

Programming Exam

CSCI 201L Fall 2017

10% of course grade

War Overview

For this exam, you will create the card game “war”. This game is very simple, but we will create a slightly modified version of the game to hopefully make your life a little easier programming it. Here are the rules we will follow.

Two players are in each game. The first player will place a bet between \$1 and \$100. The second player will either accept the bet (meaning that he also makes the same bet) or resigns from the game. If the player resigns from the game, the first player will need to wait for another player to join, at which point he will place a bet again.

If the second player accepts the bet, both players now have the same amount of money on the game. Each player will draw a card from the deck. Whoever has the higher card wins the bet of the other player. For purposes of this game, we will assume the following order of cards from lowest to highest:

2 3 4 5 6 7 8 9 10 J Q K A

In the event that the cards have the same value (i.e. both players draw a 4), the game goes to a war. This is where we will simplify the game. Each player will then draw one more card and compare again. This will continue until one of the cards is bigger than the other.

Each step will need to be acknowledged by each player before proceeding. In other words, I don’t want the entire game to be played without the user interacting with the program. Acknowledging could be as easy as having a button that needs to be clicked or pressing the “Enter” key at a prompt.

After a game finishes, the players will have to reconnect to play another game. The total amount of money a person has is not relevant to the game, so you can assume that a player always has at least \$100 at the beginning of a game.

Part 1 – Stand-alone Non-networked Version

For this part, you will create a stand-alone version of the above game that is *not* networked. In other words, you just need to allow two players who are sitting at the same computer to be able to play the game. Make sure that the player acknowledges when there is a war before just drawing the next cards (just by entering “Yes” in the example below). Here is a sample execution of the program with user input bolded.

```
Player 1 Bet - $10  
Player 2 Agree? Yes
```

```
Player 1 Card - J  
Player 2 Card - 8
```

```
Player 1 wins $10.  
Player 2 loses $10.
```

Thank you for playing war.

Here is another sample execution of the program with user input bolded.

```
Player 1 Bet - $10  
Player 2 Agree? No
```

Thank you for playing war.

Here is another sample execution of the program with user input bolded.

```
Player 1 Bet - $50  
Player 2 Agree? Yes
```

```
Player 1 Card - J  
Player 2 Card - J
```

```
Ready to continue? Yes
```

```
Player 1 Card - Q  
Player 2 Card - Q
```

```
Ready to continue? Yes
```

```
Player 1 Card - 2  
Player 2 Card - 3
```

```
Player 1 loses $50.  
Player 2 wins $50.
```

Thank you for playing war.

Part 2 – Stand-alone Networked Version

For this part, you will create a networked version of the above game. You will have two players able to play the game, either with a server in the middle or with the first player to start the game acting as a server – it’s your choice. Make sure you include a README file so the grader knows how to execute your program.

When there is a war, both players must acknowledge before proceeding with the game play. Here is a sample execution of the game, with player 1 on the left and player 2 on the right. The timing of the moves starts from the top and progresses down.

Player 1	Player 2
Joins the game through the stand-alone program	
	Joins the game through the stand-alone program
Bets \$10	
	Resigns from the game
	Joins the game through the stand-alone program
Bets \$10	
	Accepts \$10 bet
Draws J	
	Draws 8
Wins \$10	Loses \$10

Player 1	Player 2
Joins the game through the stand-alone program	
	Joins the game through the stand-alone program
Bets \$50	
	Accepts \$50 bet
Draws J	
	Draws J
Ready to continue?	Ready to continue?
yes	yes
Draws Q	
	Draws Q
Ready to continue?	Ready to continue?
yes	yes
Draws 2	
	Draws 3
Loses \$50	Wins \$50

Part 3 – Stand-alone and Web-based Networked Version

For this part, you will need to make one client accessible through a browser and the other one accessible through a stand-alone program. The users of the two programs should be able to play against each other. You can assume that you will always have one player accessing the game through the browser and the other player accessing the game through a stand-alone program. Name the page accessed through the browser `war.html`. (NOTE: You could have other pages, but that is the first one that needs to be accessed.)

Make sure you include a README describing everything the grader needs to know in order to start a game both from a web program and the stand-alone program.

The look of the game is up to you, but you need to make sure that each step is acknowledged by the player and clearly displayed. Here is a sample execution of the game, with player 1 on the left and player 2 on the right. The timing of the moves starts from the top and progresses down.

Player 1	Player 2
Joins the game through the web	
	Joins the game through the stand-alone program
Bets \$10	
	Resigns from the game
	Joins the game through the stand-alone program
Bets \$10	
	Accepts \$10 bet
Draws J	
	Draws 8
Wins \$10	Loses \$10

Player 1	Player 2
Joins the game through the stand-alone program	
	Joins the game through the web
Bets \$50	
	Accepts \$50 bet
Draws J	
	Draws J
Ready to continue?	Ready to continue?
yes	yes
Draws Q	
	Draws Q
Ready to continue?	Ready to continue?
yes	yes
Draws 2	
	Draws 3
Loses \$50	Wins \$50

Grading

% of Final Grade	Criteria
<i>Part 1 – 3.5%</i>	
0.5%	Player 1 can only enter a bet between 1 and 100.
0.5%	If player 2 does not agree with the bet, the game terminates.
0.5%	If player 2 agrees with the bet, one random card is dealt to each player.
1.0%	If one player has a higher card than the other, that player wins the game.
1.0%	If both players have the same card (i.e. there is a war), the player must acknowledge before dealing the next cards.
<i>Part 2 – 2.5%</i>	
0.5%	There is a README document explaining how to run the program.
1.0%	Two players are able to join the game.
0.5%	Acknowledgements by both players must be made when there is a war.
0.5%	Game logic is the same as in Part 1.
<i>Part 3 – 4.0%</i>	
0.25%	Web page is named war .html.
0.25%	There is a README document explaining how to run the program.
0.75%	Player 1 is the player who joins first, which could be either through the browser or the stand-alone program.
0.75%	Once player 2 joins, player 1 is prompted to enter a bet.
0.5%	If player 2 does not agree with the bet, the game terminates.
0.5%	If player 2 agrees with the bet, one random card is dealt to each player.
0.5%	Acknowledgements by both players must be made when there is a war.
0.5%	Game logic is the same as in Part 1.