**Question #1: Examine the pseudocode that follows the introductory comments, then find and correct all the errors.**

**DEBUG-01.txt**

// This pseudocode segment is intended to describe

// computing the price of an item on sale for 20% off

start

input origPrice

set discount = price \* 0.20

set finalPrice = origPrice - discount

output final

Stop

**Answer:(Find and correct all the errors)**

Price should be changed to “origPrice”

Final should be changed to “finalPrice”

**DEBUG-02.txt**

// This pseudocode segment is intended to describe computing

// the number of miles per gallon you get with your automobile.

start

input milesTraveled

input gallonsOfGasUsed

set milesPerGallon = milesTraveled + gallons

output milesPerGallon

start

**Answer:(Find and correct all the errors)**

milesPerGallon equation should use / not +

“gallons” should be “gallonsOfGasUsed”

Start should be stop.

**DEBUG-03.txt.**

// This pseudocode segment is intended to describe

// computing the per day cost of your rent

// in a 30-day month

start

output rent

set costPerDay = rent + 30

input cost

stop

**Answer:(Find and correct all the errors)**

“Output” should be “input”

“input” should be “output”

“cost” should be “costPerDay”

Rent should be multiplied by 30

**Question #2: Write pseudocode to represent the logic of a program that allows the user to enter a value. The program divides the value by 2 and outputs the result.**

**Answer:**

Start

Input number

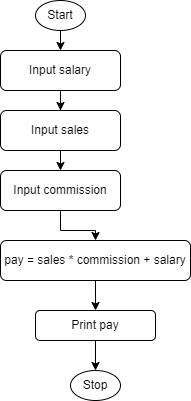
Ans = number / 2

Output Ans

stop

**Question #3: Draw a flowchart (use Draw.io) to represent the logic of a program that allows the user to enter values for a salesperson's base salary, total sales, and commission rate. The program calculates and outputs the salesperson's pay by multiplying the total sales by the commission rate and then adding the base salary.**

**Answer:**

****

**Question #4: The pseudocode below represents the logic of a guessing game program. The player inputs a number between 1 and 10, and tries to guess the 'magic number', which is 6 in our program. After each guess, the player is told if the guess is too high or too low. The process continues until the player guesses the correct number.**

start

input guess

while guess is not 6

if guess is greater than 6

output "Guess a number lower than the previous guess."

else

output "Guess a number higher than the previous guess."

endif

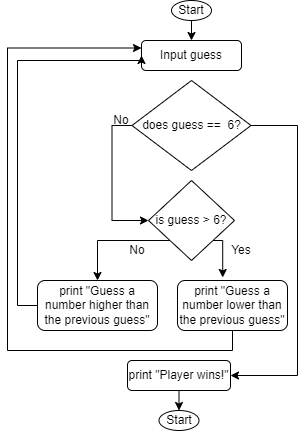
endwhile

output "Player wins!"

stop

**Answer the following questions based on the pseudocode:**

**i. Draw a flowchart (use Draw.io) for this logic**.

**Answer:**

**ii. Test the logic by stepping through the flowchart using the following test data: 5, 7, 3, 9, 6. Record the output of the program after each number is entered. Remember to use the flowchart to test the logic, don't just do it in your head automatically!**

**Answer:**

5: "Guess a number higher than the previous guess"

7: "Guess a number lower than the previous guess"

3: "Guess a number higher than the previous guess"

9: "Guess a number lower than the previous guess"

6: “Player Wins!”

**iii. What do you think would happen if non-numeric input data is entered, for example, "hello"?**

**Answer: Error will occur when code reaches the decision “is guess > 6?”**