Question 1:

Write a Python program that asks for a string as input and prints a version with each word capitalized and a full stop added at the end (if not already present). Constraints: Use character-by-character processing and string concatenation, avoiding slicing and string methods like ~~upper()~~, title() and endswith().

Question 2:

Write a Python program that asks for a string as input and determine whether it's a palindrome (reads the same backward as forward) or not.

Constraints: Use only individual character comparisons and string reversal techniques, avoiding slicing and built-in functions like reversed().

Question 3:

A video store has multiple employees who worked different hours last week. Both employees earn a base rate of $14.50 per hour for the first 40 hours. Overtime pay is calculated differently: for the first five hours exceeding 40 hours, they receive time-and-a-half pay (1.5 times the base rate) and for any hours beyond that, they get double-time pay (twice the base rate). Taking a 28% tax rate into account, write a Python program that prompts the user to enter their hours worked. The program should then calculate and display the employee's gross pay (total earnings before taxes), taxes withheld, and net pay (gross pay minus taxes) with clear labels and comments throughout the code.

```
Enter the number of hours worked: 48

**Employee Pay Summary**

Gross Pay:  $775.75
Taxes Withheld (28%):  $217.21
Net Pay:  $558.54
Do you have another employee (yes/no)?
```

Question 4:

Movie theaters often have different ticket prices depending on factors like age, day of the week, and showtime. Here's what you need to consider:

- Adult tickets: Cost $12.50 on weekdays and $15.00 on weekends (Friday-Saturday) and holidays.
- Children tickets (under 12): Cost $8.00 on weekdays and $10.00 on weekends and holidays.
- Senior tickets (over 65): Cost $9.00 on weekdays and $11.50 on weekends and holidays.

Write a Python program that does the following:

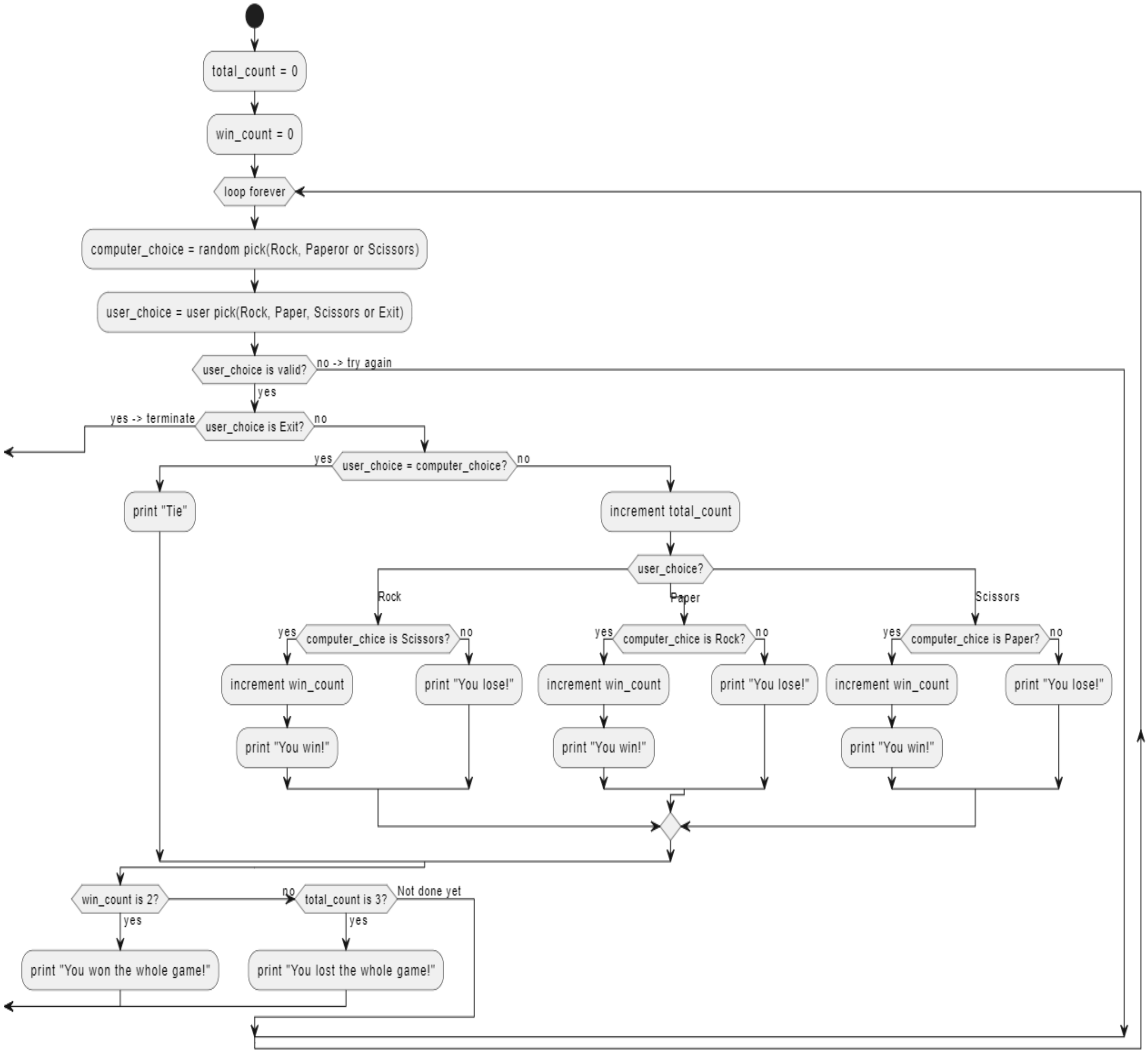
1. Asks the user for:
    a. The number of adult tickets purchased.
    b. The number of child tickets purchased.
    c. The number of senior tickets purchased.
2. Whether the movie is being shown on a weekday, weekend (Friday-Saturday), or holiday.
3. Calculates the total ticket price based on the number of tickets of each type, their corresponding prices, and the day of the show.
4. Offers the user a discount if they are purchasing 5 or more total tickets (10% off).
5. Prints a detailed receipt showing the number of tickets purchased for each category, the price per ticket, and the final total price after applying any discount.

Sample Program Execution:

```
Number of adult tickets: 2
Number of child tickets: 1
Number of senior tickets: 0
Is the movie showing on a weekday (w), weekend (e), or holiday (h)? w
Program Output:

Movie Ticket Receipt

Adult Tickets (x2): $12.50 each
Child Tickets (x1):  $8.00 each
Senior Tickets (x0):  $0.00 each

Subtotal: $33.00
Discount (5+ tickets): 10%
Discount amount: $0.00

Total: $33.00

Thank you for coming to the movies!
```

Question 5:

Write a Python program that plays a rock-paper-scissors game against the user. The program should randomly generate its choice of rock, paper, or scissors, and then prompt the user for their choice. The program should determine the winner based on the rules of the game (rock beats scissors, scissors beats paper, paper beats rock) and display the result to the user. Use random number generation, conditional statements, and input/output in your code. The program flow should adhere to the provided flowchart. However, you have the freedom to modify the output text to include more informative details



Question 1:

```
def capitalize_and_punctuate(text):
  """Capitalizes each word and adds a full stop at the end (if missing).

  Args:
      text: The input string.

  Returns:
      A string with each word capitalized and a full stop at the end.
  """
  is_first_word = True
  result = ""
  for char in text:
    if char.isalpha():
      if is_first_word:
        result += char.upper()
        is_first_word = False
      else:
        result += char.lower()
    else:
      result += char
      # Check for punctuation at the end (excluding spaces)
      if char not in " \t\n" and is_first_word:
        is_first_word = True
  # Add full stop if missing
  if not result.endswith("."):
    result += "."
  return result

# Get user input
text = input("Enter a string: ")

# Print the capitalized and punctuated string
print(capitalize_and_punctuate(text))
```

Question 2:

```
def is_palindrome(text):
  """Checks if a string is a palindrome.

  Args:
      text: The input string.

  Returns:
      True if the string is a palindrome, False otherwise.
  """
  left = 0
  right = len(text) - 1
  while left < right:
    # Compare characters ignoring case
    if not text[left].lower() == text[right].lower():
      return False
    left += 1
    right -= 1
  return True

# Test cases
text1 = "Racecar"
text2 = "Hello"
text3 = "A man, a plan, a canal: Panama"  # Punctuation ignored

print(f"{text1} is a palindrome: {is_palindrome(text1)}")
print(f"{text2} is a palindrome: {is_palindrome(text2)}")
print(f"{text3} is a palindrome: {is_palindrome(text3)}")
```

Question 3:

```
import random

# Define options for the game
options = ["Rock", "Paper", "Scissors"]

# Game loop
while True:
  # Generate computer's choice
  computer_choice = random.choice(options)

  # Get user's choice
  user_choice = input("Choose Rock, Paper, or Scissors (or 'q' to quit): ").capitalize()

  # Check for quit option
  if user_choice == 'Q':
    break

  # Validate user input
  if user_choice not in options:
    print("Invalid choice. Please try again.")
    continue

  # Determine winner
  if user_choice == computer_choice:
    print(f"It's a tie! Both players chose {user_choice}.")
  elif (user_choice == "Rock" and computer_choice == "Scissors") or \
       (user_choice == "Paper" and computer_choice == "Rock") or \
       (user_choice == "Scissors" and computer_choice == "Paper"):
    print(f"You win! {user_choice} beats {computer_choice}.")
  else:
    print(f"You lose! {computer_choice} beats {user_choice}.")

  # Play again prompt
  play_again = input("Play again? (y/n): ").lower()
  if play_again != 'y':
    break

print("Thanks for playing!")
```

```
def calculate_ticket_price(adult_tickets, child_tickets, senior_tickets, day):
  """
  Calculates the total ticket price based on ticket types, quantities, and day of show.

  Args:
      adult_tickets (int): Number of adult tickets purchased.
      child_tickets (int): Number of child tickets purchased.
      senior_tickets (int): Number of senior tickets purchased.
      day (str): Day of the show (w: weekday, e: weekend/holiday).

  Returns:
      float: Total ticket price after applying discount (if applicable).
  """
  # Define ticket prices based on day
  if day.lower() == 'w':
    adult_price = 12.50
    child_price = 8.00
    senior_price = 9.00
  else:
    adult_price = 15.00
    child_price = 10.00
    senior_price = 11.50

  # Calculate subtotal
  subtotal = (adult_tickets * adult_price) + (child_tickets * child_price) + (senior_tickets * senior_price)

  # Apply discount for 5+ tickets
  if adult_tickets + child_tickets + senior_tickets >= 5:
    discount = subtotal * 0.1  # 10% discount
    subtotal -= discount

  return subtotal

# Get user input
adult_tickets = int(input("Number of adult tickets: "))
child_tickets = int(input("Number of child tickets: "))
senior_tickets = int(input("Number of senior tickets: "))
day = input("Is the movie showing on a weekday (w), weekend (e), or holiday (h)? ")

# Calculate total price
total_price = calculate_ticket_price(adult_tickets, child_tickets, senior_tickets, day)

# Print receipt
print("\nMovie Ticket Receipt")
print(f"Adult Tickets (x{adult_tickets}): ${adult_price:.2f} each")
print(f"Child Tickets (x{child_tickets}): ${child_price:.2f} each")
print(f"Senior Tickets (x{senior_tickets}): ${senior_price:.2f} each")
print(f"\nSubtotal: ${subtotal:.2f}")

if adult_tickets + child_tickets + senior_tickets >= 5:
  print(f"Discount (5+ tickets): 10%")
  print(f"Discount amount: ${discount:.2f}")

print(f"\nTotal: ${total_price:.2f}")
print("\nThank you for coming to the movies!")
```

Question 1:

Write a Python program that generates a random number between 1 and 100. The user has to guess the number, and the program provides hints (higher or lower) until the correct number is guessed. Using random number generation, loops (while), conditional statements, input/output

Question 2:

Write a Python program that asks for a string as input and prints a version with each word capitalized and a full stop added at the end (if not already present). Constraints: Use character-by-character processing and string concatenation, avoiding slicing and string methods like upper(), title() and endswith().

Question 3:

Write a Python program that asks for a string as input and determine whether it's a palindrome (reads the same backward as forward) or not.

Constraints: Use only individual character comparisons and string reversal techniques, avoiding slicing and built-in functions like reversed().

Question 4:

A local bakery employs three bakers with varying hourly rates and overtime rules:

- Baker 1:
  - $15.00 per hour for regular hours
  - Time-and-a-half for hours over 35
- Baker 2:
  - $16.25 per hour for regular hours
  - Double-time for hours over 40
- Baker 3:
  - $17.75 per hour for regular hours
  - Time-and-a-half for hours from 41 up to 45
  - Double-time for hours over 45

Write a Python program that does the following:

1. Asks for the number of hours worked by each baker last week.
2. Stores this information in separate variables.
3. Determines each baker's gross pay based on their respective pay rates and overtime rules. Consider the different thresholds for overtime pay.
4. Applies a 30% tax rate to each baker's gross pay.

5. Subtracts the calculated taxes from each baker's gross pay to determine their net pay.
6. Displays each baker's hours worked, gross pay, taxes, and net pay in a clear and organized format, using appropriate labels.

Question 5:

A hotel offers various room types with different pricing structures:

- Standard Room:
  - $155 per night for single occupancy
  - $160 per night for double occupancy
  - $165 per night for triple or more occupancy
- Deluxe Room:
  - $195 per night for single or double occupancy
  - $210 per night for triple occupancy
  - $225 per night for quadruple or more occupancy
- Additional charges:
  - 12% tax on room charges
  - $15 per person per day for breakfast (optional)
  - $25 per day for parking (optional)
- Discounts:
  - 10% off for stays longer than 5 nights

Write a Python program that:

1. Prompts for user input:
   a. Room type (Standard or Deluxe)
   b. Number of nights
   c. Number of people
   d. Meal charges (if any)
2. Calculates room charges: Uses appropriate conditional statements to determine the correct rates based on room type and occupancy.
3. Calculates additional charges: Includes taxes, breakfast charges (if applicable), and other service charges.
4. Generates a detailed bill: Clearly lists each charge with a descriptive label (e.g., room rate, tax, breakfast, parking).
5. Provides a subtotal of charges before tax.
6. Displays the total amount due, including all taxes and fees.