

인공지능! 체험과 실습을 통한 이해

건양대학교 박 헌 규 교수

010-5084-8123 / ingenium@konyang.ac.kr

대학연계 참학력 공동교육과정 (23.7.25 ~ 8.1)

수업 일정

일차	날짜	차시	시간	수업내용	비고
1	7. 25 (화)	1~3 (3H)	2:30 ~5:20	<ul style="list-style-type: none"> 오리엔테이션 인공지능의 정의, 역사, 종류 인공지능 체험 1 	구글 계정
2	7. 26 (수)				
3	7. 27 (목)	4~7 (4H)	2:00 ~5:50	<ul style="list-style-type: none"> 인공지능, 머신러닝, 딥러닝 관계 이해 머신러닝의 종류 (지도학습, 비지도학습, 강화학습 사례) 인공지능 체험 2 인공지능 실습환경 구축 (구글 코랩 환경 설정) 	구글 계정
4	7. 28 (금)	8~10 (3H)	2:00 ~4:50	<ul style="list-style-type: none"> 인공지능으로 구현한 틱택토 게임 틱택토 게임으로 보는 인공지능 원리 학습 인공지능으로 구현한 오목 게임 인공지능 오목 게임의 원리 	구글 계정
5	7. 31 (월)	11~14 (4H)	2:00 ~5:50	<ul style="list-style-type: none"> 인공지능 바둑 "알파고" 구현 원리 이해 머신러닝 지도 학습의 종류 (분류, 회귀) 구글 코랩을 이용한 인공지능 지도학습 실습 학습한 모델을 통해 새로운 데이터의 예측 	구글 계정
6	8. 2 (수)	15~17 (3H)	2:00 ~4:50	<ul style="list-style-type: none"> 내가 쓴 손글씨 숫자 인식시키기 (이미지 인식) 구글 코랩을 이용한 MNIST 이미지 인식 실습 이미지를 인식하는 인공지능(CNN) 학습 	구글 계정

4일차

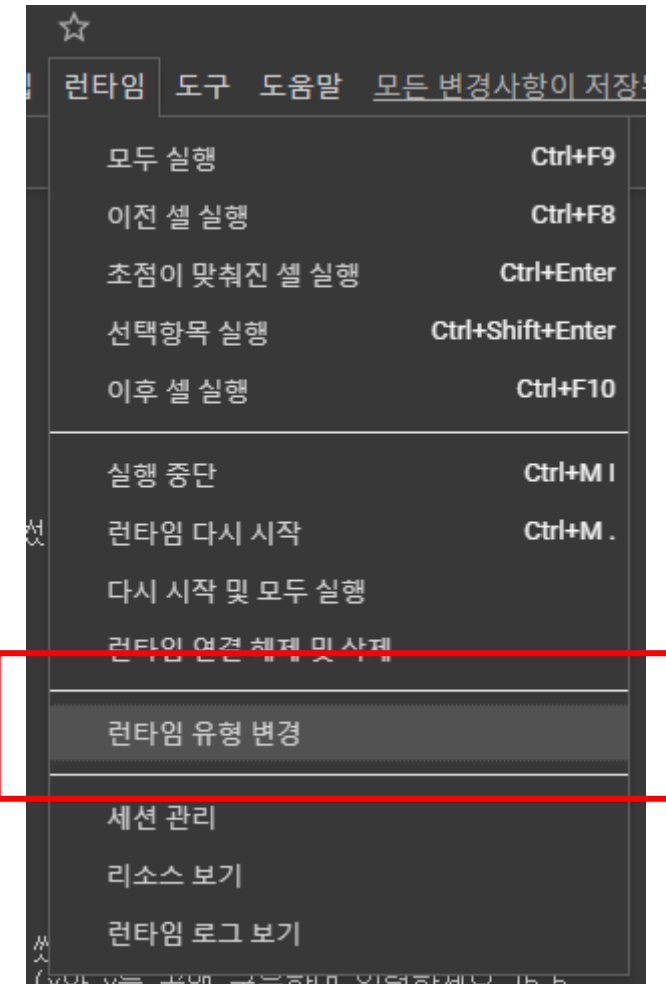
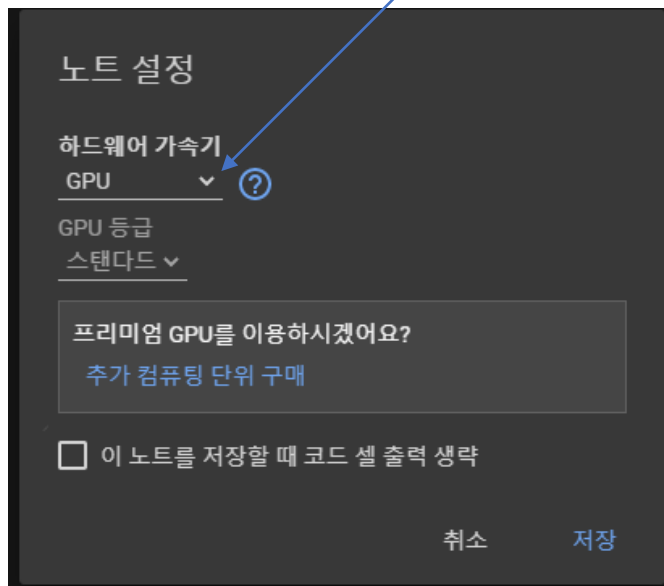
- 인공지능으로 구현한 틱택토 게임
- 틱택토 게임으로 보는 인공지능 원리 학습
- 인공지능으로 구현한 오목 게임
- 인공지능 오목 게임의 원리

4일차 1교시

인공지능으로 구현한 틱택토 게임

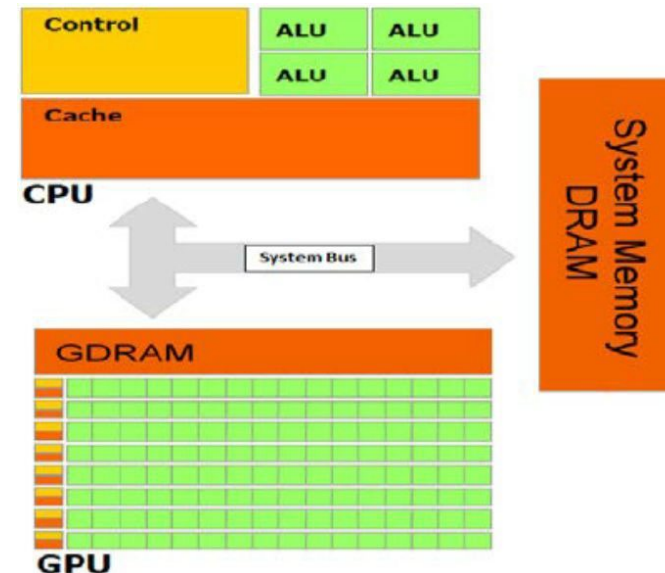
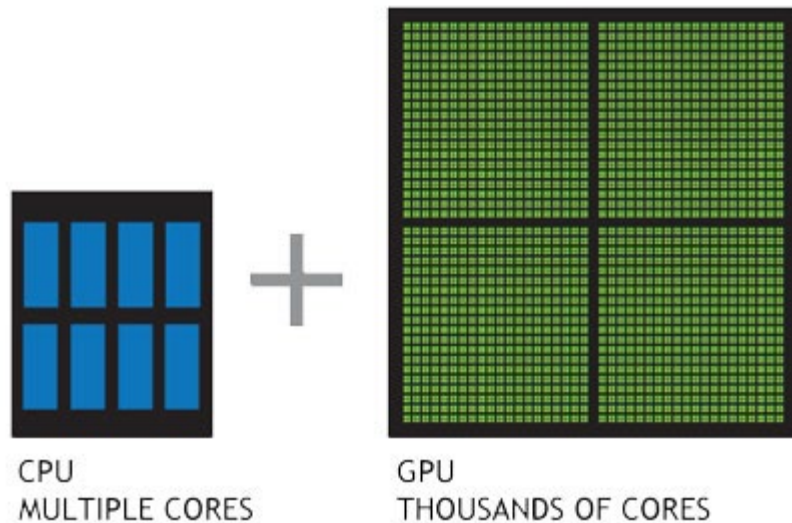
구글 코랩 GPU 활용하기

- GPU : Graphics Processing Unit
 - 더 작고 전문화된 코어로 구성된 프로세스
- 메뉴 - 런타임 - 런타임 유형 변경 - 하드웨어 가속기 - GPU - 저장



CPU vs GPU

- CPU Central Processing Unit : 명령어가 입력되는 순서대로 데이터를 처리
 - 컴퓨터를 지휘하는 역할, 많은 코어를 가지지 못함 (4 ~ 10개)
- GPU Graphic Processing Unit : 여러 명령어를 동시에 처리하는 병렬 처리 방식
 - 단순하지만 동시에 많은 계산에 적합, 수천 개의 코어 가능



틱택토 게임 실습

- 틱택토 게임이란? [틱택토 게임 해보기](#)
 - 두 명이 번갈아가며 O나 X를 3x3판에 두면서, 먼저 가로나 세로, 대각선 한 줄을 만드는 사람이 이김

	0	1	2
a	0 a	1 a	2 a
b	0 b	1 b	2 b
c	0 c	1 c	2 c

0	0	0
	0	X
	X	X

틱택토 게임 실습

- "code2 tictactoe.py" 다운로드
- 메모장에서 읽어서 복사
- 구글 코랩에 붙여넣기
- 실행 버튼 or Ctrl+Enter
- 입력 방법 : x축 y축 (0부터 시작함)

	0	1	2
a	0 a	1 a	2 a
b	0 b	1 b	2 b
c	0 c	1 c	2 c

- 코드 맨 아래에 사람이 먼저 할 것인지, 컴퓨터가 먼저 할 것인지 정할 수 있음

4일차 2교시

인공지능 틱택토 게임 원리

틱택토 게임트리

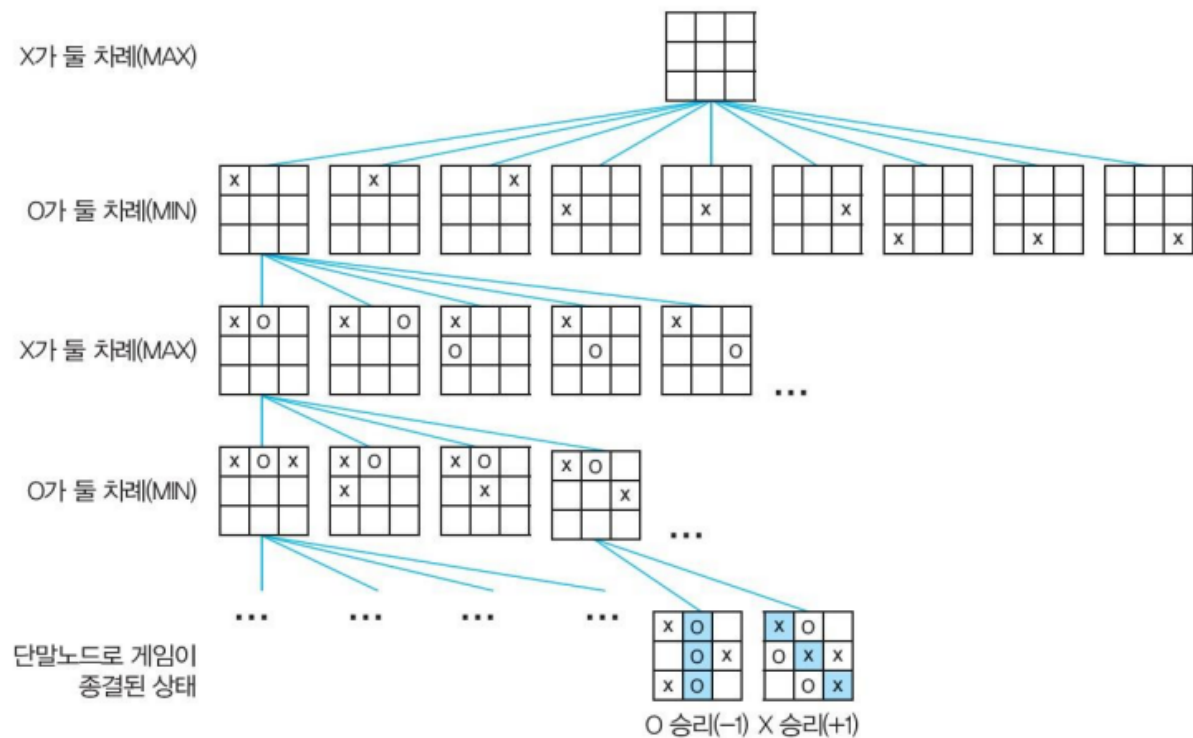
- 경우의 수는? $9! = 362,880$

- 상대방이 이기면 -1
- 내가 이기면 +1

- 깊이 우선 탐색

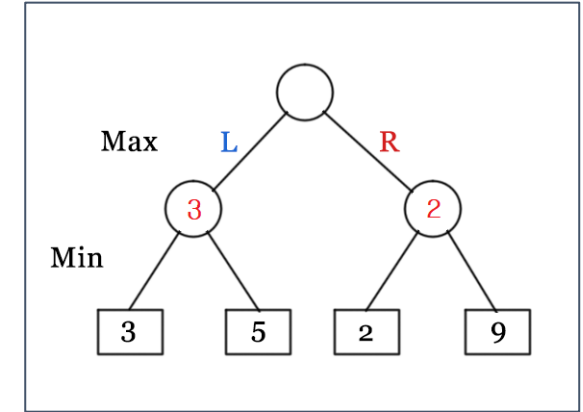
- Depth First Search (DFS)

- 매번 모든 경우의 수를
계산해야하나?

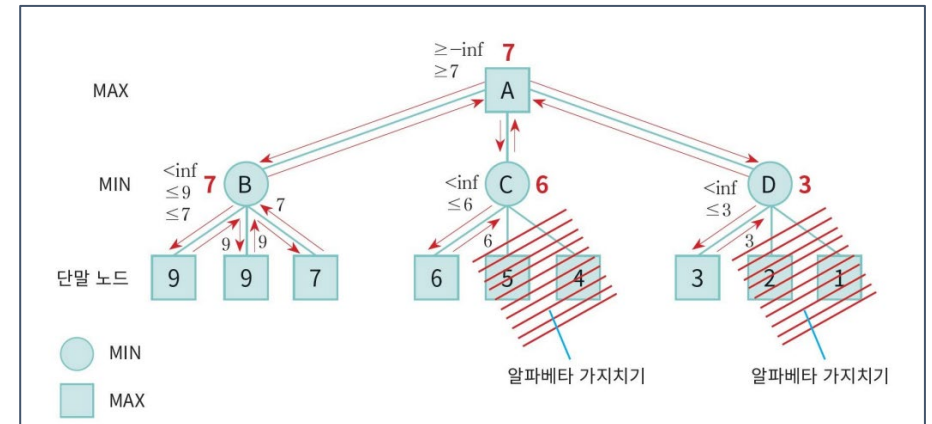


틱택토 알고리즘

- 미니맥스(minimax) 알고리즘
 - 가능한 상태를 시각화하여 트리 형태로 구성한 후,
 - 각 노드에서 이길 확률을 계산하여 확률이 높은 노드를 선택

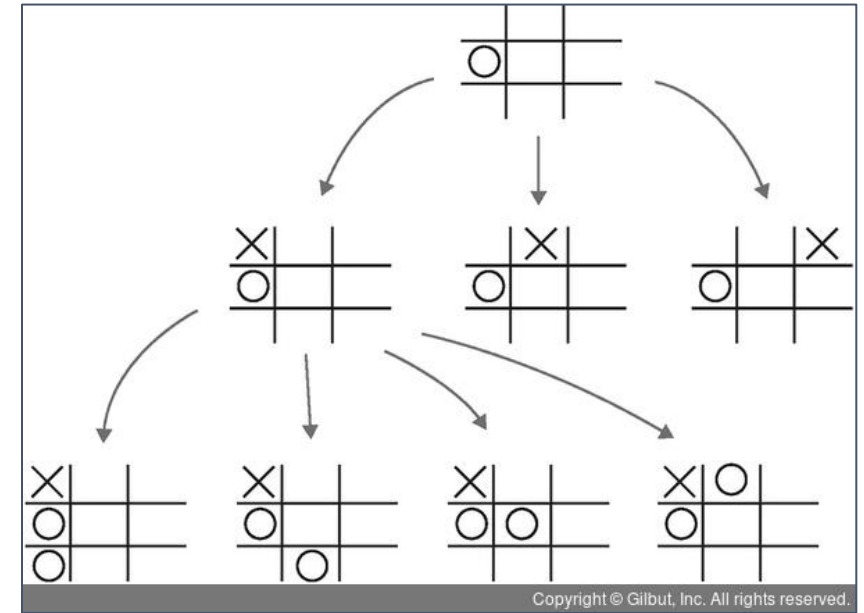


- 알파베타 가지치기
 - 미니맥스 알고리즘에서 결과에 영향을 주지 않는 가지들을 쳐낸다
- 몬테카를로 트리 탐색
 - 다음 장 참고

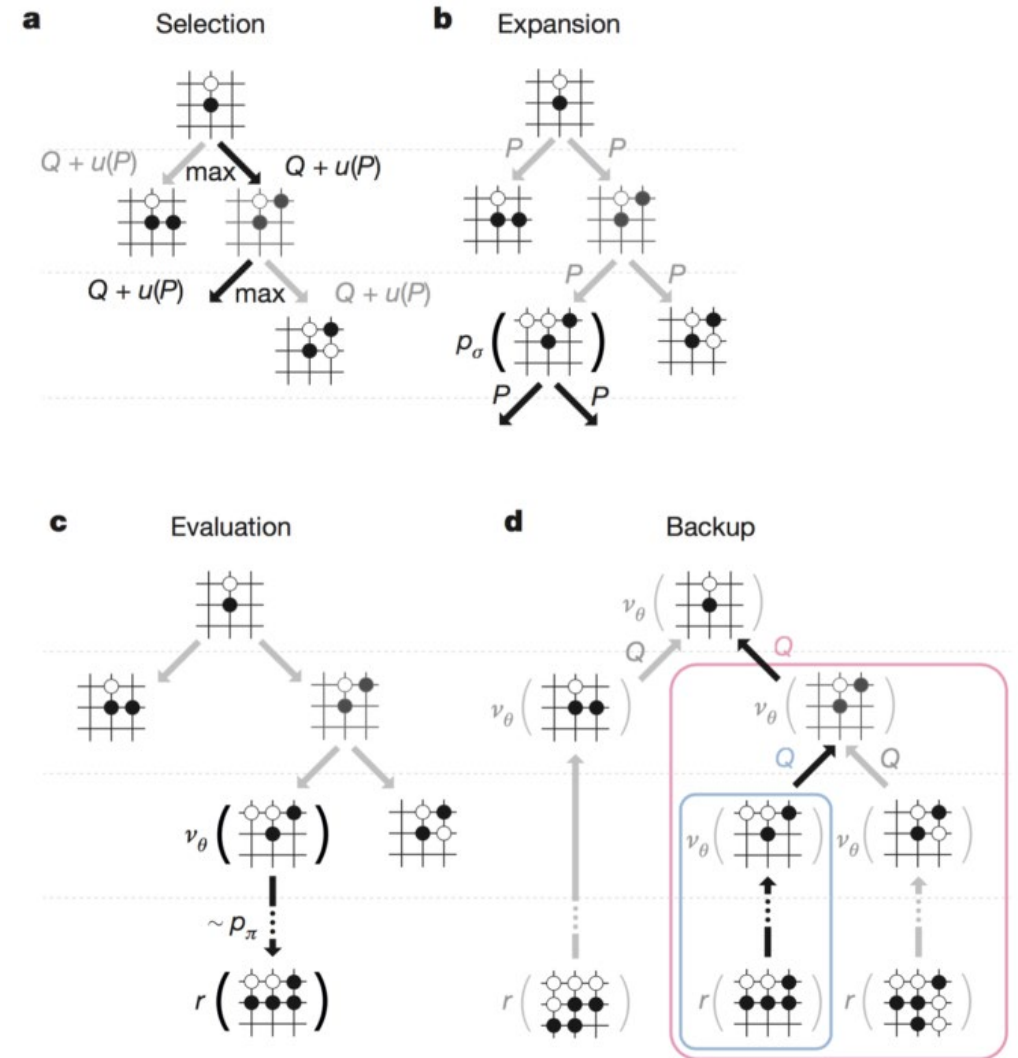
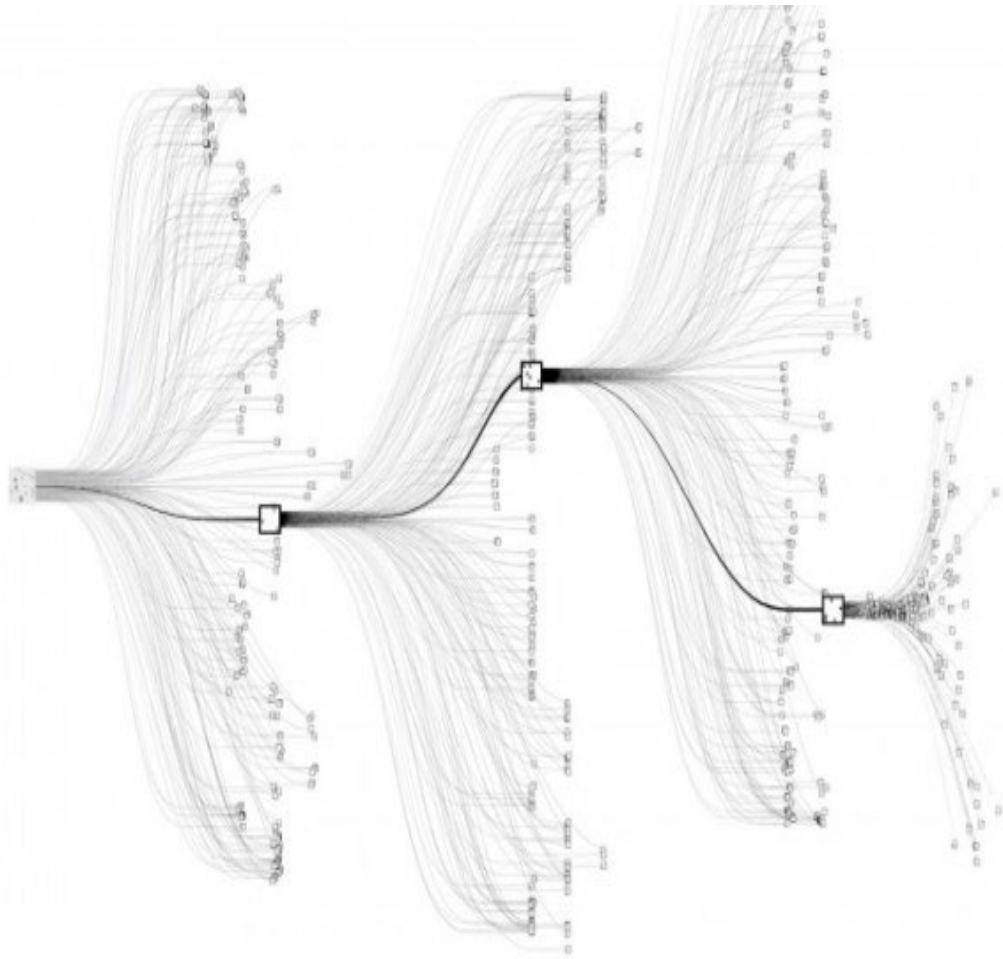


몬테카를로 트리 탐색의 기본 원리

- 모든 트리 노드를 대상으로 탐색하는 대신,
- 가장 가능성이 높아 보이는 방향으로 행동을 결정하는 탐색방법
- 즉, 경우의 수가 많을 때,
- 순차적으로 모든 경우의 수를 시도하는 것이 아닌,
- 무작위 방법 중 가장 승률이 높은 값을 기반으로 시도
- "선택 → 확장 → 시뮬레이션 → 역전파"를 반복함



몬테카를로 트리 탐색



4일차 3,4교시

인공지능으로 구현한 오목 게임

오목이란?

- 바둑판에서 연속해서 다섯개의 돌을 놓으면 승리
- m-n-k 게임의 일종
 - $m \times n$ 보드에서 가로, 세로, 대각선 방향으로 연속 k개의 돌을 두는 측이 승리
 - 오목 : 19 - 19 - 5 게임

오목 게임 실습

- "code3 omok.py" 다운로드
- 메모장에서 읽어서 복사
- 구글 코랩에 붙여넣기
 - (잠깐) 구글 코랩 GPU 사용하도록 설정하기
- 실행 버튼 or Ctrl+Enter
- 입력 방법 : x축 y축 (0부터 시작함)
- 코드가 맨 아래에 사람이 먼저 할 것인지, 컴퓨터가 먼저 할 것인지 정할 수 있음
- 오목 수준 정하기 : 많이 생각하게 하면 잘 됨

```
def mcts(state,player):  
    for i in range(10000): # 숫자를 크게하면 좋은 결과를 냄
```


인공지능끼리 오목 자율 대결

- "code4 omok_selfplay.py" 다운로드
- 메모장에서 읽어서 복사
- 구글 코랩에 붙여넣기
- 실행 버튼 or Ctrl+Enter
- 오목 수준 정하기 : 많이 생각하게 하면 잘 됨
 - def mcts(state,player):
 - for i in range(1000): # 숫자를 크게하면 좋은 결과를 냄

다음 시간

- 5일차 (7.31, 월)
 - 수업 오후 2시에 시작해서 4시간 수업 (2:00~5:50)