**ASSINGMENT 1**

|  |
| --- |
| **NAME HAJJI MYDEEN S**  **CLASS MECH AIML**  **COURSE PROGRAME SOLVING CODE 21CSS101J**  **REG NO RA2411054040009** |

**Write an algorithm, pseudocode, and flowchart for calculating the sum of all even numbers from 1 to n, where n is entered by the user.**

**Algorithm**

1. Start.
2. Prompt the user to enter a positive integer n.
3. Initialize a variable sum to 0.
4. Loop through all numbers from 1 to n.

oIf the number is even (i.e., divisible by 2), add it to sum.

1. After the loop ends, print the value of sum.
2. End.

**Pseudocode**

START

PROMPT "Enter a positive integer (n): "

READ n

SET sum = 0

FOR i FROM 1 TO n DO IF i MOD 2 = 0 THEN sum = sum + i

ENDIF

ENDFOR

PRINT "The sum of all even numbers from 1 to", n, "is:", sum

END

1. **Factorial Using Recursion**

**Design a pseudocode, algorithm, and flowchart to calculate the factorial of a number using recursion.**

**Algorithm**

* 1. Start.
  2. Define a recursive function factorial:
     + If the input number n is 0 or 1, return 1 (base case).
     + Otherwise, return n \* factorial(n-1)

(recursive case).

1. Prompt the user to input a positive integer n.
2. Call the factorial function with the input value n.
3. Display the result.
4. End.

**Pseudocode**

FUNCTION factorial(n)

IF n = 0 OR n = 1 THEN

RETURN 1

ELSE

RETURN n \* factorial(n - 1)

ENDIF

END FUNCTION

START

PROMPT "Enter a positive integer: "

READ n

SET result = factorial(n)

PRINT "The factorial of", n, "is:", result END

**3. Prime Number Check**

**Write the algorithm, pseudocode, and flowchart to check whether a given number is a prime number or not.**

**Algorithm**

1. Start.
2. Prompt the user to enter a positive integer n.
3. If n is less than or equal to 1, it is not a prime number.
4. For numbers greater than 1:
   * Initialize a loop to check divisors from 2 to n\sqrt{n}n

.

* + If n is divisible by any number in this range, it is not a prime number.

1. If no divisors are found, n is a prime number.
2. Display the result.
3. End.

**Pseudocode**

START

PROMPT "Enter a positive integer: "

READ n

IF n <= 1 THEN

PRINT n, "is not a prime number."

EXIT

ENDIF

SET isPrime = TRUE

FOR i FROM 2 TO √n DO

IF n MOD i = 0 THEN

SET isPrime = FALSE

BREAK

ENDIF

ENDFOR

IF isPrime THEN

PRINT n, "is a prime number."

ELSE

PRINT n, "is not a prime number."

ENDIF

END

**4. Finding the Largest Number in an Array**

**Create an algorithm, pseudocode, and flowchart for finding the largest number in an array of n elements provided by the user.**

**Algorithm**

1. Start.
2. Prompt the user to enter the size of the array n.
3. Prompt the user to enter n elements of the array.
4. Initialize a variable largest with the value of the first element of the array.
5. Iterate through the array starting from the second element:

oIf the current element is greater than largest, update largest to the current element.

1. After the loop, largest contains the largest value in the array.
2. Display the largest number.
3. End.

**Pseudocode**

START

PROMPT "Enter the number of elements in the array (n): "

READ n

DECLARE array[n]

PROMPT "Enter the elements of the array: "

FOR i FROM 1 TO n DO

READ array[i]

ENDFOR

SET largest = array[1]

FOR i FROM 2 TO n DO

IF array[i] > largest THEN

SET largest = array[i]

ENDIF

ENDFOR

PRINT "The largest number in the array is:", largest

END

**5. Simple Interest Calculation**

**Write an algorithm, pseudocode, and flowchart to calculate simple interest based on the formula SI = (P × R × T) / 100, where P is the principal amount, R is the rate of interest, and T is the time in years.**

**Algorithm**

1. Start.
2. Prompt the user to enter the principal amount P.
3. Prompt the user to enter the rate of interest R.
4. Prompt the user to enter the time in years T.
5. Calculate the simple interest using the formula:

SI=P×R×T100\text{SI} = \frac{P \times R \times

T}{100}SI=100P×R×T

1. Display the calculated simple interest SI.
2. End.

**Pseudocode**

START

PROMPT "Enter the principal amount (P): "

READ P

PROMPT "Enter the rate of interest (R): "

READ R

PROMPT "Enter the time in years (T): "

READ T

SET SI = (P \* R \* T) / 100

PRINT "The Simple Interest (SI) is:", SI

END

**6. Generate Fibonacci Series**

**Design the algorithm, pseudocode, and flowchart to generate the first n terms of the Fibonacci sequence**.

**Algorithm**

1. Start.
2. Prompt the user to enter the number of terms n to generate in the Fibonacci series.
3. If n is less than or equal to 0, display an error message and stop.
4. Initialize two variables: oa = 0 (first term).
   * b = 1 (second term).
5. If n >= 1, display a.
6. If n >= 2, display b.
7. Use a loop to calculate and display the next terms of the Fibonacci sequence until n terms are generated: o Compute next = a + b. o Update a = b and b = next.
   * Display next.
8. End.

**Pseudocode**

START

PROMPT "Enter the number of terms (n): "

READ n

IF n <= 0 THEN

PRINT "Invalid input. Enter a positive integer."

EXIT

ENDIF

SET a = 0

SET b = 1

IF n >= 1 THEN

PRINT a

ENDIF

IF n >= 2 THEN

PRINT b

ENDIF

FOR i FROM 3 TO n DO

SET next = a + b

PRINT next

SET a = b

SET b = next

ENDFOR

END

**7. Number Guessing Game**

**Write an algorithm, pseudocode, and flowchart for a number guessing game where the program generates a random number between 1 and 100, and the user has to guess the number with hints provided (e.g., higher/lower).**

**Algorithm**

1. Start.
2. Generate a random number target between 1 and 100.
3. Initialize a flag variable guessed = False.
4. Repeat until guessed is True: oPrompt the user to enter a guess. oIf the guess is equal to the target:
   * Display "Congratulations! You guessed the number."
   * Set guessed = True. oIf the guess is less than the target:
   * Display "Too low, try again." oIf the guess is greater than the target:
   * Display "Too high, try again."
5. End.

**Pseudocode**

START

SET target = RANDOM(1, 100)

SET guessed = FALSE

WHILE guessed = FALSE DO

PROMPT "Enter your guess (1-100): "

READ guess

IF guess = target THEN

PRINT "Congratulations! You guessed

the number."

SET guessed = TRUE

ELSE IF guess < target THEN

PRINT "Too low, try again."

ELSE

PRINT "Too high, try again."

ENDIF

ENDWHILE

END

1. **Temperature Conversion**

**Develop an algorithm, pseudocode, and flowchart for converting a temperature value from Celsius to Fahrenheit using the formula F = (C × 9/5) + 32.**

**Algorithm**

* 1. Start.
  2. Prompt the user to enter the temperature in Celsius (C).
  3. Calculate the temperature in Fahrenheit using the formula:

F=(C×95)+32F = (C \times \frac{9}{5}) + 32F=(C×59)+32

* 1. Display the converted temperature in Fahrenheit (F).
  2. End.

**Pseudocode**

START

PROMPT "Enter the temperature in Celsius

(C): "

READ C

SET F = (C \* 9 / 5) + 32

PRINT "The temperature in Fahrenheit (F) is:", F

END

1. **Count Vowels in a String**

**Write the algorithm, pseudocode, and flowchart to count the number of vowels in a given string input by the user.**

**Algorithm**

* 1. Start.
  2. Prompt the user to enter a string.
  3. Initialize a counter variable vowelCount = 0.
  4. Loop through each character in the string:

oConvert the character to lowercase. oIf the character is a vowel ('a', 'e', 'i',

'o', 'u'), increment vowelCount by 1.

* 1. Display the total number of vowels (vowelCount).
  2. End.

**Pseudocode**

START

PROMPT "Enter a string: "

READ inputString

SET vowelCount = 0

SET vowels = "aeiou"

FOR EACH character IN inputString DO

SET character = LOWERCASE(character)

IF character IN vowels THEN

SET vowelCount = vowelCount + 1

ENDIF

ENDFOR

PRINT "The number of vowels in the string

is:", vowelCount END

**10. Find the GCD of Two Numbers**

**Create an algorithm, pseudocode, and flowchart to find the greatest common divisor (GCD) of two numbers using the Euclidean algorithm.**

**Algorithm**

1. **Input:** Two positive integers, a and b.
2. **Step 1:** Divide a by b and find the remainder r, i.e., r = a % b.
3. **Step 2:** If r == 0, then the GCD is b. Stop.
4. **Step 3:** Otherwise, set a = b and b = r, and go back to Step 1.
5. **Output:** The value of b when r == 0 is the GCD.

**Pseudocode:**

Function GCD(a, b): While b != 0: r = a % b

* 1. = b
  2. = r End While

Return a

End Function