Team: BlackJack Noobz

Members: Jorge Bautista & Luis Batz

**CIS-7 Project: BlackJack Game**

**Our program is the BlackJack game:**

The premise of the game is to simulate a BlackJack game. The game involves a deck of playing cards with assigned values. The game is designed to ask the player if he is ready to play, if the player chooses “Yes” it will continue and ask how much “money” the player wants to bet. The goal of the game is to win against the computer. The computer, much like in a real game of BlackJack, must get as close to 21 as possible without going over. The computer has rules like a real card dealer has. If the computer draws card values totaling 17 or more then it will stay and not draw any more cards from the deck. If It draws 16 or less it will automatically draw a card. The other player, in our case the human Player1, is not bound by these rules. Even if the players draws 17 he/she may still draw more from the deck. If you are tied with the computer it is best to just end the game because you don’t win or lose the game therefore will not lose any money, this is called a “Push”.

**Card values with odds explained:**

Cards with the numbers 2 through 10 are all worth their face value regardless of suit. Jack, Queen and King are all worth ten points regardless of suit. Aces are the only cards with dual values, they can be worth a 1 or an 11.

Odds of each card being picked is 1/52 because the sample space of the deck is 52 cards. Each card is an event.

Odds of the same card number in different suits is 4/52 or 1/13

Odds of a King, Queen, Jack or a 10 is 16/52 or 1/3.25 and this is significant because the cards are worth the same whether it is a King, Queen, Jack or a Ten in any suit for all cards. Those cards represent almost one third of the total cards so the odds of getting those cards is quite high.

The odds of getting a perfect 21 on BlackJack are: 128/2652 or 1/21 hands of play.

The Ace card is the tricky one because it has a probability of 4/52 or 1/13 but it has two different values. It can be worth a 1 or an 11.

**Odds of “busting” which is going over the targeted 21 after drawing 17:**

Going over the score of 21 after drawing 17 is roughly: 30/50 or 1/1.6 because you already drew 2 cards in the greater than 5 range and anything over 4 will give you a ”bust”. Unless you draw an Ace and a 6 which only removes one card which means that the odds change very little. So odds are very high that you will draw a higher than 5 card because we have established that there are 30 cards remaining in the deck that will cause you to lose because they will give you more points than you need.

So what is it the the dealer knows? Staying on 17 an not drawing anymore cards makes it seem like the dealer knows something. Let’s see if we can take a look at the odds of getting to 21 after drawing a 17 with the first two cards.

**Odds of drawing a 4 or less to get closer to 21 after drawing 17:**

There are only 16 cards that can get you closer to the score of 21 without going over. That means that you have a 16/50 or 1/3.12 if you already have a score of 17. The odds of getting the 4 value card in any suit is roughly 4/48 or 1/12 which is extremely low. Why 48 cards? Because 4 cards have already been drawn. With the odds of getting a card that will cause the dealer to lose already set at 1/1.6 you can see why the dealer will stay on 17. This means that if you decide to follow the same guide as the dealer then you have a higher percentage of winning the game.

**If the dealer draws a 16 or less:**

There are now 20 cards that will get you closer to 21 without going over and that is 20/48 or 1/2.4 which is more favorable and therefore the dealer sees it as less of a risk. Odds of getting a 5 value card are roughly 4/48 or 1/12 which is very low. Again if you copy the dealer’s strategy then you also have better chances of winning the game. If you decide to go for it on 16 points or less then you are facing a 30% chance of drawing a 10 which will cause you to lose the game. You are still facing a 32/48 or 1/1.5 of drawing a card that will get you 22 points or higher which will cause you to lose the game.

**Program Goals:**

Some of the things that the program needs to achieve is to give random cards to player1 and the dealer. Permutation is needed to make sure that the same card number in the same suit is dealt to both players. If this happens then the program does not work. We hope that the program gives us insight to the inner workings of a BlackJack game. Show the reasoning through odds as to why the dealer has set rules and the player does not. We hope that this game shows why it might be a good idea to follow dealer rules instead of leaving it to chance. We have already seen that the odds are stacked against us from the start if Player1 or the Dealer draws a certain range of cards.

**Some of the functions of the game/program are:**

Dealing the cards in random and correct order. Each player is to get tow cards on the initial draw and the cards cannot be identical but they can have the same value. What I mean is that both players can draw 18 by a combination of:

Player1: Jack of Clubs + 8 of Diamonds

Computer: Queen Diamonds + 8 of Clubs

The players cannot draw the same exact card in the same exact suit because it will mean the the program is flawed.

The program is to get the totals of the cards and follow the set rules for the dealer. The player has more freedom to draw cards after the initial draw regardless of what the initial score is.

**Class Lesson Applications:**

Probability: The mathematical theory of **probability** is a way of formally **representing and reasoning about uncertain events.**

Permutation: is an ordered arrangement of a set or subset of objects.

**C++ Implementation in the program:**

The program uses Random which is:

An experiment or random experiment yields one of a possible set of outcomes. The sample

space is the set of all possible outcomes. We will assume that all outcomes are equally likely.

An event is a subset of the sample space, i.e. a set of outcomes.

Permutation:

A permutation is an ordered arrangement of a set (or subset) of objects.

Algorithm implementation is to make sure that the program draws random cards for every hand. The point values can match but the cards dealt cannot be the same to different players so the algorithm is used to randomize the cards and deal them that way.

**Limitations of the BlackJack game:**

The game is limited in the sense that it may not represent a real live BlackJack game. The number of players is limited to 2 players which are Player1 and computer so they must play against each other.

**Improving limitations:**

There aren’t any improvements that I can think of. It is meant to be a game with set parameters and rules for a person to see what happens in a computer simulation of the game.

**Flowchart/pseudocode of the program:**

Yes/No

Deal cards to player1/dealer

Player1 responds sets “bet”

Ready player:

Win or bust?

Wait on Dealer

Win? Yes or NO

Win? Yes or NO

Wait on Player1

Get new score

Assign 1 card to each player

>17 take card from deck

17 or more Stay

Draw cards from deck

Stay

Hit

Player1

Dealer

Assign cards

Does the dealer have 17 or > 17

Show totals

Get totals

Player1

Dealer