정보과학3

| [메인 문서](https://docs.google.com/document/u/0/d/1F0OwcWBFX517r8jL3AMEKxwHbz69J6QT-eIlSF7itT8/edit) | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [정보과학2](https://docs.google.com/document/d/15AdhkSiDUprTBnnYPVRh8PAQD0A9JpMvgcTfWb6Y0lM/edit?usp=sharing) | | > | 정보과학3 | | > | [알고리즘](https://docs.google.com/document/d/1PVAkVO21l5gAHZlR-MHv1Sk4aOQbPio_wAydMlvhJH4/edit?usp=sharing)  [이산구조](https://docs.google.com/document/d/1oiPmYuU9pyva070_ic_jCYa73jv3XpZpyruTMRjJ70g/edit?usp=sharing)  [정보과학세미나](https://docs.google.com/document/d/1RgBPfTw_vcfDMrV4Ezl9PsP4TsZYu5KdIs6Ax25dOss/edit?usp=sharing)  [정보과학특강](https://docs.google.com/document/d/1VYa_tymRA92ImyEeX2Zs0kdwGVzWLvho3LBqcimiBQM/edit?usp=sharing)  [창의적문제해결기법](https://docs.google.com/document/u/0/d/165q0XksmtcTiEkWfrtf59ALGNXuzBFK4zbntHUhtkgw/edit) | |

## 개요

실질적으로 정보과학 이론에 대해 처음으로 다루는 과목.

## 정보

3학점, 정현우 선생님 담당. 정보과학1과 정보과학2가 재밌어서 수강하는 것은 지양하길 바람. (사실 과목명을 “기초전산학”으로 바꾸는 것이 나을듯… 아니면 ‘기초컴퓨터학’) 앞의 경우에는 프로그래밍과 문제 해결을 수강하는 것을 권함.

이 과목을 수강하는 당시에는 뭘 가르치는 것인지 싶겠지만, 나중에 정보과학을 좀 더 공부하게 되면 굉장히 명강의(?) 라는 것을 느낄 수 있다.

1. Data Representation: 컴퓨터 내에 정수와 실수가 어떻게 저장되는지를 배운다~~. 이걸 배우면 0.1 + 0.2가 0.3이 아닌 이유를 유도할 수 있다.~~
2. Toy Programming: 컴퓨터의 메모리 작동 원리를 알 수 있다. 처음에는 상당히 헷갈리지만 익숙하면 쉬워진다. ~~19 다음에 꼭 20 써서 틀리는게 국룰이다.~~
3. Time Analysis: 시간복잡도에 대해 배운다. 알고리즘 문제해결에 항상 따라다니게 될 친구이니 잘 배우도록 하자!
4. Divide and Conquer: 분할정복을 배운다. 이 과정에서 O(n log n) 정렬 알고리즘을 배울 수 있다. 익숙하지 않다면 코드를 이해하는게 힘들 수도 있지만, 코드를 이해하면 왠만하면 중간고사에서 이 문제는 맞는다.
5. Dynamic Programming: 다이나믹 프로그래밍 또는 동적 계획법이다. 알고리즘 문제해결에 거의 필수불가결적인 요소이다.
6. Logic Circuit: AND, OR, NOT, XOR 이 네 요소로 논리 회로를 만든다. 놀랍게도 시험 문제로 낼 부분이 꽤 있는데, XOR Gate를 AND, OR, NOT만 사용해서 나타낸다거나, 비트 3개를 정렬한다거나, 비트 3개를 더한다거나 등등…
7. CYK Algorithm: 정보과학2에서 Stable Matching을 배웠을 것이다. 정보과학3에서 배우는 대표적인 알고리즘을 뽑으라면 이 알고리즘을 뽑을 수 있다.
8. NP-Complete: 당신이 기말고사에서 단 한 문제를 틀린다면, 이 부분에서 틀릴것이다. P 문제와 NP 문제가 무엇인지를 알고, Poly-time reduction 이라는 방법을 사용해서 어떤 문제를 다른 문제로 변환시키는 방법을 배운다. 증명 전개 방식이 상당히 까다로운 경우가 많다.
9. Search Engine: 검색 엔진의 작동 원리에 대해서 매우 간단한 모델부터 현재 구글이 쓰는 Random Surfer Model에 대해서까지 배운다. ~~근데 꼭 시험은 어려운 모델이 나오더라...~~
10. Thread: 사실 이걸 이용해서 풀어야하는 알고리즘 문제면, 매우 어렵다. 하지만 정보과학3에서는 Thread가 무엇이고 어떻게 작동하는지, 그리고 기초적인(?) 코드를 통해서 알 수 있으니 그리 걱정하지 않아도 된다.
11. Network: 시험에 나오지는 않는 파트이다. 그러나 기초전산학이라는 별명에 걸맞는 단원이라 할 수 있다. ~~이것으로 카카오톡 부러운 채팅방을 만들 수 있다!~~

## 과제

프로그래밍이나 문제 풀이가 과제로 나온다. 가끔 증명 문제도 있다. 난이도는 수업 내용을 잘 이해했으면 별로 어렵지 않다.

## 시험

선생님께서 시험 전에 기출문제를 주시니 몇 개만 풀어보면 좋다. 형식은 매번 비슷하다.

중간고사는 때마다 다르지만 보통 어려운 문제는 Toy Programming이랑 Dynamic Programming에서 나온다. 수십 줄이나 되는 toy code를 보면 절망감이 밀려온다. Big O Notation도 함정에 걸리지 않게 조심하자.~~이번 학기에도 O(n^1000) vs O((log n)^(log n))에 낚인 학생들이 있다~~

기말고사는 거의 무조건 NP 증명문제가 하나는 나온다.

## 난이도

강의자료를 꼼꼼히 보고 주어진 과제를 열심히 한다면 A+, A0는 충분히 획득 가능.

자료구조, 프로그래밍과 문제해결에 비해 과목 난이도는 높은 편이나, 개념을 완벽히 이해한다면 시험 문제는 무난히 해결 가능함.

## 팁

1. 강의 자료가 친절하지 않기에 수업에 집중하고 성실히 참여하자.

## 기타

상당히 문맥없는 수업의 연속이다. 전 강의와의 연관성을 찾지 말자.

~~수업 전체가 context-free~~

의외로 중간에 스쳐지나가듯 배우는 defaultdict가 쓸모가 있다. 파이썬을 한다면 지금 구글에 한 번 쳐보자.