 측면에서, 모든 데이터를 살펴보는 것에 제약이 따르므로, 시각화의 기술적인 요소와 더불어 데이터를 요약하고, 한 눈에 살펴볼 수 있도록 돕는 시각화 방법론적 요소에 대한 이론과 실습을 수행한다. 아울러 텍스트 마이닝, 이미지 마이닝과 같은 다양한 데이터사이언스 내용들도 함께 다루도록 한다.

Learn how to visualize and communicate data analysis results easily. In particular, since visualization of Big Data has restrictions on looking at all the data, the theoretical and practical aspects of the visualization methodological elements that summarize the data together with the technical elements of visualization and help to look at a glance are performed . We will also cover various data science contents such as text mining and image mining.

[전공 선택 - 미래자동차·로봇 트랙 - 트랙 필수]

• **동역학 (Dynamics)**

기계역학의 기초가 되는 운동학과 운동역학을 주로 취급하여 힘의 효과와 운동에 대한 해석과 기초역학의 이해능력을 다룬다.

This course deals with the basic principles of mechanics, such as kinematics and kinematics. It deals with the effects of force, motion, and understanding of fundamental mechanics.

• **전기전자회로 (Basic Electric Circuits)**

전자 전기 회로의 기본 개념과 설계가 소개한다. Circuit elements, parameters, resistance, capacitance, inductance, impedance, admittance, charge, current, voltage, energy, power, Kirchhoff's laws, superposition, periodic functions, RMS values, phasor, resonance, bandwidth, balanced three phase systems, steady and transient states 등에 관하여 학습한다. Basic analysis and design of electric circuits are introduced. Circuit elements and parameters, resistance, capacitance, inductance, impedance, admittance, charge, current, voltage, energy, power, Kirchhoff's laws are studied. Superposition, periodic functions, RMS values, phasor, resonance, bandwidth, balanced three phase systems, steady states and transient states are also studied.

• **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 강의. 과제물은 C/C++을 이용하여 프로그래밍 능력을 향상시킨다.

This course deals with basic concepts and techniques of mathematical representation, analysis, and signal synthesis of continuous and discrete signals and systems. Fundamental theory of signal and system analysis method based on Fourier transform, Z-transform, Laplace transform, and application examples of filtering and modulation. Assignments improve programming skills using C / C ++.

• **미래자동차·로봇프로그래밍 (Future Cars·Robot Programming)**

미래자동차와 로봇의 하드웨어를 이해하고 핵심 기술 소프트웨어를 직접 개발하도록 한다. 먼저, 학생들은 임베디드 하드웨어를 다

루기 위한 기본 기술을 학습한다. 그리고 센서, LiDAR, Point-Cloud, 컴퓨터 비전/인식 등과 같은 다양한 요소 기술의 이론을 학습하고 실습을 통하여 직접 개발한다.
Understand the hardware of future cars and robots, and develop core technology software directly. First, students learn the basic skills to handle embedded hardware. Students learn the theory of various element technologies such as sensors, LiDAR, Point-Cloud, and computer vision / recognition, and develop them through hands-on experience.

• **메카트로닉스 (Mechatronics)**
기계와 전자가 결합된 형태를 메카트로닉스라 하고 있으며 필연적으로 전산에 대한 부분도 포함되고 있다. 기구학, 전장용소, 열부품 그리고 유체부품 등을 기계부분으로 강의 되며, 이에 대한 제어부분인 전자와 소프트웨어 및 그 기계와의 인터페이스에 대한 학습을 제공한다. 수강생들은 실습을 통하여 각자 자유 제목으로 선정될 수 있는 학기 프로젝트를 완성해야한다.
The combination of machine and electron is called mechatronics, and inevitably includes part of computerization. Mechanics, electric field, heat parts and fluid parts are taught in the mechanical part, and the control part of electronics, software and interface with the machine are provided. Students are required to complete a semester project that can be selected as a free title through hands-on training.

[전공 선택 - 미래자동차·로봇 트랙 - 트랙 선택]

• **계측공학 (Measurements in Mechanical Systems)**
일반적 계측기의 구성, 계측기의 측도설정, 측정오차의 원인과 측정결과의 처리방법을 이해한 후에 각종 기초 전기 계측기의 원리와 응용법을 다룬다. 주요한 내용으로는 변형도, 힘, 토크 및 압력의 측정, 유량계의 기초원리, 열전대의 응용과 온도측정, 열량의 측정 등에 대해서 배운다.
This course deals with the construction of a general instrument, setting the instrument's measure, understanding the causes of measurement errors, and how to handle measurement results, and then discuss the principles and applications of various basic electrical instruments. Topics include strain, force, torque and pressure, basic principles of flowmeters, applications of thermocouples, temperature measurements, and calorimetric measurements.

• **자동제어 (Automatic Controls)**
선형 자동제어계에 대한 기본 개념에서부터 회로 제어이론과 그 응용을 다룬다.
With recent developments in electronic industry automatic control becomes one of the most important subjects in modern engineering education. This course deals with be basic mathematical and computational tools for modeling and analysis of dynamic system to be controlled and s unified methodology to identify, model, analyze, design, and simulate dynamic systems in various engineering disciplines. Based on these foundations principal concepts of linear feedback control will be taught. MATLAB will be introduced and used as a practical computation tool. It is desired that students have minimum background in dynamics, and ordinary differential equations.

• **혁신적미래자동차로봇디자인 (Innovative Future Car and Robot Design)**
미래 자동차디자인의 혁신적 접근 방법과 창조적 조형 개발을 위하여 문제해결 및 아이디어 전개 등 기초적인 디자인 프로세스를 실습을 통해 학습한다.
Students will learn basic design process such as problem solving and idea development for the innovative approach of future automobile design and creative modeling through hands - on experience.

• **시스템동역학 (System Dynamics)**
역학시스템의 수학적 모델링과 응답을 다루는 본 교과는 역학시스템의 모델링과 해석을 완벽히 다루고 제어시스템의 해석 및 설계를 위한 개론을 제시한다. 제어 및 역학시스템의 해석적 연구를 위한 내용으로 구성되어 있으며 이 과목을 듣기 위해서는 수강생들

은 미분방적식, 행렬-벡터 해석 그리고 회로해석에 대한 기본적인 지식이 요구된다.
This course deals with the mathematical modeling and response of dynamics systems and presents an overview for the analysis and design of control systems. It is composed of content for analytical study of control and dynamics systems. Students are required to have basic knowledge of differential equation, matrix-vector analysis and circuit analysis.

- **미래자동차·로봇캡스톤디자인 (Future Cars·Robot Cap Stone Design)**

미래 자동차와 로봇에 대하여 학습한 내용을 기반으로, 직접 기초적인 형태의 자동차와 로봇을 설계하고 개발하여 본다. 본 교과에서 개발하는 결과물은 추후 소프트웨어융합 캡스톤디자인의 기반으로 활용한다. 이를 위하여 오픈소스 자동차/로봇 소프트웨어 플랫폼인 ROS, AGL, OSVehicle 등에 대한 이해와 실습을 수행한다.
Based on the learning about future cars and robots, we design and develop directly basic forms of automobiles and robots. The results of this course will be used as a basis for future software fusion capstone design. To do this, we will conduct an understanding and practice on ROS, AGL, and OSVehicle which are open source automotive / robot software platforms.

- **디지털신호처리 (Digital Signal Processing)**

디지털신호처리 시스템의 기본이 되는 디지털필터(FIR, IIR 필터) 설계방법, 입출력 신호의 주파 특성을 해석하는 방법, Z-변환의 성질 및 응용 예를 강의하고 실제적인 다양한 응용 시스템을 MATLAB 이용하여 직접 프로그래밍 해봄으로써 공학적인 응용력을 배양한다.(선수과목:신호와시스템)
This course will study basic theory, filter design about necessity for system analysis and apply method for computer simulation, acoustics, image processing and communication software. These are all done with signal and system background. The main topics are Z-transform, system transform coefficient, filtering, modulation, Fourier Transform, sampling theory, etc.(Prerequisite:Signals and Systems)

- **디지털통신 1 (Digital Communications 1)**

확률 이론, 신호와 시스템의 시간 영역과 주파수 영역에서의 분석 방법을 기반으로 통신 시스템을 이해하고 분석하는 능력을 배운다. 먼저 AM, FM과 같은 아날로그 통신에 견주어 디지털 통신이 갖는 장점과 새로운 기능을 이해한다. 구체적으로, 디지털 정보를 전송하기 위한 샘플링 및 양자화 기법, 기저대역 및 통과대역 변조 방식과 최적 수신을 위한 정합 필터 및 검출 기법을 배우고, M진 통과대역 변조 방식과 그 성능을 분석하는 방법을 배운다.(선수과목:신호와 시스템, 확률 및 랜덤변수)
Students learn the basic functions of communication systems and how to analyze them based on the probability theory and time-domain and frequency-domain analysis of signals and systems. First, they learn the advantages of digital communications compared with analog communications and basic processes to obtain digital information such as sampling and quantization. Then, the course deals with baseband and bandpass modulation techniques to transmit digital information reliably over a channel. High order bandpass modulation techniques will be also introduced with their performance analysis.(prerequisites:Signals and Systems, Probability and Random Variables)

- **로봇공학 (Introduction to Robotics)**

로봇 매니퓰레이터를 위주로 로봇 동작과 제어에 관련된 수학적 도구와 알고리즘 등을 학습하고 이를 현실에서의 사용하기 위한 응용기법을 학습한다. 구체적으로 본 과목에서는 좌표계 설정, Homogeneous Transform, Forward/Inverse Kinematics, Forward/Inverse Dynamics, 위치 및 컴플라이언스 제어, 경로설정, 장애물 회피, 여유자유도 로봇과 같은 기초적 개념과 응용 기법 등을 학습한다.
Students will learn mathematical tools and algorithms related to robot operation and control, focusing on robot manipulators, and learn application techniques for their use in real life. In this course, basic concepts and application methods such as coordinate system setting, homogeneous transform, forward / inverse kinematics, forward / inverse dynamics, position and compliance control, path setting, obstacle avoidance and redundancy robot are studied.

• **로봇제어공학 (Robotics Engineering)**

센서, 액츄에이터, 지능의 융합으로 이루어진 로봇의 제어를 위해서 요구되는 경로계획, 피드백 제어기 구성 및 로봇 시뮬레이션 방법을 학습한다. 또한 로봇제작에 사용되는 센서와 구동기 및 비전시스템 등에 대하여 소개한다.

This class is about navigation, feedback control and robot simulation, which are the key topics to develop a robot system. Also, it includes the brief introduction about sensor, actuator and intelligence.

• **IoT소프트웨어 (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.

Based on the understanding of basic subjects in computer engineering, students will experience developing application programs in embedded systems. Using the Intel PXA255-based Embedded System Toolkit, we will build a cross-compilation development environment, develop Embedded Linux kernel porting and various device controls to develop embedded systems, and then team up to develop creative applications.

• **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.

[전공 선택 - 게임콘텐츠 트랙 - 트랙 필수]

• **스토리텔링 (Storytelling)**

애니메이션과 미디어 등 각종 동적인 시각적 정보의 근본이 되는 작업의 특성을 이해하고 창의적 아이디어를 표현 할 수 있는 시나리오 및 콘티를 제작할 수 있는 능력을 키워준다.

Class presents ability of understanding text material work (scenario), and successful way to transfer into story-board format work in order for best communication between planning group and production group, in the fields of animation and media.

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

텍스트 기반 사용자 인터페이스(CLI: Command Line Interface)를 활용한 기초적인 게임 프로그램을 직접 개발하고, 개인의 포트폴리오로 관리하는 방법을 학습한다. 이를 위하여, 오픈소스로 공개된 고전적인 게임에 대한 코드 레벨의 이해를 수행한 후, 직접 게임을 설계 및 개발하는 과정을 통하여 게임 개발에 대한 개발 프로세스를 이해한다.

Develop a basic game program utilizing the CLI (Command Line Interface). Perform an understanding of the classic game code released as open source. And through a process of direct design and develop games understand the development process for the game development.

• **게임 UI/UX 디자인 (Game UI/UX Design)**

디지털 게임에 필요한 다양한 디자인 요소와 유저 중심의 다양한 경험 설계에 대해 이해하고 이를 구현하는 능력을 배양한다.

Understand the different experiences of various design elements and user-centered design required for the digital game and their ability to implement them.

• **게임학 (Ludology)**

본 수업은 게임미디어를 통하여 다양한 디지털미디어 기반이론들에 대해 고찰하고 기초적인 게임과 스마트 인터렉티브 콘텐츠의 기획 및 제작을 위한 가장 근본적인 이론교육인 게임이론, 게임역사, 게임분석을 교육하고, 융합적 논리로 디지털 미디어를 이해하는 것에 목표를 둔다.

Understanding these fundamental theories is vital for video game research and development. This course will be focused on basic theories such as history of video game, interaction theory and play theory that can form the basis for video game field. Based on theoretical approach to the video game area, current trends and future directions of video game can be considered. course will provides you practice on the process and the expression for the most creative game software producing.

• **게임그래픽프로그래밍 (Game Graphic Programming)**

디지털 게임 제작에 있어서 가장 기본이 되는 그래픽 게임 개발 능력을 학습한다. 특히 그래픽 관련 선형대수(좌표변환 등), 게임에 사용되는 물리학(이동, 충돌) 등의 이론적 지식을 OpenGL/DirectX 등의 실습을 통해서 실현한다.

Students will learn the basic skills of graphic game development in digital game production. In particular, students will realize the theoretical knowledge on graphics related linear algebra (coordinate transformation etc.) and game related physics (movement, collision etc.) through practicing OpenGL/DirectX.

• **게임엔진프로그래밍 (Game Engine Programming)**

게임 엔진에 대한 이해와 실습을 수행한다. 먼저 상용 엔진인 유니티에 대한 이해와 실습을 수행한다. 그리고, 오픈소스 소프트웨어 엔진인 Steam을 이해하고, 보다 창의적인 게임 개발을 위한 아이디어를 제안하며, 이를 실제 엔진에 구현해 본다.

Students will learn and practice game engines. First, students will learn and practice Unity, which is a commercial game engine. Then, students will learn Steam, which is an open source software engine. Students will propose ideas to develop creative games, and implement these games using game engines.

• **게임디자인 (Game Design)**

게임 디자인 요소 등에 대한 기본적인 이론 강의부터 미래 게임을 위한 다양한 분야의 융합적인 아이디어 도출에 대한 고찰 그리고 게임의 순기능에 기반 한 기능성 게임 및 그 확장성을 넓히고 있는 동작기반 게임을 포함한 다양한 게임에 대한 이해를 바탕으로 캡스톤디자인 프로세스를 통해 학생들이 게임 시나리오 및 직접 게임을 기획하고 프로토타입을 제작하게 하여 디지털 콘텐츠의 주요 장르 중 하나인 게임의 구성적 요소에 대해 살펴 볼 수 있는 기회를 마련한다.

From basic theory lesson like the game design elements and a fusion of ideas drawn from various fields for future games. Learn the functional game corresponding to the game sungineung. It performs an understanding of various operations based game, including games. Capstone students through the design process to the production planning and prototyping a game scenario and the game will provide a direct opportunity to examine the constitutive elements of one of the main genres of digital content game.

[전공 선택 - 게임콘텐츠 트랙 - 트랙 선택]

• **3D 모델링 (3D Modeling)**

3차원 그래픽스 전반에 대한 개념과 기술의 이해를 바탕으로 폴리곤 모델링과 맵핑에서부터 조명, 렌더링의 요소에 이르기까지 풍부한 3차원 디지털 오브젝트들을 만들고 표현하는 능력을 갖추도록 한다.

Based on, understanding of 3D Graphic's technology and concept, the purpose of this course aims creating of 3 dimensional digital objects from polygon modeling and mapping to lighting to rendering.

• **인터랙션디자인 (Interaction Design)**

인터랙티브 미디어 디자인의 심화된 수업으로 인터랙션 디자인의 기획과 제작에 대한 일련의 과정을 보다 체계적으로 훈련하여 구체적인 문제해결 능력과 제작 능력을 배양한다.

Developing the production capability and the ability to solve problems specific to production planning and interaction design with an advanced knowledge of design in interactive media.

• **사운드디자인 (Sound Design)**

디지털콘텐츠를 위한 미디와 음향을 설계하고 제작하는 기법을 배운다.

This course provides the techniques of midi planing and sound making for your visual art work.

• **3D 애니메이션 (3D Animation)**

3차원 디지털 어플리케이션을 도구로 사용하여, 특수효과와 모션그래픽을 3차원 애니메이션으로 제작할 수 있는 능력을 갖추도록 한다.

Using a 3D digital application as a tool, This course will provide you the ability to create the FX and the motion graphics in your 3D animation work.

• **체감형기술 이론 및 실습 (Haptics and HCI Programming)**

Haptics와 HCI(Human Computer Interaction)에 대한 이론과 실습을 수행한다. 이를 통해서 다양한 센서에 대해 이해하고 이를 활용하는 피지컬 인터랙션과 동작기반게임에 대해 이해한다.

Perform the theory and practice of the Haptics and HCI (Human Computer Interaction). Understand this through a variety of sensors for an understanding of the physical interaction and motion-based games that take advantage of it.

• **가상/증강 현실 이론 및 실습 (Virtual and Augmented Reality Programming)**

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.

• **모션그래픽스 (Motion Graphics)**

움직임이 있는 그래픽은 TV 뿐만 아니라 웹, 모바일 등 다양한 미디어에서 활용되고 있다. 따라서 본 수업에서는 디자인에 시간의 개념을 더한 움직임이 있는 그래픽에 대하여 그 개념과 활용방안을 이해하고, 방송용, 광고홍보용 등 다양한 종류의 모션 그래픽 실습을 통해 창의적인 제작 능력을 배양한다.

Graphics with the movement, are utilized in a variety of media as well as TV, Web, and mobile. Therefore, in this class the goal is to understand how to take advantage of graphics with movement plus the concept of time to design. This class will also explore creativity through motion graphics the way to practice various types of broadcast, and advertising promotional and foster the production capability.

[전공 선택 - 공통 선택]

• **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.

• **시스템분석 및 설계 (System 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

• **컴퓨터네트워크 (Computer Networks)**

컴퓨터 네트워크를 구성하는 각종 네트워킹 장치들의 계층 모델, 특성, 동작 방법, 그리고 운용 기술에 대하여 학습한다. 또한 이들 장치를 상호 연결한 인터넷의 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 hierarchical models, characteristics, operating methods, and operating techniques of various networking devices that make up a computer network. In addition, we introduce the configuration and operation method of internet connecting these devices. This course introduces the structure and operation of computer network. Through this course, students will be able to understand the seven-layer structure of the computer network and the four-layer structure of the Internet, design a simple LAN (Local Area Network), and develop the theoretical understanding and analysis ability of the computer network based on the layer model You can learn the theory and practical techniques of computer networking concepts.

• **컴퓨터그래픽스 (Interactive Computer Graphics)**

2D와 3D 객체의 생성과 디스플레이를 위한 기본적인 기술들을 소개한다. 주요 강의 내용은 그래픽스를 위한 자료구조, 그래픽 프로그래밍 언어, 기학학적 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.

• **SW스타트업비지니스 (Software Start-Up Business)**

실리콘밸리의 수많은 성공한 창업가들이 컴퓨터공학 출신이다. 이미 컴퓨터공학은 창업가들로 하여금 가장 핵심적이고 중요한 기술로 부각되고 있다. 구글, 마이크로소프트, 샤오미 들에서 제시하는 차세대 기술에 대해 연구하고, 이러한 기술을 기반으로 하는 모의 창업을 통해서, 졸업후 사업자 역량을 배운다.

Many successful entrepreneurs in Silicon Valley are from computer engineering. Computer technology has already become the most important and important technology for entrepreneurs. We study the next generation technologies presented by Google, Microsoft and Xiaomi, and learn business capabilities after graduation through mock-ups based on these technologies.

• **Java프로그래밍 (Java Programming)**

자바는 인터넷의 대중화와 더불어 가장 강력한 객체지향 프로그래밍 언어로 자리 잡고 있다. 자바는 현재 엔터프라이즈 솔루션의 핵심적인 웹 애플리케이션 서버의 책임 언어이며, 인터넷 분야뿐만 아니라 네트워크, 멀티미디어, 그래픽스, 임베디드 시스템까지 광범위하게 응용되고 있다. 본 과목에서는 자바 프로그래밍의 기본 개념을 배우고 여러 응용 프로그래밍을 구현함으로서 실제 업무에 적용 가능한 실무능력을 키운다.

With popularization of Internet, Java becomes one of the most powerful object-oriented programming languages. Now Java is a core language of web application servers, which plays a key role in enterprise solutions. Besides Internet, Java is being used in various areas such as network, multimedia, graphics, embedded systems, etc. This course studies the basic concepts of Java programming. By practicing implementing diverse application programs, it also helps

students improve practical skills that can be used in real environments.

• **인공지능 (Artificial Intelligence)**

매칭, 제한조건의 이용, 탐색, 문제해결, 논리제어 등과 같은 개념을 익힘으로서 복잡도가 높은 문제들에 대한 해답을 찾는 방법을 다룬다. 지능적 탐색, 논리 증명, 지식을 학습하는 기계학습과 지능에이전트의 원리를 다룬다.

This course treats various problem solving methods and knowledge representations to solve highly complicated problems. Intelligent search, logic and proof, machine learning, intelligent agent are major topics.

• **머신러닝 (Machine Learning)**

머신러닝은 데이터에 숨겨져 있는 규칙을 찾아내는 방법을 연구하는 분야이다. 본 과목에서는 머신러닝의 기본 개념들을 소개하고, 최근 각광받고 있는 딥러닝 모델을 TensorFlow로 구현해서 이를 실제 문제에 응용해 본다.

Machine learning is the research area that tries to identify hidden rules from data sets. This course introduces the basic concepts of machine learning and how to implement deep learning methods using TensorFlow for benchmark problems.

• **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.

• **독립심화학습 1/2(소프트웨어융합학과) (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 소프트웨어융합학과 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 software convergence 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 소프트웨어융합학과 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 software convergence 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.

• **융합연구 1/2/3/4/5/6 (Convergence Research 1/2/3/4/5/6)**

전공과목멘토링, 외국인학생프렌드십, 프로그래밍재능기부, 오픈소스기여, 프로그램앱스토어등록 등을 설계 및 진행하거나 소프트웨어공모전, 창업공모전, 프로그래밍경진대회, 학술공모전 등을 참가 및 출품

Students design his/her own course about Subject Mentoring, Foreign Student Friendhood, Programming Talent Donation, Open Source Contribution, Program AppStore Registration, Software Competition, Entrepreneurial Competition, Programming Competition, Academic Competition.

- 전공 과목 멘토링 (Subject Mentoring): 학생이 이미 수강한 과목을 듣는 후배를 멘토링한다. 멘토를 신청하는 학생은 해당 과목에 대해서 A- 이상의 학점을 받아야 한다. 멘토는 멘토링에 대한 계획서와 실적서를 제출하고, 실질적인 성적 향상 여부에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.

Mentor the junior who listens to the class that the student has already taken. Students who apply for a mentor must receive an A- or higher credit for the subject. The mentor submits plans and achievements for mentoring, and receives appropriate credits from P/F from the advisor based on actual improvement in grades.

- 외국인 학생 프렌드십 (Foreign Student Friendhood): 학생은 소프트웨어융합학과 소속의 외국인 학생의 학업/프로그래밍기술 학습/전공수업적응에 대한 기여를 하거나 멘토링을 수행한다. 멘토는 멘토링에 대한 계획서와 실적서를 제출하고, 실질적인 외국인 학생에 대한 기여 여부에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.

The student contributes to the adaptation of the academic / programming skill learning / major instruction of the foreign student belonging to the Software Convergence Department and performs the mentoring. The mentor submits plans and achievements for mentoring and receives appropriate credits from P/F from the advisor based on whether he or she actually contributes to the foreign student.

- 프로그래밍 재능기부 (Programming Talent Donation): 학생은 본인이 확보한 소프트웨어 수요처 혹은 학교에서 제공하는 소프트웨어 수요처의 문제를 해결하는 소프트웨어를 개발하여 주는 방법으로 재능기부를 수행한다. 지역사회 초중고등학교 학생에 대한 프로그래밍 교육도 고려할 수 있다. 신청하는 학생은 계획서와 실적서를 제출하고, 실질적인 기여에 기반하여 지도교수로 부터 P/F 중 적합한 학점을 부여 받는다.

A student donates talent in a way that develops software that solves the problem of the software demanded by the user or the demand of the software provided by the school. Programming education for elementary, middle and high school students in the community can also be considered. Applicants are required to submit a proposal and an achievement form, and receive appropriate credits from P/F from their advisor on a practical contribution basis.

- 오픈소스 기여 (Open Source Contribution): 학생은 지도교수의 지도하에, 본인이 희망하거나 지도교수가 권장하는 오픈소스 커뮤니티에 소프트웨어 코드를 기여하도록 한다. 오픈소스 소프트웨어에 코드를 기여하기를 희망하는 학생은 계획서와 실적서를 제출하고, 실질적인 기여 여부 등에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.

Under the guidance of your supervisor, the student will be asked to contribute the software code to the open source community that he or she would like or recommended by the professor. A student who wishes to contribute code to open source software must submit a proposal and an achievement report and be eligible for an P/F credit from an advisor based on actual contribution.

- 프로그램 앱스토어 등록 (Program AppStore Registration): 학생은 지도교수의 지도하에, 본인이 만든 소프트웨어를 앱스토어를 통하여 판매하는 과정을 거치도록 한다. 소프트웨어 판매를 희망하는 학생은 계획서와 실적서를 제출하고, 실질적인 앱스토어 등록 여부 및 결과에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.

Under the supervision of the supervisor, the student goes through the process of selling the software he created through the App Store. Students who wish to sell software are required to submit a proposal and an achievement

book, and receive appropriate credits from P/F from their advisor based on the actual App Store registration and results.

- **소프트웨어 공모전 (Software Competition):** 학생은 지도교수가 제시하는 공모전들 혹은 본인이 희망하는 공모전에 응모하는 과정을 통하여 실질적인 포트폴리오를 확보하도록 한다. 공모전을 신청하는 학생은 계획서와 실적서를 제출하고, 실질적인 공모 여부 및 결과에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.
The student will have to obtain a substantial portfolio through the process of applying for the contest presented by the advisor or his / her desired competition. Students who apply for the competition must submit a proposal and an achievement book, and receive appropriate credits from the advisor based on whether they are actually participating in the competition or not.

- **창업 공모전 (Entrepreneurial Competition):** 학생은 지도교수가 제시하는 공모전들 혹은 본인이 희망하는 공모전에 응모하는 과정을 통하여 실질적인 포트폴리오를 확보하도록 한다. 공모전을 신청하는 학생은 계획서와 실적서를 제출하고, 실질적인 공모 여부 및 결과에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.
The student will have to obtain a substantial portfolio through the process of applying for the contest presented by the advisor or his / her desired competition. Students who apply for the competition must submit a proposal and an achievement book, and receive appropriate credits from the advisor based on whether they are actually participating in the competition or not.

- **프로그래밍 경진대회 (Programming Competition):** 학생은 지도교수가 제시하는 경진대회들 혹은 본인이 희망하는 경진대회에 참여하는 과정을 통하여 본인의 실력을 향상하는 기회를 갖도록 한다. 경진대회에 참가하는 학생은 계획서와 실적서를 제출하고, 실질적인 참여 여부 및 결과에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.
The student will have the opportunity to improve his / her ability through the competitions presented by the advisor or through the process of participating in the contest that he / she wants. Students participating in the competition will submit a proposal and an achievement report, and will be given appropriate credits from P/F based on their actual participation and results.

- **학술 공모전 (Academic Competition):** 학생은 지도교수가 제시하는 공모전들 혹은 본인이 희망하는 공모전에 논문을 응모하는 과정을 통하여 실질적인 포트폴리오를 확보하도록 한다. 공모전을 신청하는 학생은 계획서와 실적서를 제출하고, 실질적인 공모 여부 및 결과에 기반하여 지도교수로부터 P/F 중 적합한 학점을 부여 받는다.
The student has to obtain a substantial portfolio through the process of submitting the papers to the contest presented by the advisor or the contest he / she wants. Students who apply for the competition must submit a proposal and an achievement book, and receive appropriate credits from the advisor based on whether they are actually participating in the competition or not.

• **현장실습/장기현장실습 (Internship in Software Convergence)**
관련 기업에서 실무 경험을 통해 전공지식을 응용한다. (활동기간에 따라 산학필수 3학점, 6학점, 12학점)
This course gives a chance to apply theoretical knowledges in a field.