**CMSC 203**

**Fall 2015**

**Assignment #2**

This assignment is to create a Java program that simulates a slot machine. Get the amount to be bet as input from the user, and then display the output of the randomly selected words from the spin. Depending on the matches of the output, display the prize amount won, total amount that has been bet so far, and the total amount won so far.

**Algorithm/Pseudocode**

\*Execute the program at least once, and then repeat as necessary if user wants to play again\*

do{

Display: title of application: “Slot Machine”

Display: “Enter your betting amount:”

Input: bettingAmount

while (bettingAmount is less than 1 or greater than 0)

Display “You must enter a bid between $1 and $20. Please re-enter:”

Randomly pick a number between 0-6, and depending on the number that is selected, assign it with one of the 7 words in the slot machine. If 0 (Cherry), 1 (Orange), 2 (Plum), 3 (Bell), 4 (Melon), 5 (Bar), 6 (Lucky 7).

d, then a number from 0-6, and if (0 is randomly selected from the range 0-6, then assign to it the word “Cherry”)

else if (1 is randomly selected from the range 0-6, then assign to it the word “Orange”)

else if (2 is randomly selected from the range 0-6, then assign to it the word “Plum”)

else if (3 is randomly selected from the range 0-6, then assign to it the word string “Bell”)

else if (4 is randomly selected from the range 0-6, then assign to it the word Melon”)

else if (5 is randomly selected from the range 0-6, then assign to it the word “Bar”)

else if (6 is randomly selected from the range 0-6, then assign to it the word “Lucky 7”)

Store the word in a String array named output.

Perform this 4 times in total, using a for loop, so there are 4 randomly chosen words placed in the variable output.

Display the words in the string[] output, showing each word followed by a space. This will represent the output of the slot machine images from a spin.

Calculate the prize amount won (pseudocode of the calculation is in the method documentation section)

Calculate total amount betted (pseudocode of the calculation is in the method documentation section)

Calculate total amount won (pseudocode of the calculation is in the method documentation section)

Display: “You win $[*prize amount won during the current spin*]”

Display: “The total amount you have bet so far is: [*the running total, of the amount of money bet so far*]”

Display: “The total amount you have won so far is: [*the running total, of the amount of money won so far*]”

Display: “Do you want to play again? (Y/N):”

Display: “Enter Y for yes or N for no: ”

Input: Y or N

Store the input in the variable named input

Store the first character in input into the variable playAgain

while( if the user decides to play the program again)

Repeat the program again. Otherwise stop.

**Classes:**

class SlotMachine

//description: Simulates a slot machine. Get the amount to be bet as input from the user, and then display the output of the randomly selected words from the spin. Depending on the matches of the output, display the prize amount won, total amount that has been bet so far, and the total amount won so far.

{

double bettingAmount;

char playAgain;

String results;

Final int OUTPUT\_SIZE = 4;

String [] output = new String [OUTPUT\_SIZE];

}

class SlotMoneyTracker

//description: Keeps track of all the money bet and won by the user, while playing the game. It also calculates the different payouts of the slot machine, depending on the possible winning spin results.

{

double totalAmountbet;

double totalAmountWon;

double prizeWon;

void totalBetted(double amount);

void totalWon (double amountWon);

double getTotalBetted();

double getTotalWon();

void results (double amount, String[] images);

double getPrizeMoney();

}

**Method Documentation:**

void main(String args[])

//precondition : none

//postcondition: display the 4 randomly selected words, the users prize won, total amount won, and total amount betted

//description:

1. Ask the user for amount to bet
2. Print out the random screen word matches
3. Print out the users prize won
4. Print out the users total amount betted so far
5. Print out the users total amount won so far

void totalBetted(double amount)

//precondition : amount betted must be > 1 & < 20

//postcondition: calculate and store the users bet amount to the total of the previous spins.

//description: calculate and store the total amount betted so far, while playing

totalAmountBetted += amount

void totalWon(double amountWon)

//precondition: none

//postcondition: calculate and store the users bet amount to the total of the previous spins

//description: calculate and store the total amount won so far, while playing

totalAmountWon += amountWon

double getTotalBetted()

//precondition: none

//postcondition: none

//description: return the total amount betted so far, while playing

double getTotalWon()

//precondition: none

//postcondition: none

//description: return the total amount won so far, while playing

void results(double amount, String[] images)

//precondition: amount betted must be > 1 & < 20, and the 4 words representing the images must be randomly selected

//postcondition: calculates the prize payouts for the slot machines, depending on the different random matches of the words that may occur

//description: Store the prize money won

d, then a number from 0-6, and if (all four words are the same)

if (all four of the words equal “Lucky 7”)

Display: "You hit the grand prize! You got all 4 words that say Lucky 7, and have won 8 times the amount you bet plus a $1,000 bonus."

Calculate prize amount, which is 8 times the amount betted plus $1,000 for hitting the grand prize.

else

Display: "You got all 4 words that match, and have won 8 times the amount you bet."

Calculate the prize amount, which is 8 times the amount betted dlmvfdlcm

sx cdelse if (three of the words are matching in a row)

Display: "You got 3 words that match, and have won 3 times the amount you bet."

Calculate the prize amount, which is 3 times the amount betted dfdcdcdvcv

else if (two of the images match and are in a row)

if (there are two pairs that match in a row)

Display: "You got 2 pairs of words that match, and have won double the amount you bet."

Calculate prize amount, which is double the amount betted

else

Display: “You got 1 pair of words that match, and have won back the amount you bet.

Calculate prize amount, which is just returning the amount bet at start of spin.

else

Display: "You got no match. You have lost your bet."

Calculate the prize amount, 0 times the amount betted dlmvfdlcm

void getPrizeMoney()

//precondition: none

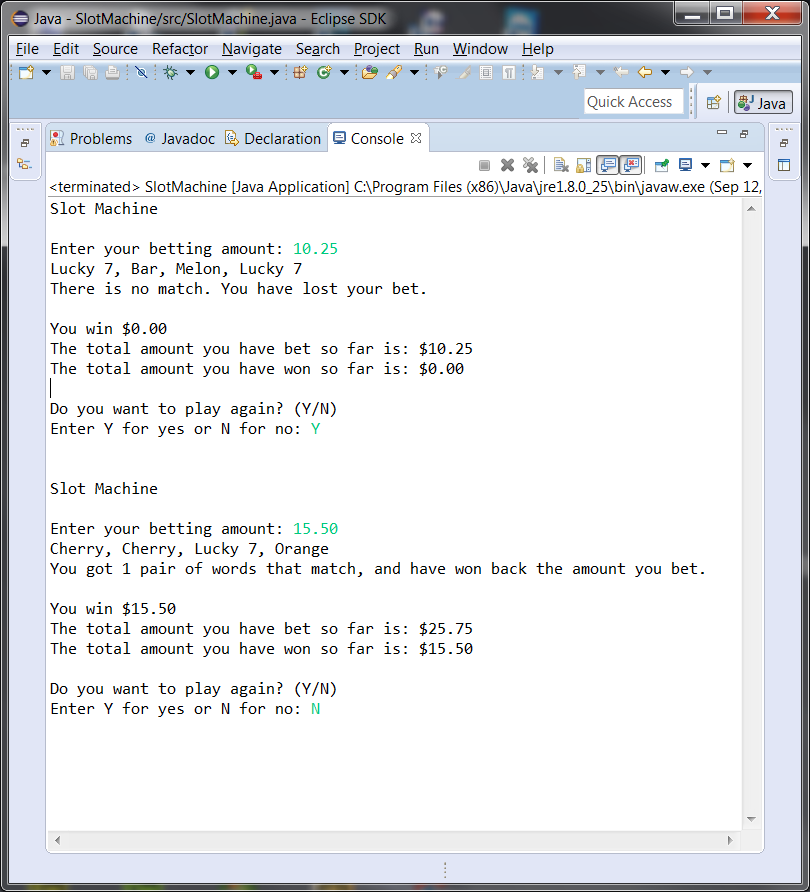
//postcondition: none

//description: prize money is returned

**Test table:**

|  |  |  |
| --- | --- | --- |
| Input | Expected Output | Program runs correctly |
| Bet amount:1 | If (4 different words):  Lucky 7, Melon, Plum, Cherry  Prize Won: 0  Total Won: 0  Total Betted: 1  Display: “Do you want to play again? (Y/N)” |  |
| Bet amount: 3 | If (only 1 pair of matching words in a row):  Melon, Melon, Plum, Cherry  Prize Won: 3  Total Won: 3  Total Betted: 3  Display: “Do you want to play again? (Y/N)” |  |
| Bet amount: 5.25 | If (two pairs of different matching words, in a row):  Melon, Melon, Plum, Plum  Prize Won: 10.50  Total Won: 10.50  Total Betted: 5.25  Display: “Do you want to play again? (Y/N)” |  |
| Bet amount:15 | If (3 of the words match in a row):  Plum, Plum, Plum, Cherry  Prize Won: 45  Total Won: 45  Total Betted: 15  Display: “Do you want to play again? (Y/N)” |  |
| Bet amount:20 | If (all four words are matching, except for Lucky 7):  Melon, Melon, Melon, Melon  Prize Won: 160  Total Won: 160  Total Betted: 20  Display: “Do you want to play again (Y/N)” |  |
| Bet amount: 3.50 | If (all four words are Lucky 7):  Lucky 7, Lucky 7, Lucky 7, Lucky 7  Prize Won: 1028  Total Won: 1028  Total Betted: 3.50  Display: “Do you want to play again (Y/N)” |  |
| Bet amount: 50 | Display: “You must enter a bid amount between $1and $20. Please re-enter:” |  |
| Bet amount: 0 | Display: “You must enter a bid amount between $1and $20. Please re-enter:” |  |

**Screenshots:**



**Assumptions (if any)**

1. There must be at a minimum, one pair of words appearing in a row that match, to win some prize money.

2. If any word does not appear more than once in the output and is not display in a row at any point, then it cannot result in a winning spin.

**Lessons Learned:**

**In 3+ paragraphs, highlight your lessons learned and learning experience from working on this project. How did you do? What have you learned? What did you struggle with? How will you approach your next project differently?**

While reading the assigned chapter’s in the textbook for module 2 and 3 , I found many of the concepts as review from the CMSC140 class I took during the summer, and it was just a matter of applying it into Java. I found this assignment 2, a very helpful project to refresh my memory on the different kinds of loops that can be used, and how the for loop, while loop, and do-while loop all can be used to execute different types of scenarios, depending on the specific situations.

I learned many new things while working on this assignment, such as understanding and using the Random class, to generate random numbers from any range that you want too. I also learned how to create classes and methods in Java, and how to create objects from those classes to use in my program, in order to perform certain functions that were needed. For this assignment, I created a new class called SlotMoneyTrackers, and created methods in that class for keeping track of the different payouts that a user might receive, depending on the randomly selected words chosen. There were other methods in the class that I wrote to help with keeping track of the total amount won and betted. I found it really helpful to create a new class that performed many of the calculations I needed for the slot machine execution, that way the main method in my SlotMachine.java file wasn’t too congested.

I felt like I did pretty well in this assignment, and understood most all of the concepts that were needed to run this program correctly. I initially had trouble on how to properly display the 4 randomly chosen words, which mimicked the slot Machine results. After thinking about it for a while, I decided to assign each of the 7 words a number from 0-6, and then depending on the randomly chosen number, assigning it to the corresponding word. Each time the loop ran, it stored the word into an array of size 4, to represent the 4 images in a slot machine output.

I will approach my next assignment the same way I did this current assignment and the previous one. I felt like I managed my time wisely, and was able to finish up everything in a timely manner. Overall, I found writing the actual code for this program as pretty straight forward, however, doing all the other documentation and write-up portion of the assignment was very time consuming.