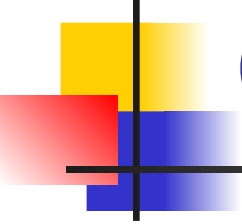


Black Jack

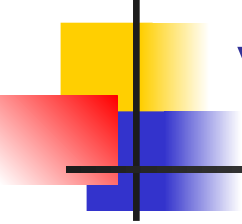
Dr. Bernard Chen

Troy University



Class Goal

- Upon completion of this course, you should be able to:
 - Know how to play black jack ^^
 - Program a simple black jack game that you can share with someone you like



Some major program skills you will use

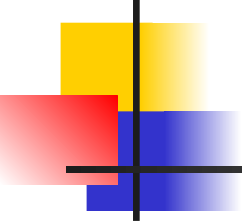
- Loop
- If statement
- Function

(So, basically, it will be a great practice on what you have learned so far on this semester)



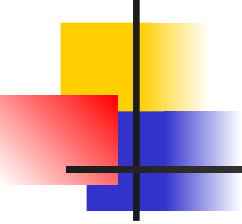
Black Jack Rules

- In blackjack at a casino, the dealer faces between five and nine (commonly seven) playing positions from behind a semicircular table. Each position may be occupied by up to three players. A single player is often permitted to control or bet on as many positions as desired. At the beginning of each round, bets are placed in the "betting box" at each position in play. The player whose bet is at the front of the betting box is deemed to have control over the position, and the dealer will consult the controlling player for playing decisions regarding the hand. Each wagered-on position is dealt an initial hand of two cards visible to the player of that box, and often to any other players. The dealer receives a single face up card, and in some games draws a second card, concealed from the players (the "hole" card, used to establish immediately whether the dealer has blackjack). In European casinos, the dealer's second card is neither drawn nor consulted until the players all play their hands.
- Cards are dealt either from one or two hand-held decks, from a box (known as a "shoe"), or from a shuffling machine. Single cards are dealt to each of wagered-on position clockwise from the dealer's leftmost position, followed by a single card to the dealer, followed by an additional card to each of the positions in play. The players' initial cards may be dealt face-up, or face-down (more common in single-deck games).
- The players' object is to win money by getting a card total which will turn out to be higher than the dealer's hand, but without exceeding 21 ("busting"/"breaking"), and taking advantageous opportunities to wager where the rules permit. Each hand is played by choosing whether to "hit" (take a card), "stand" (pass), "double" (double wager, take a single card and pass), "split" (make two hands out of a pair) or "surrender" (give up a half-bet and retire from the game). Number-cards count as their natural value; the jack, queen, and king (also known as "face cards" or "pictures") count as 10; aces are valued as either 1 or 11 according to the player's best interest. If the hand value exceeds 21 points, it busts, and its bet is immediately forfeit. After all boxes have finished playing, the dealer's hand is resolved by drawing cards until the hand busts or achieves a value of 17 or higher (a dealer total of 7 including an ace, or "soft 17", must be drawn to in some games). The dealer never doubles, splits nor surrenders. If the dealer busts, all remaining bets win and are paid out at 1:1. If the dealer does not bust, each remaining bet wins if its hand is higher than the dealer's, and loses if it is lower. In the case of a tied score, known as "push" or "standoff", the bet is normally returned without adjustment.
- The best possible hand is a "blackjack" or "natural", which is an ace and a ten-value card in either order on the initial two cards (not after a split). A blackjack beats any hand which is not a blackjack, even those with value 21. Blackjack vs. blackjack is a push. When the dealer's upcard is an ace, the player is usually allowed to make a side bet called "insurance," of up to half his wager, which pays 2:1 if the dealer gets a blackjack, and is forfeited otherwise.
- The minimum and maximum bets are posted on the table. The payoff on most bets is 1:1, meaning that the player wins the same amount as he or she bets. The traditional payoff for a player blackjack is 3:2, meaning that the casino pays \$3 for each \$2 originally bet, but many casinos today pay less at some tables.^[3]
- **House wins the tie.**



Black Jack

- Let's have some fun!!!



Black Jack Game basic logics

Initiation

Loop

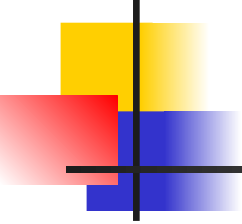
Deliver two cards to each player

Players' response

Deliver two cards to house

House's response

WIN or LOSE



Initiation

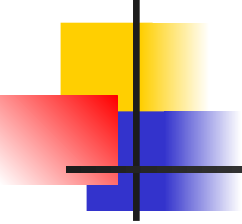
```
import random
```

```
print "Welcome to Black Jack Table!!"
```

```
num_players=input("How many players? ")
```

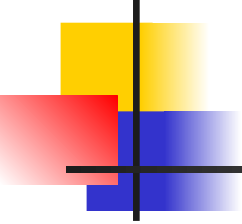
```
cards=[1,1,1,1,2,2,2,2,3,3,3,3,4,4,4,4,5,5,5,5,6,6,6,6,7,7,7,  
,7,8,8,8,8,9,9,9,9,10,10,10,10,11,11,11,11,12,12,12,12,13  
,13,13,13]
```

```
random.shuffle(cards)
```



Initiation

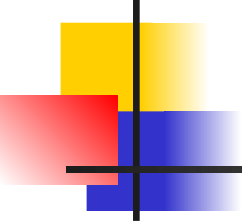
- Besides, we try to define two function for:
 - Print Card Image
 - Add Card points:
 - 1~9 (A=1 for simplicity)
 - 10, J, Q, K



Initiation (print out cards)

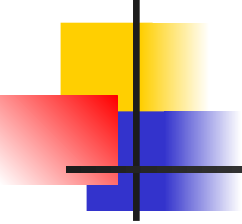
12
12
9
9

```
def print_card(x):
    if x < 10:
        print "-----"
        print "|",x,"|"
        print "|      |"
        print "|      |"
        print "|    ",x,"|"
        print "-----"
    else:
        print "-----"
        print "|",x,"|"
        print "|      |"
        print "|      |"
        print "|    ",x,"|"
        print "-----"
    return
```



Initiation

```
def card_add(sum, x):  
    if x < 10:  
        sum+=x  
    else:  
        sum+=10  
    return sum
```



Black Jack Game basic logics

Initiation

Loop

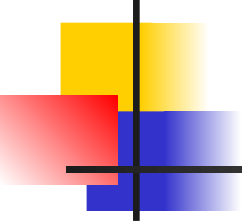
Deliver two cards to each player

Players' response

Deliver two cards to house

House's response

WIN or LOSE



Black Jack Game basic logics

Initiation

Loop

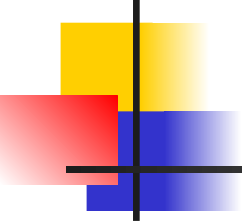
Deliver two cards to all players

Player's response

Deliver two cards to house (if
necessary)

House's response

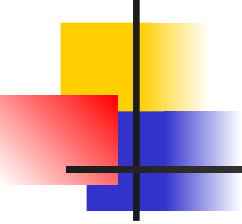
WIN or LOSE



Deliver two cards to player

Player's response

```
    player_sum = card_add(player_sum,  
cards[-1])  
    print_card(cards.pop())  
    player_sum = card_add(player_sum,  
cards[-1])  
    print_card(cards.pop())  
    print "Your point:",player_sum
```

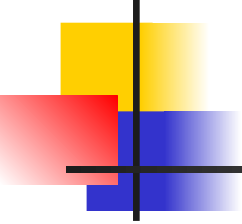


Deliver two cards to player

Player's response

- Build-in function

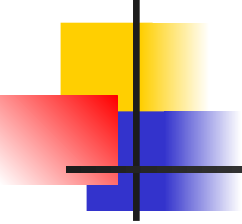
`cards.pop()`



Deliver two cards to player

Player's response

```
while (int(input('Do you need an extra card? (1:yes, 0:no)'))):  
    player_sum = card_add(player_sum, cards[-1])  
    print_card(cards.pop())  
    print "Your point:", player_sum  
    if player_sum > 21:  
        print "You lose!!!"  
        money -= chip_in  
        print "Now you have $", money, "dollars"  
        break
```



Black Jack Game basic logics

Initiation

Loop

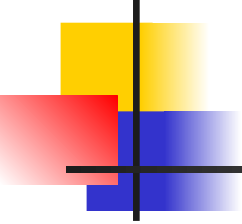
Deliver two cards to all players

Player's response

Deliver two cards to house (if
necessary)

House's response

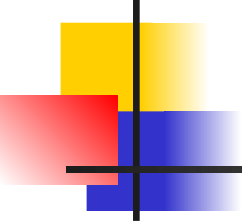
WIN or LOSE



Deliver two cards to house

House's response

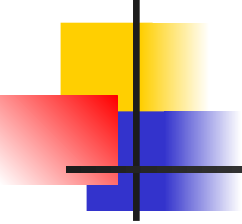
- If all player's point is > 21 , then house does not need to play anymore (House win)
- If player's point $== 21$, then house does not need to play anymore (Player win)
- Otherwise, house needs to play



Deliver two cards to house

House's response

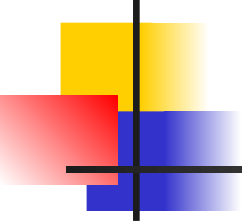
```
if player_sum == 21:  
    print "You Win!!!"  
    money+=chip_in*2  
    print "Now you have $", money, "dollars"
```



Deliver two cards to house

House's response

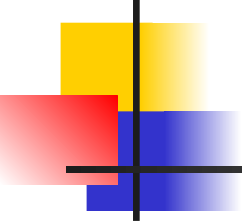
```
if player_sum < 21:
    print "Now, it's my turn..."
    house_sum = card_add(house_sum,
cards[-1])
    print_card(cards.pop())
    house_sum = card_add(house_sum,
cards[-1])
    print_card(cards.pop())
    print "House point:",house_sum
```



Deliver two cards to house

House's response

```
while (house_sum < player_sum):  
    house_sum = card_add(house_sum,  
cards[-1])  
    print_card(cards.pop())  
    print "House point:",house_sum
```



Black Jack Game basic logics

Initiation

Loop

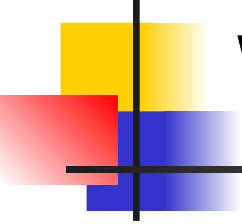
Deliver two cards to player

Player's response

Deliver two cards to house (if
necessary)

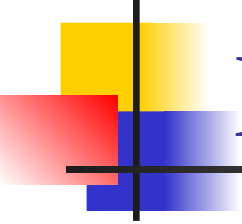
House's response

WIN or LOSE



WIN or LOSE

Print out the number of Players who wins



Number of decks: 3 in this project

- Using fewer decks decreases the house edge.
- This mainly reflects an increased likelihood of player blackjack,
 - since if the player draws a ten on his first card, the subsequent probability of drawing an ace is higher with fewer decks.

Number of Decks	House Advantage
Single deck	0.17%
Double deck	0.46%
Four decks	0.60%
Six decks	0.64%
Eight decks	0.66%