Planning and Algorithm: Due: November 16, 2018

1. The theme for your phrases is: Memes!

2. ALL Phrases that you will include in your game (min 25 phrases).

Phrase No.	Phrase
1	CHURN DOWN FOR WHAT
2	WHAT IF I TOLD YOU IT REALLY WAS BUTTER?
3	WHO YA GONNA CALL? GOATS BUTTERS!
4	BUTTER SAFE THAN SORRY
5	YOU BUTTER BELIEVE IT
6	I DON'T ALWAYS RUN, BUT WHEN I DO, I RUN FOR GRADE NINE REP
7	NO U
8	ONE DOES NOT SIMPLY PUT MEMES FOR A SCHOOL ASSIGNMENT
9	CASH ME OUSSIDE HOWBOW DAH?
10	YOU DON'T SAY?
11	RICK ROLL
12	BUT THAT'S NONE OF MY BUSINESS
13	YOU MAD BRO?
14	DOGE!
15	I DON'T KNOW WHO GAVE MY CHILD A WHISTLE, BUT I WILL FIND YOU AND I WILL KILL YOU
16	WHAT LIFE? DAB! FORTNITE
17	ORANGE JUSTICE
18	ASIANS: A+ ON MS HUNG'S MATH TEST, WON'T HAVE TO SLEEP IN DOG HOUSE ANYMORE
19	F RESH AVA CA DO
20	DONALD TRUMP
21	THE FLOOR IS LAVA!
22	SALT BAE
23	IDIOT SANDWICH
24	WHERE IS THE LAMB SAUCE
25	TWO PLUS TWO IS FOUR, MINUS ONE THAT'S THREE QUICK MATH

3. Complete an **algorithm** for the program that you are going to design, code, test and submit. Please be sure to consider all games rules and features that you must include in your program.

Detailed Algorithm:

- import Scanner
- create class
- create main method
    - create scanner
    - create string option
    - initialize the phrase array with the premade phrases
    - create do loop
        - ask user for name and print their name
        - set a counter for number of rounds, players final score and computers final score
        - create while loop for 3 rounds
            - create counter for number of turns and the current scores for the computer and player
            - set up the game board
            - initialize the alphabet and game board
            - print the current round
            - who ever guesses the phrase keeps their winnings if their winnings are below 1000, they get 1000, else they get whatever they got
            - turns are decided if the counter of turn is divisible by 2, if its is, its the computer's turn, if not, it's the players turn
            - each turn, print what letters are left
            - create a while loop that runs until the round is over
                - if it's the player's turn
                    - create a while loop that runs until there's a valid action used
                        - get the spin value
                        - print out the options teh player has
                        - if they got bankrupt skip their turn at set their money to 0
                        - if they lost their turn, skip their turn
                        - print 1, 2, 3 for guess a letter, buy vowels and guessing the phrase respectively
                        - ask the user for choice
                        - if input is 1
                            - ask user to enter a letter
                            - after you get a valid input
                                - update the board, the alphabet board
                            - break out of the while loop

- if input is 2
    - if they can buy vowels
    - they can buy until they run out of money
    - update the board and the alphabet board
    - after their done break out of the while loop
    - if they can't buy vowels, continue the while loop
- if input is 3
    - ask user to input a phrase
    - if they guess correctly end the round
    - else break out of the while loop

- if its computers turn
    - give a random choice for the computer
        - if its the first choice
            - guess a valid random letter
            - update board and alphabet board
            - print their guesses
        - if its the second choice
            - guess a valid vowel and buy a random number amount of times and if they can buy vowels
            - print their actions
        - if its the third choice
            - have a random percentage from 1 - 100
            - if its 100, computer wins and ends the round
            - else continue the round
    - print the current board and currect money for both computer and player
- add 1 to the current round

- if computer has more money than player
    - print computer wins
- if player has more money than computer
    - print player wins
- if its a tie
    - print tie
- ask user if they want to play again
- continue do loop if they do