CSC 101 Programming Assignments 4 and 5
Blackjack

DUE:

> *Recommended for Program 4:* Wednesday 23 May
> *Required for both Programs 4 and 5:*
> On or before 11:59:59PM Wednesday 6 June

Overview

In these assignments you are writing a program to play the card game of Blackjack. You can read a general description of the game on Wikipedia at `http://en.wikipedia.org/wiki/Blackjack` . The program you are writing is for a much simplified version of the game. In particular, your program only has one player and one dealer.

This is a combined assignment for programs 4 and 5. The due date for programming assignment 4 is *recommended*. This means that in order to make good progress on the assignment, you should have assignment 4 completed by the recommended date. However, you are not required to submit the program by this date. The required due date for the combined assignment is Wednesday 6 June.

Programming Assignment 4: Basic Play

Blackjack is a card game in which the object is to get a hand that is as close as possible to 21, without going over. A player starts out with two cards and can request more cards, one at a time. Numeric cards have their numeric value, and face cards have value 10. An ace has the value 1 or 11, which ever value is more advantageous to the user. For example, a hand with two aces and one five has the value of $11 + 1 + 5 = 17$. In this case, the second ace is counted as a value of 1, since if it were counted 11 the hand value would be 27. A two-card hand with an ace and 10 is always counted as 21, and is called a "blackjack" hand.

The user of the program is the player, and the program acts as the dealer. The program begins by dealing two cards each to the player and the dealer. The values of player's cards are shown. The value of one of the dealer's card is shown, with the other value hidden until the hand is over. The hidden dealer card is called the "hole" card.

Once the player's hand is dealt, the player inputs the following one-character commands:

Command	Char	Description
hit	h	take another card
stand	s	stand pat on hand, proceed to dealer play
quit hand	q	quit hand, proceed to next hand

If the player's hand value goes above 21, the player "busts" and loses the hand. The dealer reveals the hole card, but does take any further cards. The hand then concludes.

If the player does not bust, or the player's hand reaches a value of 21, then the dealer plays. This is done by the dealer taking cards until the dealer's hand value is 17 or higher, or the dealer busts. If the dealer's hand has a value of 17 and the hand includes an ace, then the dealer must take one more card. At the conclusion of dealer play, all dealer cards are revealed, including the hole card.

If neither the player nor dealer busts, then the winning hand is the one with the higher value. If the hand values are the same, then the hand is a tie, which is called a "push". A two-card blackjack hand beats a hand of value 21 with more than two cards.

During game play, cards are displayed in the form used in the Program 3. The dealer's hole card is displayed as 'X'. When the player's hand value reaches 21, the player is not prompted for a command, and the dealer's play commences automatically.

If the player uses the 'q' command, the player loses immediately. The dealer does not play, and the hole card is not revealed.

Sample Basic Play

```
unix1> blackjack

Player's Hand: 3S, JC
Dealer's Hand: 5H, X

Enter command: h

Player's Hand: 3S, JC, 9H
Dealer's Hand: 5H, 2D

Player busts.

Enter 'q' to quit, anything else to play another hand: c


Player's Hand: KS, 8D
Dealer's Hand: 9C, X

Enter command: s

Player's Hand: KS, 8D
Dealer's Hand: 9C, 3H, 10H

Dealer busts.

Enter 'q' to quit, anything else to play another hand: c


Player's Hand: 2H, 4C
Dealer's Hand: QH, X

Enter command: h

Player's Hand: 2H, 4C, 10C
Dealer's Hand: QH, X

Enter command: h

Player's Hand: 2H, 4C, 10C, 3D
Dealer's Hand: QH, X

Enter command: s

Player's Hand: 2H, 4C, 10C, 3D
Dealer's Hand: QH, JS

Dealer wins.

Enter 'q' to quit, anything else to play another hand: c


Player's Hand: AS, QS
Dealer's Hand: 3C, 8H, 10D

Player wins.

Enter 'q' to quit, anything else to play another hand: c


Player's Hand: 2C, 2S
```

```
    Dealer's Hand: 5D, X

    Enter command: h

    Player's Hand: 2C, 2S, 4H
    Dealer's Hand: 5D, X

    Enter command: h

    Player's Hand: 2C, 2S, 4H, 4D
    Dealer's Hand: 5D, X

    Enter command: h

    Player's Hand: 2C, 2S, 4H, 4D, 4S
    Dealer's Hand: 5D, X

    Enter command: s

    Player's Hand: 2C, 2S, 4H, 4D, 4S
    Dealer's Hand: 5D, 10S, 5S

    Dealer wins.

    Enter 'q' to quit, anything else to play another hand: q
```

After 52 cards have been dealt, the program starts dealing blank cards, which have a value of 0. Play can continue, but it is meaningless. Alternatively, the program can simply terminate after dealing 52 cards. If the user wants to play more meaningful hands, the user must restart the program. The extra feature of reshuffling will allow meaningful play to continue without restarting the program after 52 cards have been dealt.

Programming Assignment 5: Extra Features

The extra features to be added for program 5 are deck reshuffling, hand splitting, betting, and error checking. These features are added to the basic play version of the program you do for assignment 4. The features can be added independently from each other. For example, you may implement reshuffling and betting without splitting and error checking.

Reshuffling

As described above, the program starts dealing blank cards after 52 cards have been dealt from the deck. The reshuffling feature of the program detects when 52 cards have been dealt, and continues play uninterrupted by restarting dealing with a new shuffled deck of 52 cards. Whenever reshuffling is needed, it happens automatically. The user is informed when reshuffling happens with the output of the message "Reshuffling ...". After the message, play continues where it left off.

Here is an example of reshuffling during a hand:

```
    Player's Hand: 2H, 4C
    Dealer's Hand: QH, X

    Enter command: h

    Player's Hand: 2H, 4C, 10C
    Dealer's Hand: QH, X

    Enter command: h

    Reshuffling ...

    Player's Hand: 2H, 4C, 10C, 3D
    Dealer's Hand: QH, X
```

```
Enter command: s

Player's Hand: 2H, 4C, 10C, 3D
Dealer's Hand: QH, JS

Dealer wins.
```

Hand Splitting

When the player has a hand with two of the same cards, the player may choose to split the hands into two separate hands. To do this, there is a new command character '2', which indicates that the player wants to split. This command is only available as the first player command in a hand. That is, splitting can only happen immediately after the hand is dealt, and the hand has two cards of the same value. Splitting is allowed on hands with two different face cards, e.g, a Jack and a Queen, since these both have a value of 10.

Play on the separate hands proceeds individually, in the same way as for a single hand. The first hand is played to conclusion, then the second hand, then the dealer's hand if necessary. The precise play of the hand goes like this:

- If the first hand busts, the message "Player busts." is output, and play proceeds to the second player hand.

- If the second hand also busts, the message "Player busts." is output again, and play concludes by printing all three hands, with the dealer hole card revealed

- If one or both player hands does not bust, then the dealer plays, and two messages are output indicating the conclusion of each of the split hands.

For example,

```
Player's Hand: 2C, 2S
Dealer's Hand: 5D, X

Enter command: 2

Player's First Hand: 2C, 4H
Player's Second Hand: 2S, 4D
Dealer's Hand: 5D, X

Enter command for first hand: h

Player's First Hand: 2C, 4H, 4S
Player's Second Hand: 2S, 4D
Dealer's Hand: 5D, X

Enter command for first hand: h

Player's First Hand: 2C, 4H, 4S, 8C
Player's Second Hand: 2S, 4D
Dealer's Hand: 5D, X

Enter command for first hand: s

Player's First Hand: 2C, 4H, 4S, 8C
Player's Second Hand: 2S, 4D
Dealer's Hand: 5D, X

Enter command for second hand: h

Player's First Hand: 2C, 4H, 4S, 8C
Player's Second Hand: 2S, 4D, 10C
Dealer's Hand: 5D, X

Enter command for second hand: h
```

```
Player's First Hand: 2C, 4H, 4S, 8C
Player's Second Hand: 2S, 4D, 10C, 9D
Dealer's Hand: 5D, X

Player busts.

Player's First Hand: 2C, 4H, 4S, 8C
Player's Second Hand: 2S, 4D, 10C, 9D
Dealer's Hand: 5D, 9C, 3S

Player wins first hand.
Dealer wins second hand.
```

Betting

The betting feature of the program allows the user to enter a bet before each hand. During the game, the program keeps a running total of the player's winnings. The total starts at zero, and is incremented or decremented after each hand, depending on whether the player wins or loses. When a hand is finished, the player's total winnings are printed.

The betting feature includes one more command that the player can choose during a hand. The command is 'd', for "double down". This command is only available immediately after the player sees the initial deal, and the initial deal is not a blackjack hand. If the player gives the 'd' command, the player's initial bet is doubled in value. The hand play then continues as normal, including hand splitting if that option is available.

The following table defines how winnings are computed for a hand:

Hand Outcomes	Win/Loss Amount
player wins regular hand	bet
dealer wins regular hand	-bet
player wins with blackjack hand	1.5 * bet
player and dealer tie	0
player wins doubled down hand	2 * bet
player wins doubled down hand with blackjack	3 * bet
player wins both split hands	2 * bet
player loses both split hands	-2 * bet
player wins one, loses one split hands	0
player win one, ties one split hands	bet
player loses one, tie one split hands	-bet

Here is the sample play from the beginning of the writeup, this time with betting:

```
unix1> blackjack

Place your bet: 5

Player's Hand: 3S, JC
Dealer's Hand: 5H, X

Enter command: h

Player's Hand: 3S, JC, 9H
Dealer's Hand: 5H, 2D

Player busts.

Winnings: -5

Enter 'q' to quit, anything else to play another hand: c
```

```
Place your bet: 10

Player's Hand: KS, 8D
Dealer's hand: 9C, X

Enter command: s

Player's Hand: KS, 8D
Dealer's hand: 9C, 3H, 10H

Dealer busts.

Winnings: +5

Enter 'q' to quit, anything else to play another hand: c

Place your bet: 5

Player's Hand: 2H, 4C
Dealer's Hand: QH X

Enter command: h

Player's Hand: 2H, 4C, 10C
Dealer's Hand: QH X

Enter command: h

Player's Hand: 2H, 4C, 10C, 3D
Dealer's Hand: QH X

Enter command: s

Player's Hand: 2H, 4C, 10C, 3D
Dealer's Hand: QH JS

Dealer wins.

Winnings: 0

Enter 'q' to quit, anything else to play another hand: c

Place your bet: 10

Player's Hand: AS, QH
Dealer's Hand: 3C, 8H, 10D

Player wins.

Winnings: +15

Enter 'q' to quit, anything else to play another hand: c

Place your bet: 10

Player's Hand: 2C, 2S
Dealer's Hand: 5D, X

Enter command: d

Bet doubled to 20
```

```
Enter command: h

Player's Hand: 2C, 2S, 4H
Dealer's Hand: 5D, X

Enter command: h

Player's Hand: 2C, 2S, 4H, 4D
Dealer's Hand: 5D, X

Enter command: h

Player's Hand: 2C, 2S, 4H, 4D, 4S
Dealer's Hand: 5D, X

Enter command: h

Player's Hand: 2C, 2S, 4H, 4D, 4S
Dealer's Hand: 5D, 10S, 5S

Dealer wins.

Winnings: -5

Enter 'q' to quit, anything else to play another hand: q
```

Error Checking

Without the extra error checking features, the program requires no error checking of any form. With the feature, the following conditions are detected, with the indicated error message output:

Error Condition	Message
illegal command character 'x'	`"Command 'x' unrecognized"`
bet input not integer between 1 and 1000000	`"Bet must be between 1 and 1,000,000"`
double down command not first	`"Double down can only be first command"`
split command not first or second	`"Split can only be first or second command"`

If both double down and split commands are used, double down must come first, then split.

Program Structure

Your solution to the combined assignments 4 and 5 is comprised of at least four files: `blackjack.h`, `blackjack.c`, `cards.h` and `cards.c`. You are writing `blackjack.h` and `blackjack.c` from scratch. For `cards.h` and `cards.c`, you can use your solution to program 3, or the files in the 101 solutions directory.

Your blackjack program can use the `shuffle` and `deal_two_hands` functions from the Program 3 solution. For playing blackjack, you do not need to compare cards with both face value and suit. So, you don't need the complete version of the `compare_two_cards` function. You can write whatever additional functions you need to compare hands according to the rules of blackjack.

You can have any additional `.c` and `.h` files you want, but none is necessary. In particular, you are not required to hand in any particular testing files.

Collaboration

NO collaboration is allowed on this assignment. Everyone must do their own individual work.

Program Turnin Procedure

In addition to the `.h` and `.c` files, you need to supply a Makefile that compiles the program files into an executable program named `blackjack`. The makefile must run successfully on unix1.

Submit all of your files as follows:

        handin gfisher 101_progs4_5 blackjack.h blackjack.c cards.h cards.c ... Makefile

where the "`...`" are any additional source files you may have. As noted above, no additional files are required.