

# Blackjack Game with AI

SKKU Semiconductor Systems Engineering  
Young Dae KWON

# 1 Introduction

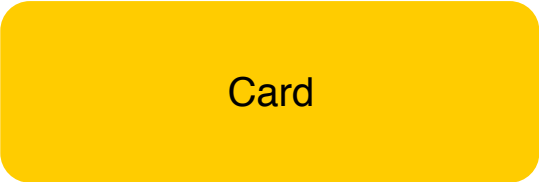
## A. 구현된 게임 규칙.

- 플레이어가 Hit, Stand 시에는 추가 배팅 없음.
- 플레이어가 Split이나 Doubling Down을 했을 경우에만 추가 배팅.

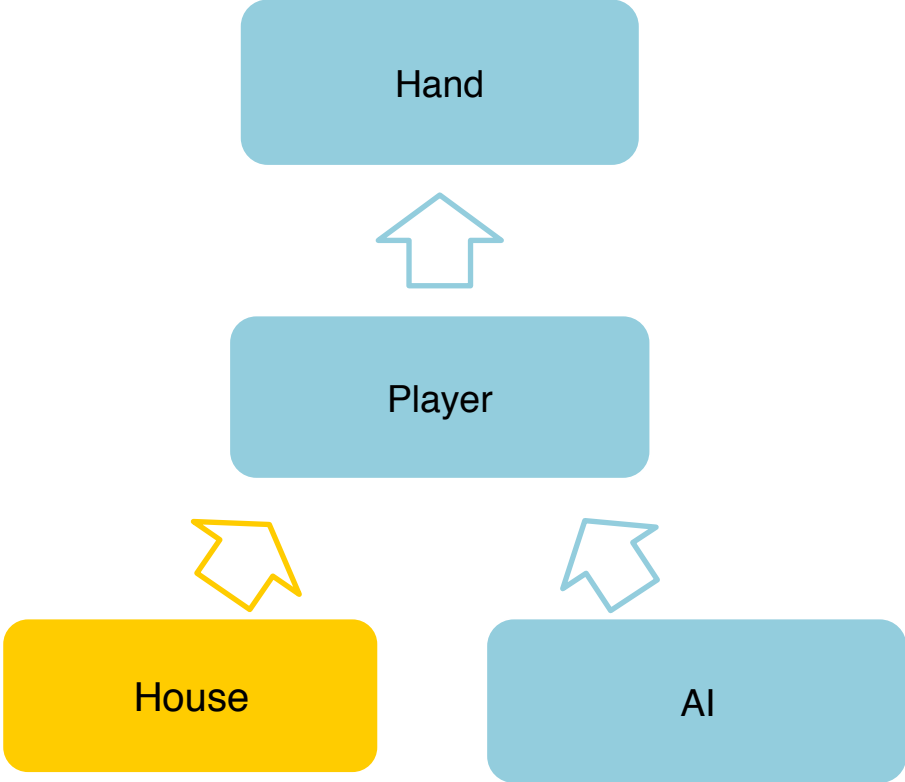


# 2 Model

## 상속관계 없는 Class

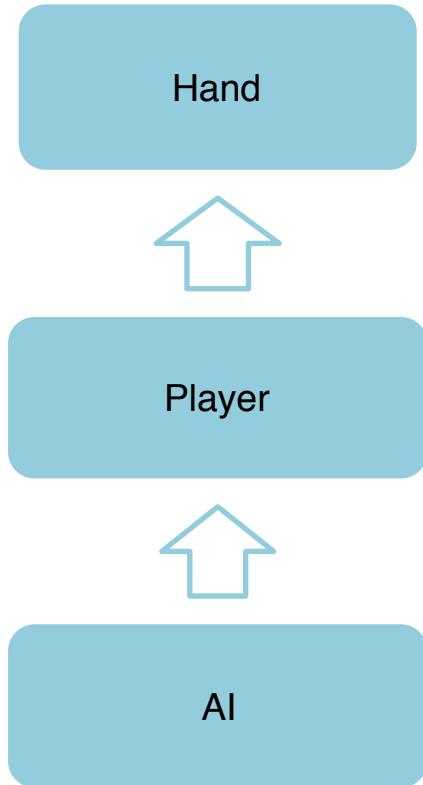


## Class 상속관계 그래프



## 2 Model

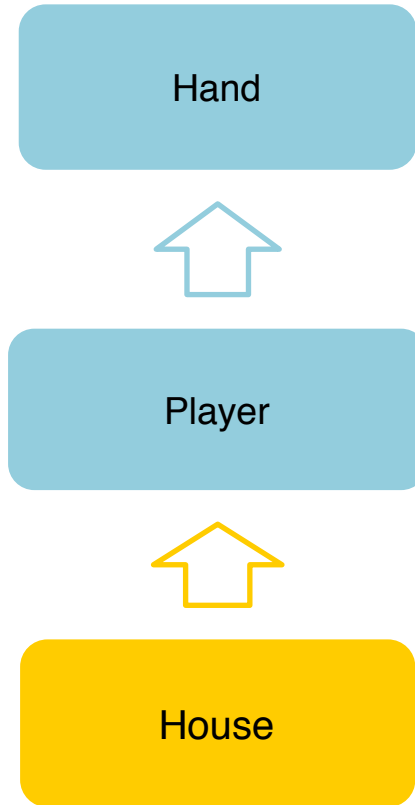
### Class 상속관계 그래프



1. Hand : 플레이어가 뽑은 카드와 점수, 배팅한 금액 등의 데이터를 관리하는 클래스
2. Player : Hand클래스를 상속받고 동적으로 Split기능을 구현하기 위한 클래스.
3. AI : Player클래스를 상속받고 AI가 지금까지 뽑은 카드, 보유한 총 금액, Player클래스 타입의 벡터 데이터를 관리하는 클래스.

## 2 Model

### Class 상속관계 그래프



1. Hand : 플레이어가 뽑은 카드와 점수, 배팅한 금액 등의 데이터를 관리하는 클래스
2. Player : Hand클래스를 상속받고 동적으로 Split기능을 구현하기 위한 클래스.
3. House : Player클래스를 상속받고 House의 역할을 수행하는 클래스

## 2 Model

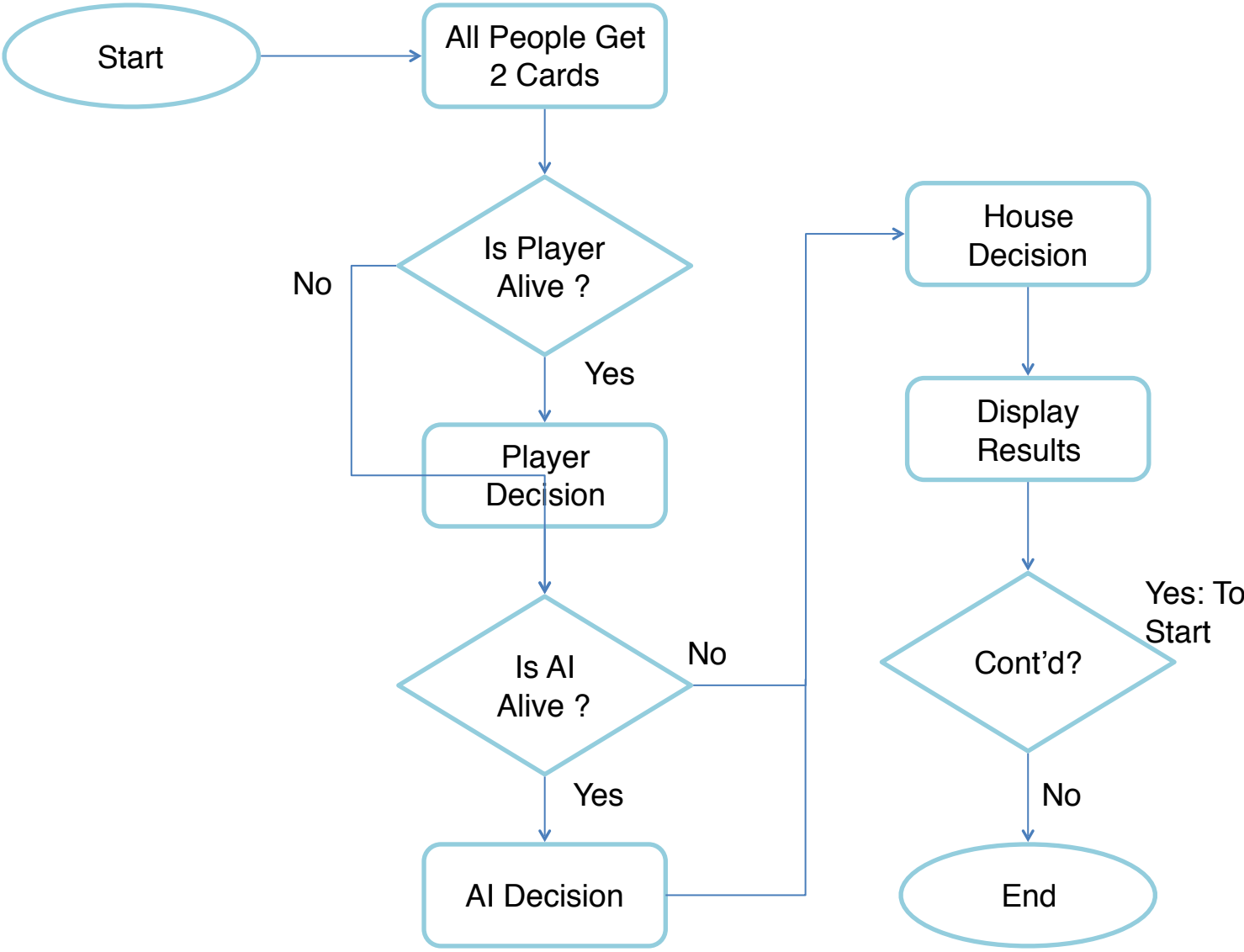
### 상속관계 없는 Class

Card

Game

1. Card : 게임에서 사용되는 카드 덱을 관리하는 클래스. 덱을 생성 및 셔플하고 카드 드로우의 기능을 지닌다.
2. Game : Blackjack Game의 시작 설정을 셋팅하고 실제로 구동시키는 클래스. AI클래스를 벡터로 사용하여 멀티AI를 구현한다.

# 3 Algorithm (Flow Chart)



### 3 Algorithm (Formula)

1. **Player**가 가질 수 있는 점수의 모든 경우의 수 **Count**
  - **Stand** 인 경우.
  - **Hit** 인 경우.
2. **House**가 가질 수 있는 점수의 모든 경우의 수 **Count**
  - 17점 이상인 경우: **Stand**
  - 13점 이상, 17점 미만인 경우: **Stand**
  - 13점 미만인 경우: **Hit**
3. **House**의 점수 **Count** 계산 후, 전체에 대해 2로 나누어 **Normalize** 시켜줌.



### 3 Algorithm (Formula)

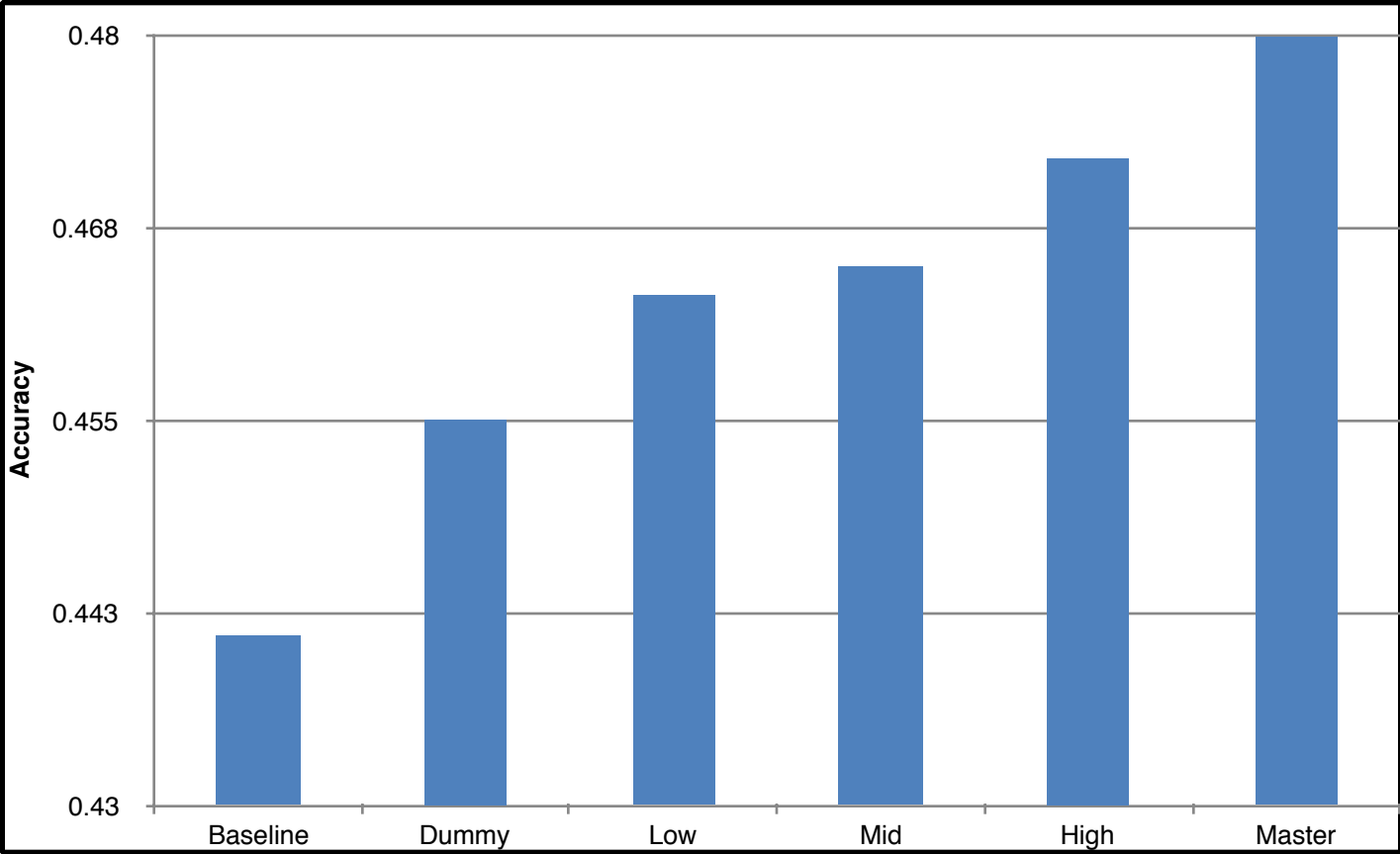
$$P(\text{Hit}) = \frac{(|\text{House Score} > 21| * |21 \geq \text{Player Score}|) + (|21 \geq \text{Player Score} > \text{House Score}| * |\text{Player Score} > \text{House Score}|)}{(\text{전체 House Score Count}) * (\text{전체 Player Score Count})}$$

$$P(\text{Stand}) = \frac{|\text{House Score} > 21| + |21 \geq \text{Player Score} > \text{House Score}|}{\text{전체 House Score Count}}$$

- **P(Hit) : Player가 Hit했을 시 이길 확률.**
- **P(Stand) : Player가 Stand 했을 시 이길 확률.**
- **AI의 Decision은 P(Hit) 와 P(Stand)의 비교를 통해 결정.**

# 4 Experiment

Blackjack Game AI 시뮬레이션 성능



## 5 Conclusion

- A. Hit 확률과 Stand 확률을 통계적으로 계산하는 방식이 효과가 있다는 것을 확인하였다.
- B. 위의 알고리즘을 적용한 AI의 금전 변화량에 긍정적 영향. 통계적인 정보와 직관적인 정보를 결합하여 좋은 성능을 이끌어 내었다.
- C. C++를 사용하여 OOP (Object Oriented Programming) 방식으로 구현해 봄으로써 C++에 대해 친숙해질 수 있었다.

만든지



권영대  
성균관대학교  
반도체시스템공학과 10

기획/개발