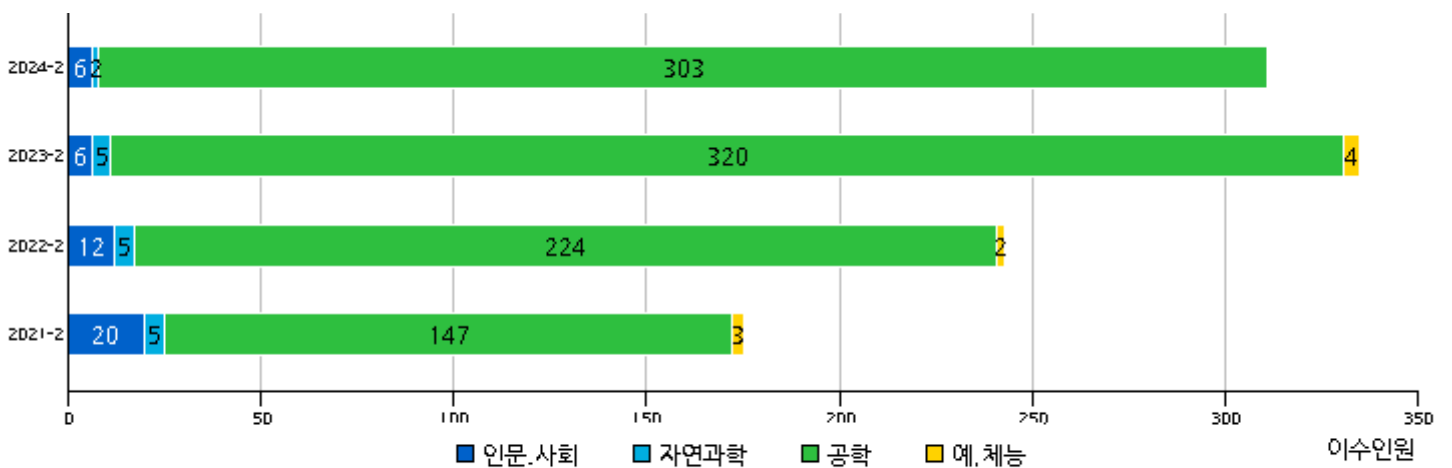
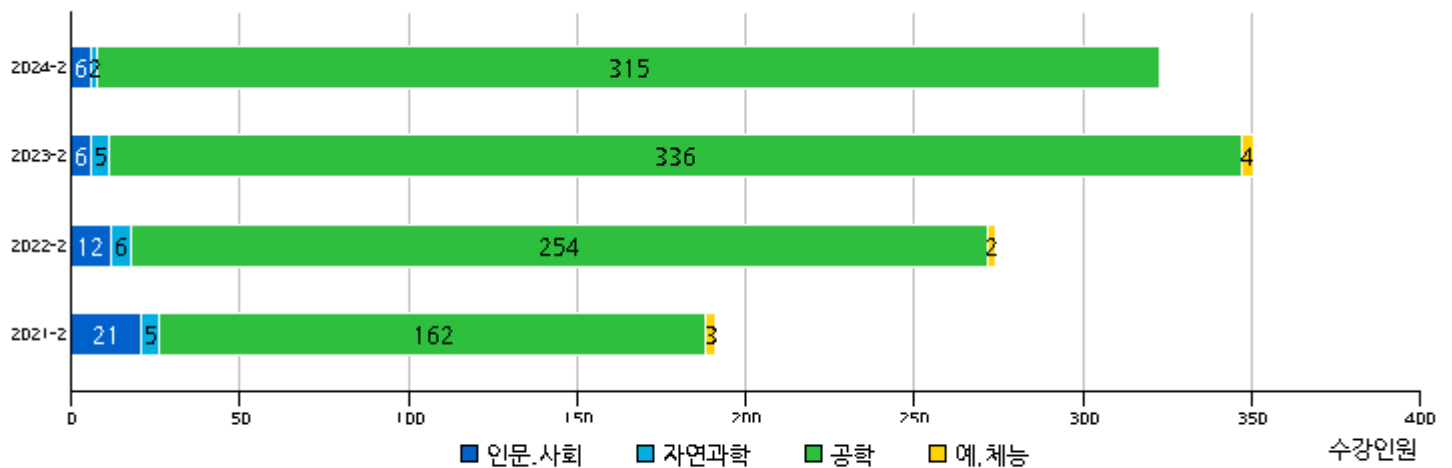
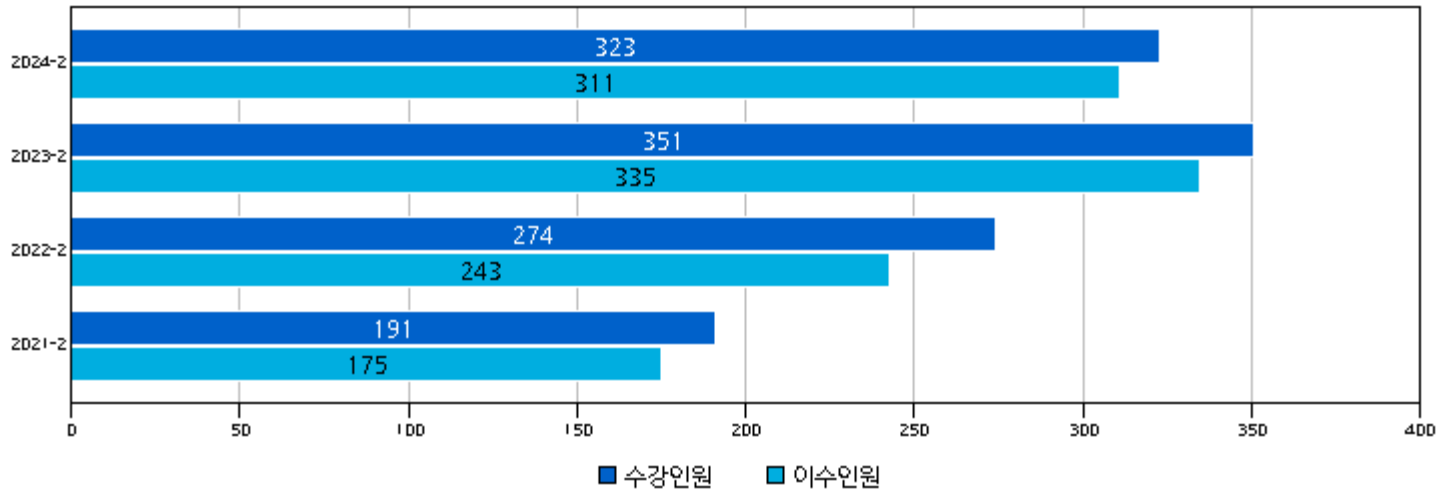


교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

1. 교과목 수강인원

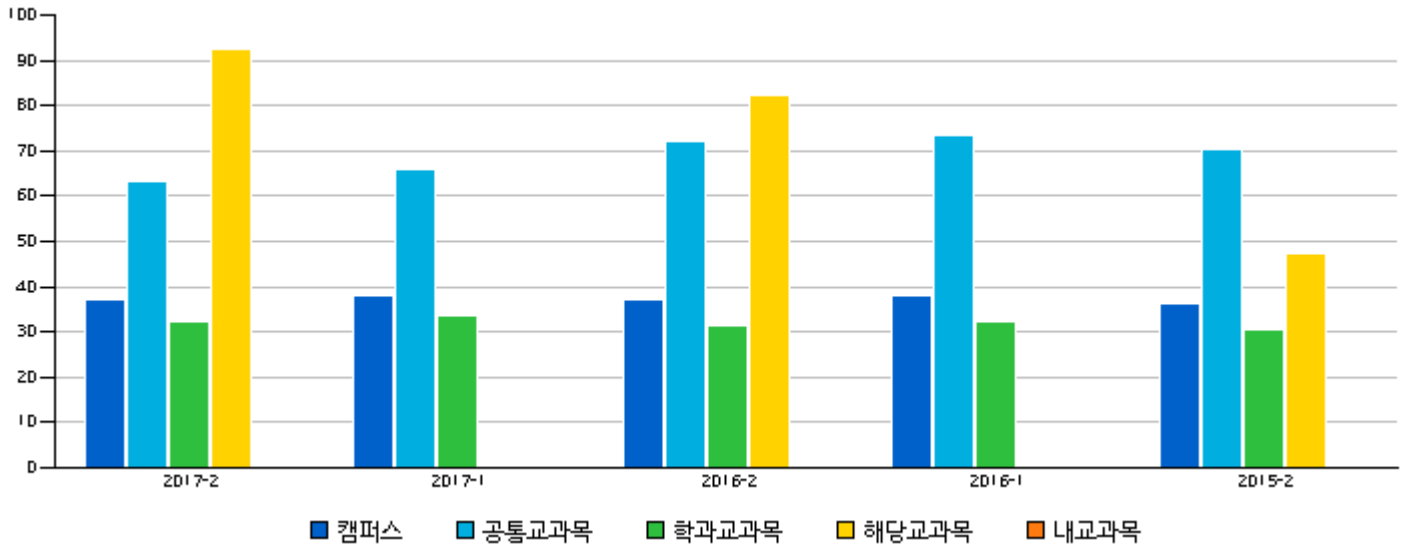


교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

수업년도	수업학기	계열구분	수강인원	이수인원
2021	2	인문.사회	21	20
2021	2	자연과학	5	5
2021	2	공학	162	147
2021	2	예,체능	3	3
2022	2	인문.사회	12	12
2022	2	자연과학	6	5
2022	2	공학	254	224
2022	2	예,체능	2	2
2023	2	인문.사회	6	6
2023	2	자연과학	5	5
2023	2	공학	336	320
2023	2	예,체능	4	4
2024	2	인문.사회	6	6
2024	2	자연과학	2	2
2024	2	공학	315	303

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

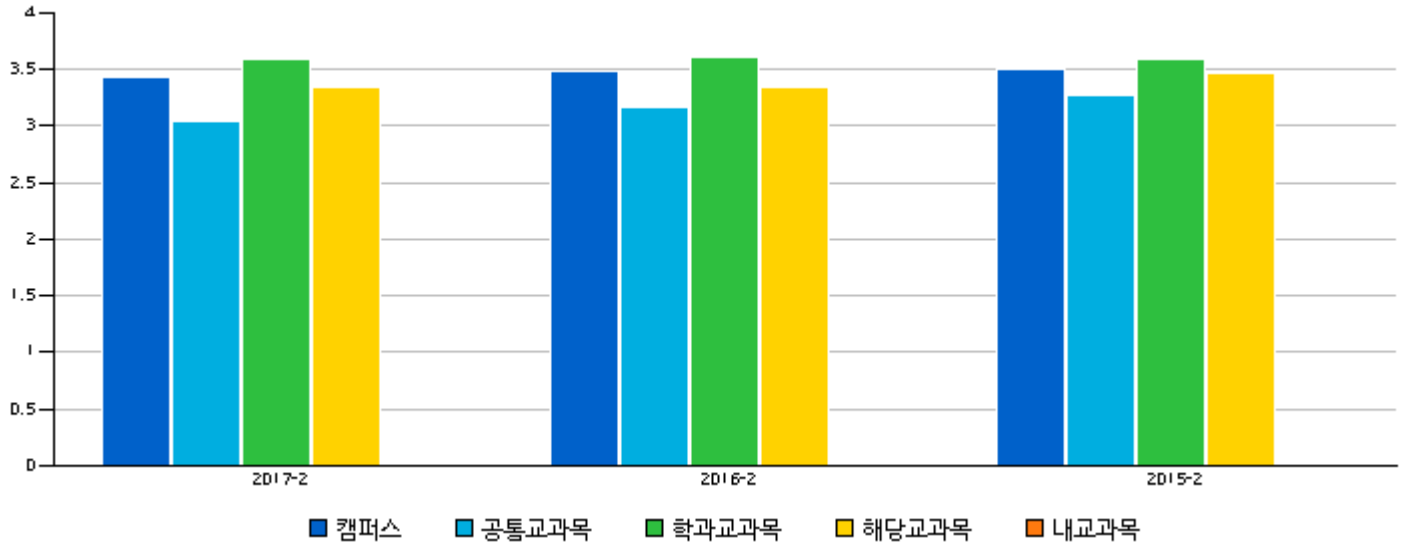
2. 평균 수강인원



수업년도	수업학기	캠퍼스	공통교과목	학과교과목	해당교과목	내교과목
2017	2	37.26	63.09	32.32	92.5	
2017	1	38.26	65.82	33.5		
2016	2	37.24	72.07	31.53	82.5	
2016	1	37.88	73.25	32.17		
2015	2	36.28	70.35	30.36	47.33	

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

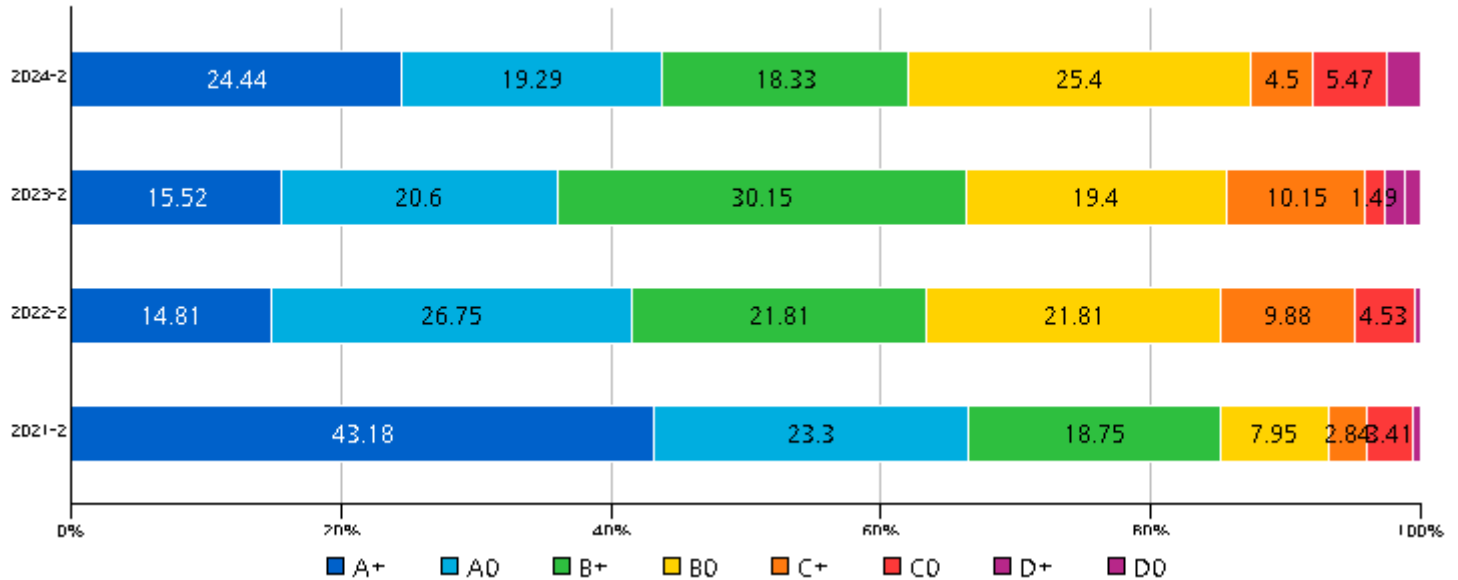
3. 성적부여현황(평점)



수업년도	수업학기	캠퍼스	공통교과목	학과교과목	해당교과목	내교과목
2017	2	3.44	3.05	3.59	3.35	
2016	2	3.49	3.16	3.61	3.35	
2015	2	3.51	3.28	3.6	3.47	

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

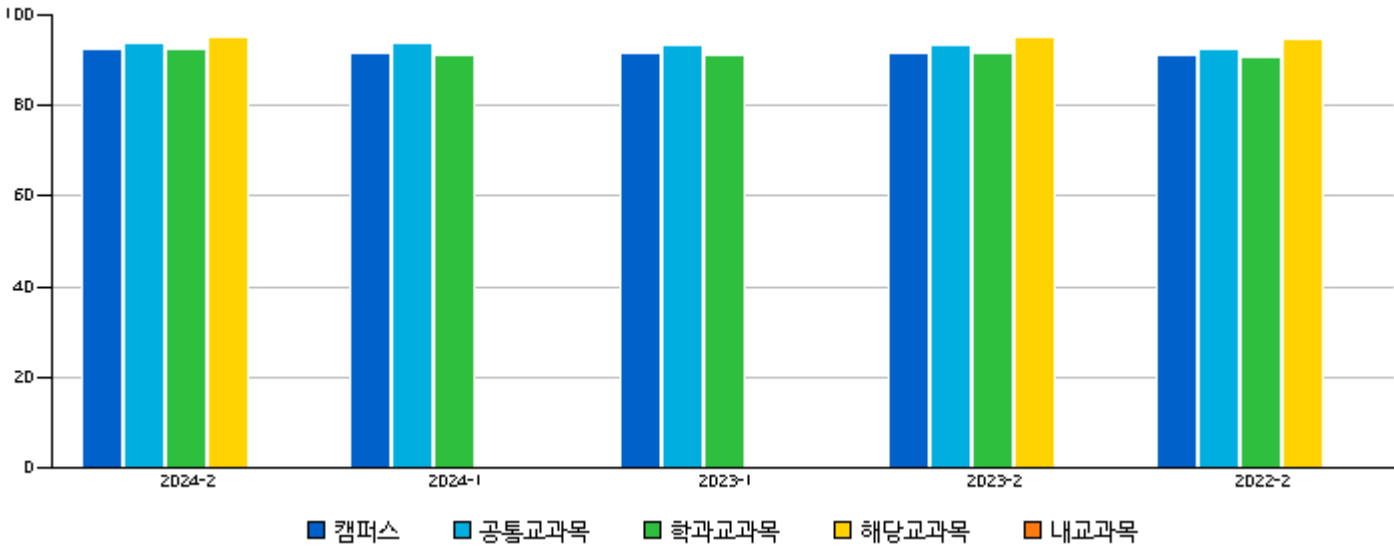
4. 성적부여현황(등급)



수업년도	수업학기	등급	인원	비율	수업년도	수업학기	등급	인원	비율
2021	2	A+	76	43.18	2023	2	D+	5	1.49
2021	2	A0	41	23.3	2023	2	D0	4	1.19
2021	2	B+	33	18.75	2024	2	A+	76	24.44
2021	2	B0	14	7.95	2024	2	A0	60	19.29
2021	2	C+	5	2.84	2024	2	B+	57	18.33
2021	2	C0	6	3.41	2024	2	B0	79	25.4
2021	2	D+	1	0.57	2024	2	C+	14	4.5
2022	2	A+	36	14.81	2024	2	C0	17	5.47
2022	2	A0	65	26.75	2024	2	D+	8	2.57
2022	2	B+	53	21.81					
2022	2	B0	53	21.81					
2022	2	C+	24	9.88					
2022	2	C0	11	4.53					
2022	2	D+	1	0.41					
2023	2	A+	52	15.52					
2023	2	A0	69	20.6					
2023	2	B+	101	30.15					
2023	2	B0	65	19.4					
2023	2	C+	34	10.15					
2023	2	C0	5	1.49					

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

5. 강의평가점수



수업년도	수업학기	캠퍼스	공통교과목	학과교과목	해당교과목	내교과목
2024	2	92.56	93.8	92.33	95	
2024	1	91.5	93.79	91.1		
2023	1	91.47	93.45	91.13		
2023	2	91.8	93.15	91.56	95.33	
2022	2	90.98	92.48	90.7	94.75	

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

6. 강의평가 문항별 현황

번호	평가문항	본인 평 균 (가 중 치 적 용)	소속학과, 대학평균과의 차이 (+초과, -:미달)		점수별 인원분포						
					매우 그 렇 지 않 다	그 렇 지 않 다	보 통 이 다	그 렇 다	매우 그 렇 다		
		5점 미만	학과		대학		1점	2점	3점	4점	5점
교강사:			차이	평균	차이	평균					
No data have been found.											

7. 개설학과 현황

학과	2025/2	2024/2	2023/2	2022/2	2021/2
예술융합소프트웨어 융합전공	1강좌(3학점)	0강좌(0학점)	0강좌(0학점)	0강좌(0학점)	0강좌(0학점)
컴퓨터소프트웨어학부	2강좌(6학점)	2강좌(6학점)	2강좌(6학점)	3강좌(9학점)	2강좌(6학점)
심리뇌과학전공	0강좌(0학점)	0강좌(0학점)	0강좌(0학점)	1강좌(3학점)	0강좌(0학점)
인문소프트웨어융합전공	1강좌(3학점)	0강좌(0학점)	0강좌(0학점)	0강좌(0학점)	0강좌(0학점)
데이터사이언스학부	1강좌(3학점)	1강좌(3학점)	1강좌(3학점)	0강좌(0학점)	0강좌(0학점)

8. 강좌유형별 현황

강좌유형	2021/2	2022/2	2023/2	2024/2	2025/2
일반	2강좌(192)	4강좌(274)	3강좌(351)	3강좌(323)	0강좌(0)

9. 교과목개요

교육과정	관장학과	국문개요	영문개요	수업목표
학부 2024 - 2027 교육과정	서울 공과대학 컴퓨터공학부 컴퓨터전공	일반적으로 널리 알려진 알고리즘의 예 및 대표적인 응용 분야에 적용되는 알고리즘의 설계기법 및 분석과정을 배우며 알고리즘의 복잡도를 분석하고 효율적인 알고리즘을 설계하는 기법을 익힌다.	This course introduces the basic techniques for designing and analyzing algorithms. Topics include the techniques for analyzing the time and space requirements of algorithms, and programming methodologies such as dynamic programming, divide and conquer, balancing. Other topics include a selection of applications such as disjoint set union/find, graph algorithms, search trees, pattern matching, and include the introduction to the polynomial complexity	

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

교육과정	관장학과	국문개요	영문개요	수업목표
학부 2024 - 2027 교육과정	서울 공과대학 컴퓨터소프트웨어학부	소프트웨어의 개발에 있어 원천이 되는 핵심 알고리즘들을 살펴본다. 다양한 응용에서 활용되는 정렬 알고리즘, 그래프 알고리즘을 학습하고, 동적 프로그래밍이나 greedy algorithm 과 같은 기법에 대해서도 예제를 통해 익힌다. 동일한 문제를 해결하는 다양한 알고리즘에 활용되는 자료구조를 기반으로 계산복잡도를 확인하고, 이를 바탕으로 실제 응용문제 해결에 있어 적합한 알고리즘을 선택할 수 있다.	classes P, NP, and co-NP; intractable problems. This course provides an introduction to mathematical modeling of computational problems. It covers the common algorithms, algorithmic paradigms, and data structures used to solve these problems. The course emphasizes the relationship between algorithms and programming, and introduces basic performance measures and analysis techniques for these problems.	다양한 응용에서 활용되는 핵심 원천 알고리즘의 기본 원리와 계산 복잡도를 이해하고 더 나아가 이를 활용하여 실제적인 문제 해결에 적용할 수 있도록 한다. This course introduces students to the analysis and design of computer algorithms. Upon completion of this course, students will be able to do the following: - Analyze the asymptotic performance of algorithms. - Demonstrate a familiarity with major algorithms and data structures. - Apply important algorithmic design paradigms and methods of analysis. - Synthesize efficient algorithms in common engineering design situations.
학부 2020 - 2023 교육과정	서울 공과대학 컴퓨터공학부 컴퓨터전공	일반적으로 널리 알려진 알고리즘의 예 및 대표적인 응용 분야에 적용되는 알고리즘의 설계기법 및 분석과정을 배우며 알고리즘의 복잡도를 분석하고 효율적인 알고리즘을 설계하는 기법을 익힌다.	This course introduces the basic techniques for designing and analyzing algorithms. Topics include the techniques for analyzing the time and space requirements of algorithms, and programming methodologies such as dynamic programming, divide and	

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

교육과정	관장학과	국문개요	영문개요	수업목표
			conquer, balancing. Other topics include a selection of applications such as disjoint set union/find, graph algorithms, search trees, pattern matching, and include the introduction to the polynomial complexity classes P, NP, and co-NP; intractable problems.	
학부 2020 - 2023 교육과정	서울 공과대학 컴퓨터소프트웨어학부	<p>소프트웨어의 개발에 있어 원천이 되는 핵심 알고리즘들을 살펴본다. 다양한 응용에서 활용되는 정렬 알고리즘, 그래프 알고리즘을 학습하고, 동적 프로그래밍이나 greedy algorithm 과 같은 기법에 대해서도 예제를 통해 익힌다. 동일한 문제를 해결하는 다양한 알고리즘에 활용되는 자료구조를 기반으로 계산복잡도를 확인하고, 이를 바탕으로 실제 응용문제 해결에 있어 적합한 알고리즘을 선택할 수 있다.</p>	<p>This course provides an introduction to mathematical modeling of computational problems. It covers the common algorithms, algorithmic paradigms, and data structures used to solve these problems. The course emphasizes the relationship between algorithms and programming, and introduces basic performance measures and analysis techniques for these problems.</p>	<p>다양한 응용에서 활용되는 핵심 원천 알고리즘의 기본 원리와 계산 복잡도를 이해하고 더 나아가 이를 활용하여 실제적인 문제 해결에 적용할 수 있도록 한다.</p> <p>This course introduces students to the analysis and design of computer algorithms. Upon completion of this course, students will be able to do the following:</p> <ul style="list-style-type: none"> - Analyze the asymptotic performance of algorithms. - Demonstrate a familiarity with major algorithms and data structures. - Apply important algorithmic design paradigms and methods of analysis. - Synthesize efficient algorithms in common engineering design situations.
학부 2016 - 2019 교육과정	서울 공과대학 컴퓨터공	소프트웨어의 개발에 있어 원천이 되는 핵심 알고리즘들을 살펴본다. 다양한 응용에서 활용되	This course provides an introduction to mathematical modeling of computational	

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

교육과정	관장학과	국문개요	영문개요	수업목표
정	학부 소프트웨어전공	는 정렬 알고리즘, 그래프 알고리즘을 학습하고, 동적 프로그래밍이나 greedy algorithm 과 같은 기법에 대해서도 예제를 통해 익힌다. 동일한 문제를 해결하는 다양한 알고리즘에 활용되는 자료구조를 기반으로 계산복잡도를 확인하고, 이를 바탕으로 실제 응용문제 해결에 있어 적합한 알고리즘을 선택할 수 있다.	problems. It covers the common algorithms, algorithmic paradigms, and data structures used to solve these problems. The course emphasizes the relationship between algorithms and programming, and introduces basic performance measures and analysis techniques for these problems.	
학부 2016 - 2019 교육과정	서울 공과대학 컴퓨터공학부 컴퓨터전공	일반적으로 널리 알려진 알고리즘의 예 및 대표적인 응용 분야에 적용되는 알고리즘의 설계기법 및 분석과정을 배우며 알고리즘의 복잡도를 분석하고 효율적인 알고리즘을 설계하는 기법을 익힌다.	This course introduces the basic techniques for designing and analyzing algorithms. Topics include the techniques for analyzing the time and space requirements of algorithms, and programming methodologies such as dynamic programming, divide and conquer, balancing. Other topics include a selection of applications such as disjoint set union/find, graph algorithms, search trees, pattern matching, and include the introduction to the polynomial complexity classes P, NP, and co-NP; intractable problems.	
학부 2016 - 2019 교육과정	서울 공과대학 컴퓨터소프트웨어학부	소프트웨어의 개발에 있어 원천이 되는 핵심 알고리즘들을 살펴본다. 다양한 응용에서 활용되는 정렬 알고리즘, 그래프 알고리즘을 학습하고, 동적 프로그래밍이나 greedy algorithm 과 같은 기법에 대해서도 예제를 통해 익힌다. 동일한 문제를 해결하는 다양한 알고리즘에 활용되는 자료구조를 기반으로 계산복잡도를 확인하고, 이를 바탕으로 실제 응용문제 해결에 있어 적합한 알고리즘을 선택할 수 있다.	This course provides an introduction to mathematical modeling of computational problems. It covers the common algorithms, algorithmic paradigms, and data structures used to solve these problems. The course emphasizes the relationship between algorithms and programming, and introduces basic performance measures and analysis techniques for these problems.	<p>다양한 응용에서 활용되는 핵심 원천 알고리즘의 기본 원리와 계산 복잡도를 이해하고 더 나아가 이를 활용하여 실제적인 문제 해결에 적용할 수 있도록 한다.</p> <p>This course introduces students to the analysis and design of computer algorithms. Upon completion of this course, students will be able to do the following:</p> <ul style="list-style-type: none"> - Analyze the asymptotic performance of algorithms. - Demonstrate a familiarity with major algorithms and data structures.

교과목 포트폴리오 (ITE2039 알고리즘및문제해결기법)

교육과정	관장학과	국문개요	영문개요	수업목표
				<ul style="list-style-type: none"> - Apply important algorithmic design paradigms and methods of analysis. - Synthesize efficient algorithms in common engineering design situations.
학부 2013 - 2015 교육과정	서울 공과대학 컴퓨터공학부 컴퓨터전공	일반적으로 널리 알려진 알고리즘의 예 및 대표적인 응용 분야에 적용되는 알고리즘의 설계기법 및 분석과정을 배우며 알고리즘의 복잡도를 분석하고 효율적인 알고리즘을 설계하는 기법을 익힌다.	This course introduces the basic techniques for designing and analyzing algorithms. Topics include the techniques for analyzing the time and space requirements of algorithms, and programming methodologies such as dynamic programming, divide and conquer, balancing. Other topics include a selection of applications such as disjoint set union/find, graph algorithms, search trees, pattern matching, and include the introduction to the polynomial complexity classes P, NP, and co-NP; intractable problems.	

10. CQI 등록내역

No data have been found.