

```

clc
clear

%prompt the user to enter a 0 if they do not know how to play memory and a
%1 if they do
playedBefore = input('Welcome to the Memory Card guessing game! Do you know
how to play? 0 for no, 1 for yes: ');

%print out an error message and prompt the user to reenter their response
%if they enter anything other than a 0 or 1
while(playedBefore ~= 0 && playedBefore ~= 1)
fprintf('Invalid input, you must enter 0 or 1!\n');
playedBefore = input('Welcome to the Memory Card guessing game! Do you know
how to play? 0 for no, 1 for yes: ');
end

%if the user enters a 0 (meaning they do not know how to play), then print
%out the rules
if(playedBefore == 0)
fprintf('\nMemory is a guessing game in which the goal is to eventually get
all of the cards that are shown/laid out to be face up.\nAt first, all of the
cards are laid face down. Then, the player chooses two cards to flip over. If
the cards are a match,\nthen they stay face up. If they aren?t a match, then
the cards go back to face down. The player continues to choose\ntwo cards
every turn until they get all of the cards to stay face up (by correctly
choosing all of the matches).\n');
pause(12);
end

%create the screen using the retro cards picture and the listed values for
%size and arrangement
my_scene = simpleGameEngine('retro_cards.png', 16, 16, 8, [255, 255, 255]);

%will be used for the creation of the card vectors
empty_sprite = 1;

%create a vector that holds all of the cards
card_spritesFace = 21:74;

%create a vector that will be used to hold the faces of the cards and
%create the final vector to be printed to the screen
card_displayFace = empty_sprite * ones(1,5);

%holds the index of the purple card back
card_spritesBack = 10;

%create a vector that will be used to hold the backs of the cards
card_displayBack = empty_sprite * ones(1,5);

%create a variable that will be used for incrementation and creation of the
%back of the cards vector
count = 1;

%set every card in the card_displayBack vector to be the purple card back
while count < 53

```

```

card_displayBack(count) = card_spritesBack(1);
count = count+1;
end

%create a new matrix that has 4 rows of 13 of the purple card backs
cardBack = [card_displayBack(1:13); card_displayBack(14:26);
card_displayBack(27:39); card_displayBack(40:52)];

%create a duplicate matrix that will be used to set wrong guesses back to
%the purple card back
cardBackDuplicate = cardBack;

%create a numGuesses variable that will increment every time the user
%guesses a pair or cards
numGuesses = 1;

%create a vector that contains the first 52 elements of the
%card_spritesFace vector (basically just get rid of the jokers)
for i = 1:52
card_displayFace(i) = card_spritesFace(i);
end

%create 2 vectors with a randomized order of the first 26 cards
boardFace = card_displayFace(randperm(26));
boardFaceTwo = card_displayFace(randperm(26));

%join the two randomized vectors
boardFaceFinal = [boardFace, boardFaceTwo];

%create a matrix that is 4 rows of 13, made up of the elements from the
%joint vector
cardFace = [boardFaceFinal(1:13); boardFaceFinal(14:26);
boardFaceFinal(27:39); boardFaceFinal(40:52)];

%print out the blank cards to the screen (it is the beginning of the game,
%and all of the cards will be turned over to start the game)
drawScene(my_scene, cardBack);

%drawScene(my_scene, cardFace);

%create a variable that will help end the loop/program
endLoop = 0;

%when the endLoop variable is equal to 52 (when all the cards have been
%flipped over) the loop will end and the program will essentially end
while endLoop ~= 52

%reset the endLoop variable
endLoop = 0;

%get the row and column of the user?s first click/guess
[row,col] = getMouseInput(my_scene);

%if the user?s guess had already been turned over, then output the error

```

```

%message and get their next click (repeat this until they guess a valid
%card)
while cardBack(row,col) ~= cardBackDuplicate(row,col)
msgbox('Invalid selection, this card has already been turned over! Select a
new card');
pause(2);
[row,col] = getMouseInput(my_scene);
end

%turn the first guessed card over
cardBack(row,col) = cardFace(row,col);

%set the firstGuess variable to the user?s first guessed card
firstGuess = cardFace(row,col);

%print out the new arrangement of cards (the user?s guessed card is now
%turned over)
drawScene(my_scene, cardBack);

%get the row and column of the user?s second click/guess
[r,c] = getMouseInput(my_scene);

%if the user?s guess had already been turned over, then output the error
%message and get their next click (repeat this until they guess a valid
%card)
while cardBack(r,c) ~= cardBackDuplicate(r,c)
msgbox('Invalid selection, this card has already been turned over! Select a
new card');
pause(2);
[r,c] = getMouseInput(my_scene);
end

%turn the second guessed card over
cardBack(r,c) = cardFace(r,c);

%set the secondGuess variable to the user?s second guessed card
secondGuess = cardFace(r,c);

%print out the new arrangement of cards (the user?s second guessed card is
also now
%turned over)
drawScene(my_scene, cardBack);

%if the user?s first guessed card matches their second guessed card, then
%print out a message saying that their guesses are matches and add to the
%amount of guesses
if cardBack(r,c) == cardBack(row,col)
matchMessage = msgbox('You found a match!');
pause(1.5);
numGuesses = numGuesses+1;

%if the first guessed card does not match the second guessed card, then
%print out a message saying that their guesses are not matches, turn
%their guessed cards back to the back of the cards, and add to the guesses
else

```

```

notMatch = msgbox('Not a match, try again!');
pause(1.5);
cardBack(r,c) = cardBackDuplicate(r,c);
cardBack(row,col) = cardBackDuplicate(row,col);
numGuesses = numGuesses+1;
end
%loop through the board and if the card that is being looked at is a
%card face, then add to endLoop (this checks to see if the
%game is over ? if all of the cards are turned over, then endLoop will
%increment to 52 and then the initial loop will end)
for i=1:4
for j=1:13
if cardBack(i, j) ~= cardBackDuplicate(i,j)
endLoop = endLoop+1;
end
end
end

end

%print out a message saying that the user has won the game
winMessage = msgbox('You won the game, congrats');

%print out the number of guess sets that it took the user to win the
%game
fprintf('Number of Guesses: %i\n',numGuesses);

```