

The
best
strategy.



PYTHON

Simulator with Python

프로그래밍의 강점은 빠른 연산속도와 반복연산!

많은 연산 횟수라는 제약으로 인해
당연하게 받아들일 수 밖에 없던 이론적 현상을
간단한 파이썬 프로그래밍을 통해 직접 계산해봅니다.

Game Theory Simulator

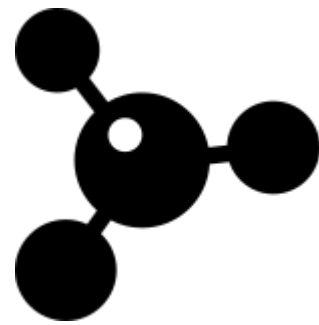
Tit For Tat

"매우 단순하지만 가장 완벽한 전략"

다른 모든 전략보다
평균적으로 가장 잘 작동하는 전략으로 알고 있는
'Tit For Tat' 전략을 반복적인 실험을 통해
현상이 이론을 잘 설명하는지 실증적으로 검증합니다.



Two Type of games



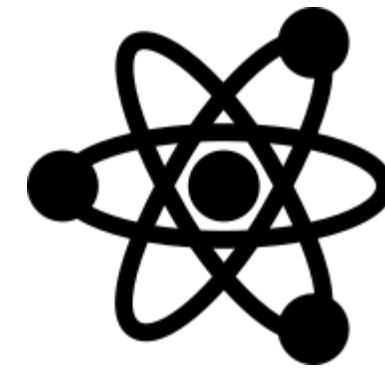
Game I

Player VS Computer

플레이어는 게임의 시작과 동시에, 상대방의 전략과 게임 회수를 지정할 수 있습니다.

지정한 라운드를 진행하며, 각 라운드에서 협조(1)와 배반(0)의 두가지 선택지 중 한가지 선택을 진행합니다.

게임이 끝나면, 해당 게임에서 얻을 수 있는 최고점과 최저점, 플레이어의 최종 점수를 확인 할 수 있습니다.



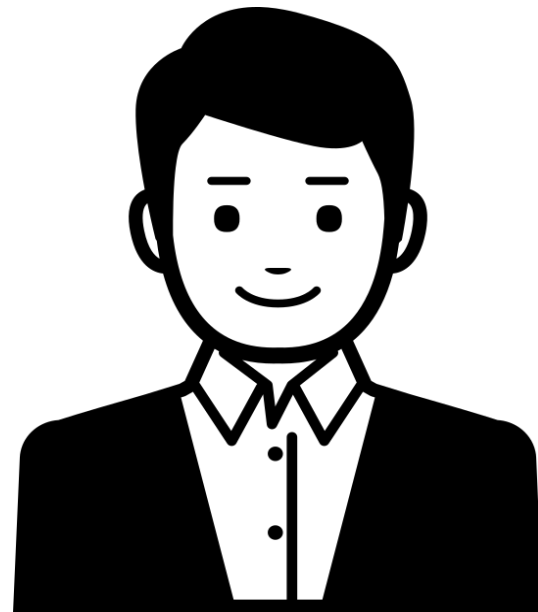
Game II

Game Simulator

플레이어는 시작과 동시에 게임의 횟수를 지정하거나 Random한 게임 횟수를 설정 할 수 있습니다.

지정한 라운드동안 컴퓨터는 서로 다른 7개의 전략 별 최종 점수와 게임의 결과를 연산합니다.

Seven Types of games



Tit For Tat

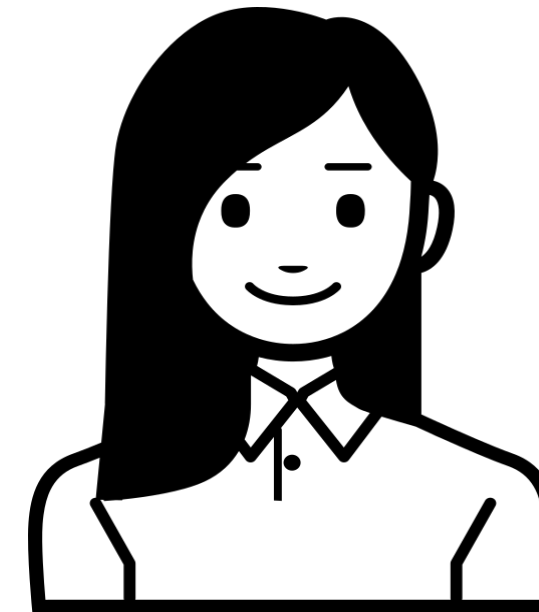
처음에는 항상 협력하지만, 이후에는 상대방이 이전 게임 라운드에서 했던 선택을 모방한다.



Always Betray

상대방의 선택과 관련 없이 항상 비협조적 태도를 유지한다.

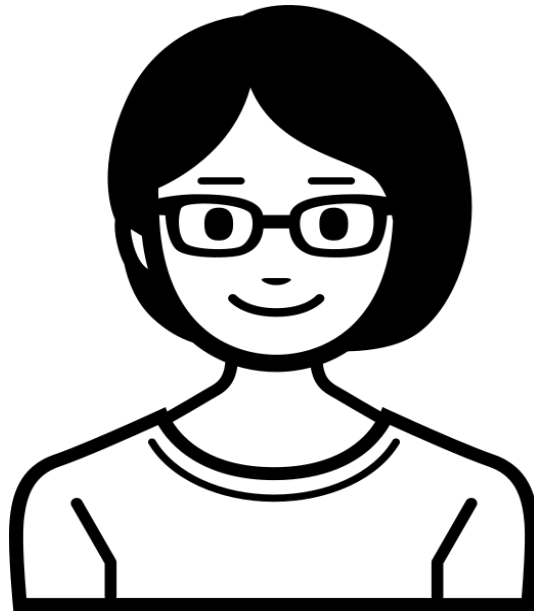
게임에서의 승률은 높지만, 얻게 되는 최종 이익은 대부분 적다.



Always Cooperate

상대방의 선택과 관련 없이 항상 협조적 태도를 유지한다.

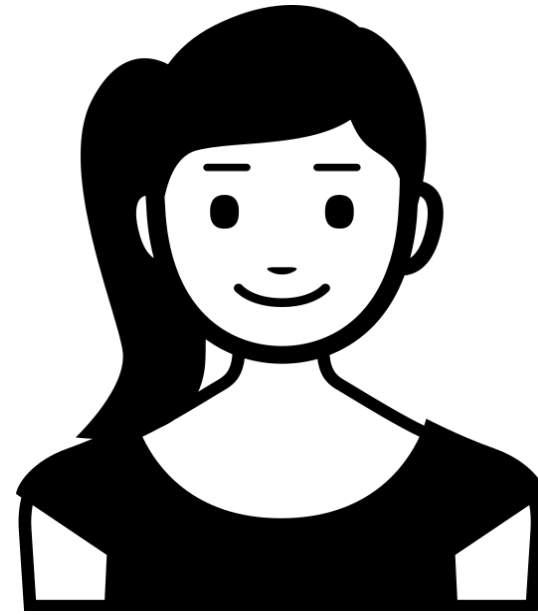
Seven Types of games



Prober

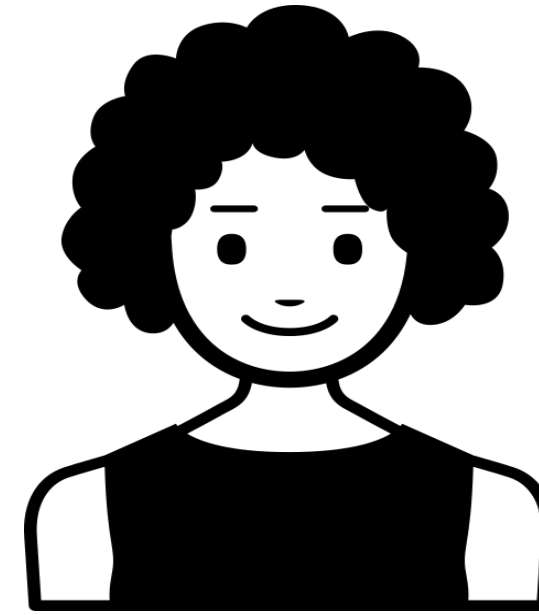
게임 시작 이후 4라운드 이전까지는 협조-비협조-협조-협조의 순서의 선택을 진행하고,

만약 4라운드 이전에 상대방으로 부터 배신을 당했다면, 그 순간부터 tit-for-tat 전략을, 배신당하지 않았다면 항상 배신하는 전략을 선택한다.



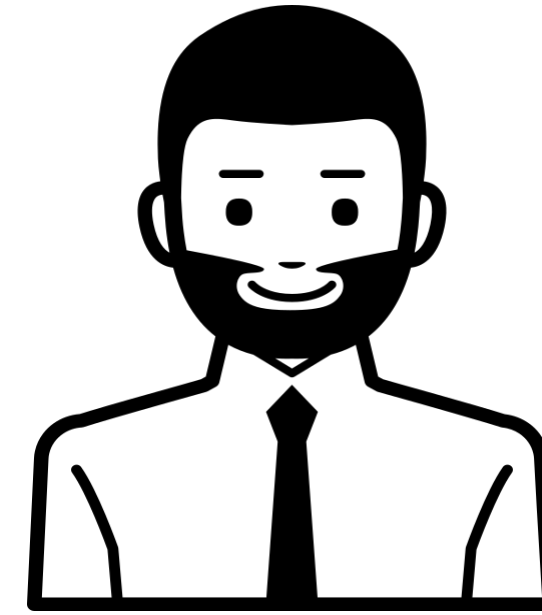
Random Selector

상대방의 전략과 관계없이 항상 랜덤의 선택을 진행한다.



Joss

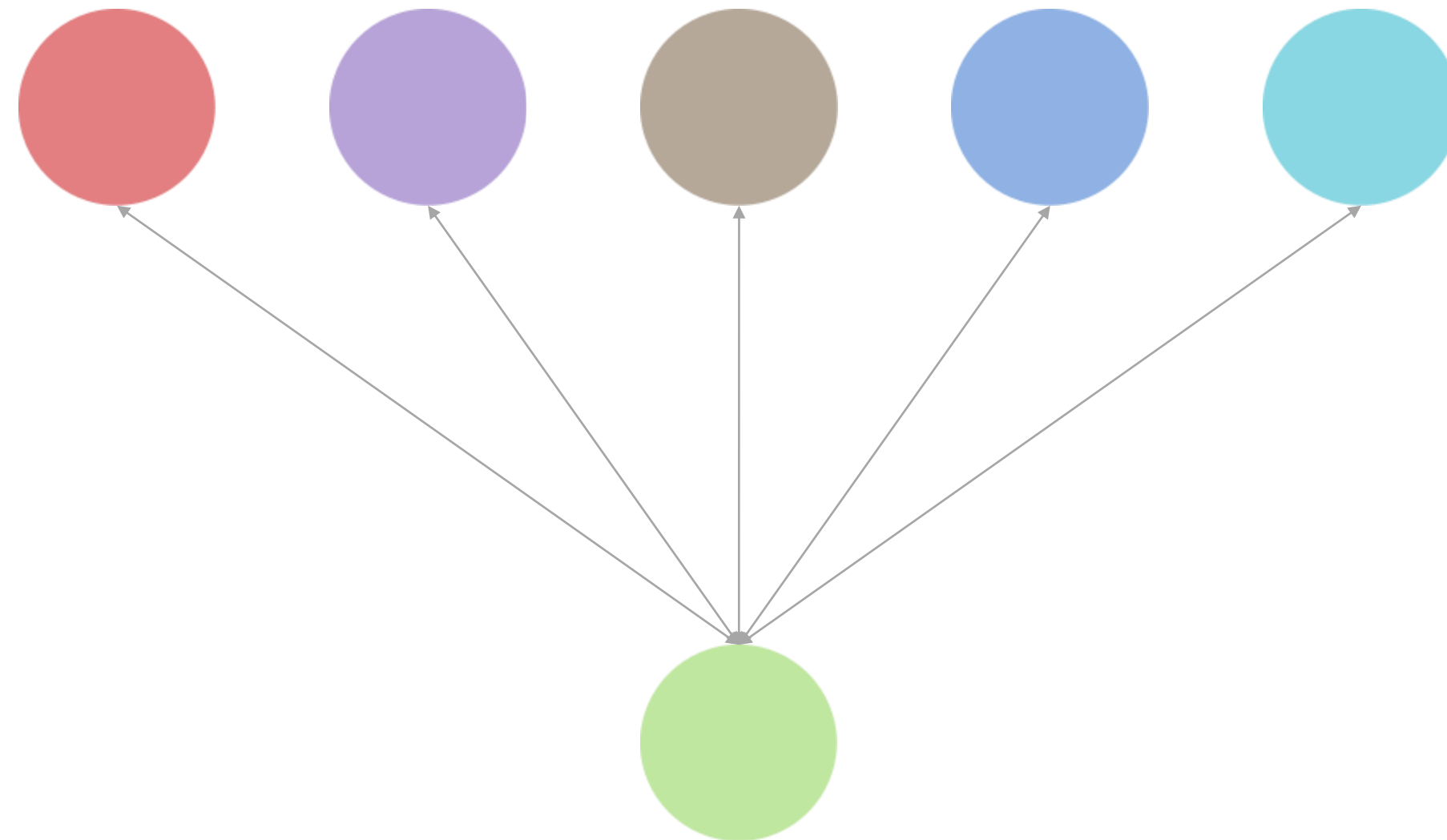
모든 것이 틱포탯과 동일하지만, 10%의 랜덤한 확률로 선제배반의 가능성을 갖는다.



Fried Man

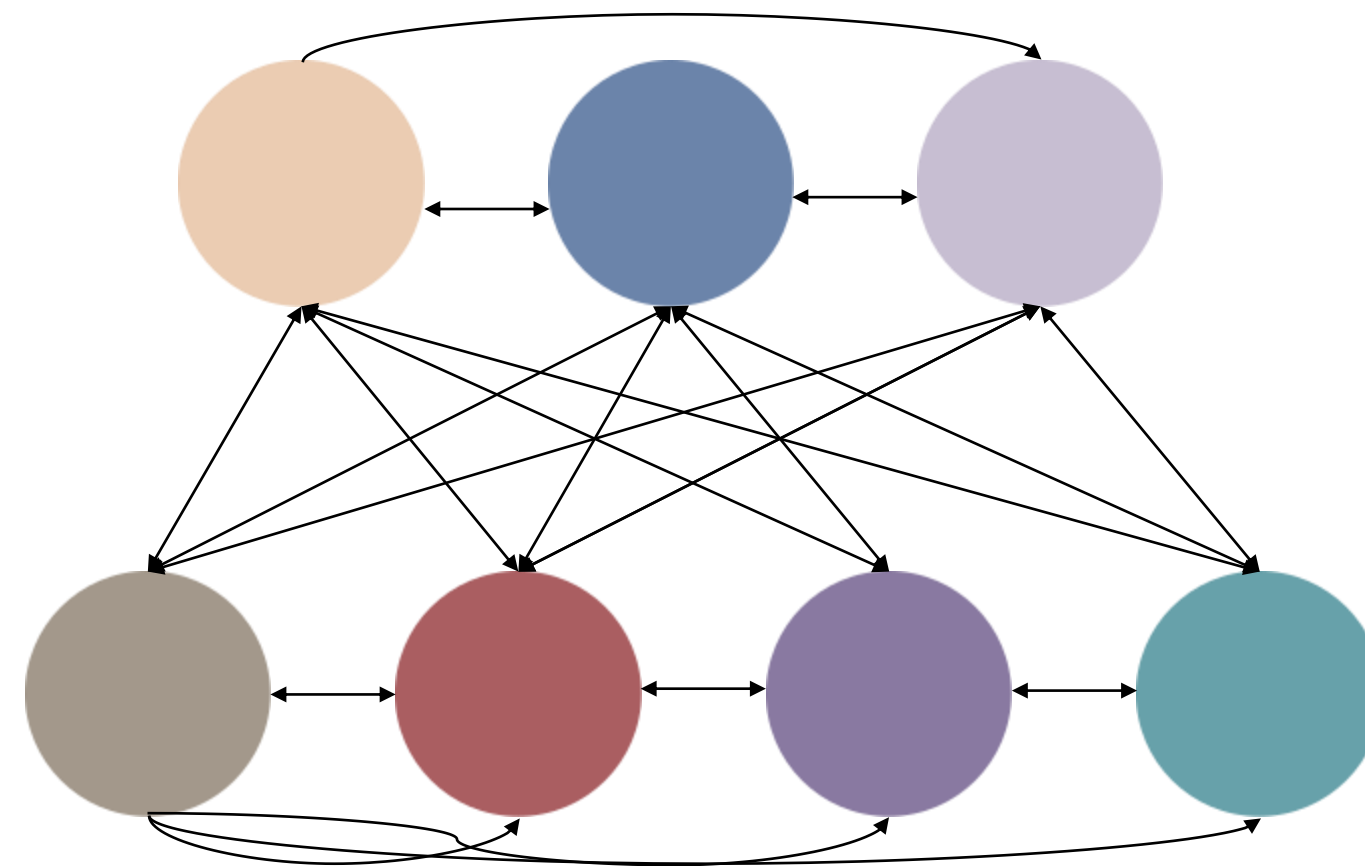
처음부터 계속 협력한다. 단, 상대가 배반을 하는 순간부터는 상대가 무엇을 하든지는 관계없이 게임이 끝날 때까지 계속 배반한다. 즉 틱포탯의 "관대함"을 완전히 포기한 프로그램이다.

Game I
Payer VS Computer



Game II

Computer simulator



Library That we used



PYTHON

Python 기본 제공 라이브러리만을 활용한 강력한 프로그램!

- ✓ Math
- ✓ Sys
- ✓ itertools
- ✓ Random

즉, Python의 기초 문법만 공부하면 강력한 프로그램을 작성할 수 있다!

프로젝트의 시사점

1

사람의 연산 능력을 보조

경제학 이론에는 복잡한 수식과 반복적인 계산을 요구하는 과정이 다수 존재합니다.

기본적인 코딩 능력을 습득한다면, 복잡하고 어려운 계산 과정을 손쉽게 다룰 수 있습니다.

2

경제학 이론 이해에 도움을 줌

복잡한 이론을 직접 눈으로 보여줌으로써 보다 구체적인 이해를 가능토록 해줍니다.

3

경제학부 학생들의 참여 유도.

재미있는 게임을 직접 실행함으로써, 코딩과정에 대한 호기심을 유도하고 이를 통해 코딩에 대한 학습을 진행하도록 유도합니다.

THANK
YOU