q,

Let's perform the following exercises:

1. Identify the a)input, b)process, and c)output phases in the following tasks:

* **Calculating the sum of two numbers using a calculator**:

Answer:

1. Input – The procedure of typing the numbers and the ‘+’ operator on the calculator interface.
2. Process – The act made by the calculator to use it’s processor for adding the two input numbers to achieve a new resultant number during this process.
3. Output – The process of presenting or printing the result on its demonstrable external interface for the end user.

* **Obtaining the juice of a fruit using a juicer machine**:

Answer:

1. Input – The pieces of fruit that goes inside the juicer.
2. Process – The machine being put on and processing the fruits with the right regulated button to create the juice of the fruits.
3. Output – The processed juice being poured out of the juicer in a drinking glass.

* **Creating knitwear**:

Answer:

1. Input – yarn is the input to create a textile or fabric and different types of yarns (fibre type, texture, and twist), needle sizes, and stitch types may be used to achieve knittable fabrics.
2. Process – This requires the process of Knitting! It may be done by hand or by using a machine. Knitting creates multiple loops of yarn, called stitches, in a line or tube.
3. Output – knitted human body wearable fabrics/textiles with different attributes like colour, texture, weight, heat retention, look, water resistance, and/or integrity.

* **Baking bread**:

Answer:

1. Input – it will include wheat, chemicals and flour.
2. Process – It has have the Baking procedure like – looking at controlled mixing of the dough, resting, loaf moulding, proving and baking for about 3.33 minutes.
3. Output – Baked bread loves as output from the oven.

* **Washing clothes in a washing machine**:

Answer:

1. Input – includes clothes that are dirty, detergent powder, softeners, temperature setup, type of cloth wash required, roller speed setup, electricity, timer, and automation inputs.
2. Process – the process of controlled washing as per desired setup for the dirty clothes.
3. Output – Machine cleaned and softened up clothes.
4. is a special program that converts the programming language into the machine language, at one go.

* Input-Process-Output
* Central Processing Unit
* Compiler «----

**Answer:**

* Compiler is the Correct Answer!

1. Discuss an algorithm to withdraw a required amount from an ATM.

Answer:

The Sequence of Steps for withdrawing money from ATM:

1. Insert ATM card into the ATM card slot.
2. Swipe out the card back for yourself.
3. Select your language.
4. Select the Transaction type – Withdraw Money.
5. Select the Account type, say – Savings.
6. Enter your ATM Security PIN.
7. Enter Amount (for withdraw)
8. Press Correct, Enter or Yes for the transaction to get processed by the ATM.
9. Collect cash from the cash outlet.
10. Select an option for getting a paper receipt of the current transaction.
11. Select ‘No’ for Another Transaction.
12. End your Transaction Session by pressing ‘Clear’ Button.
13. Discuss an algorithm to display the sum and product of two numbers.

Answer:

To add two numbers:

1. Start.
2. Read the Value initialized for integer variables A and B.
3. Declare a third integer variable SUM for the addition.
4. SUM = A+B.
5. Display SUM.
6. Stop.

Similarly, to multiply two numbers:

1. Start.
2. Read the Value initialized for integer variables of A and B.
3. Declare a third integer variable P for the product.
4. Define P = A\*B.
5. Display P.
6. Stop.
7. Consider another example of an algorithm to increment a given number by 1.

Answer:

* 1. Start.
  2. Type an integer variable.
  3. Assign any numeral value to this variable.
  4. Add 1 to this variable.
  5. Display the value of the variable.
  6. Stop.

1. Write an algorithm to find out whether a number entered by a user is divisible by 5.

Answer:

1. Start.
2. Take a number as input and store it in the variable a of integer type.
3. To check divisibility with 5, check if (num % 5 == 0) then only, num is divisible by 5.
4. Print the variable value with message of assertion on whether the number is divisible by 5 or is not divisible by 5.
5. Stop.
6. Write an algorithm to display the first 10 multiples of 9.

Answer:

1. Start.
2. Declare four integer variables n, limit, count and Mul.
3. Initialize n with a value of 9.
4. Assign the limit value to 10.
5. Create an iteration cycle with count initialized with value 1.
6. The iteration upper limit being the 'limit' itself and the count incrementing by 1.
7. For each iteration take product of n and count.
8. Assign this product to the variable Mul for each loop.
9. Print within the loop the statement for taking values of [ n '\*' count '=' Mul].
10. Finish the loop
11. End.
12. Write an algorithm to find out whether a number entered by a user is even or odd.

Answer:

1. Start .
2. Accept a number in an integer variable.
3. Divide the number by 2.
4. Use the if conditional statements.
5. If the remainder is 0, the number is even.
6. If the remainder is not 0, the number is odd.
7. End.
8. Write an algorithm to accept two numbers from the user, subtract the first number from the second number, and display the result. 

Answer:

1. Start the algorithm.
2. Get the first number in an integer variable.
3. Get the second number in another integer variable.
4. Subtract the first number from the second number.
5. Print to display the result.
6. Write an algorithm to accept the principle, rate, and time values and display the simple interest.

Answer:

1. Start the algorithm.
2. Declare float variables for principle, rate, time and simple interest.
3. Accept input value from the user for principle, interest and time individually.
4. Assign the standard SI expression for Simple Interest utilizing the input variables as:

interest= (principle\*rate\*time)/100.

1. Print the simple interest variable to display the resultant simple interest.
2. End the algorithm.

11. Write an algorithm that accepts a number and displays its cube.

Answer:

1. Start the algorithm.
2. Declare an integer variable 'N' for the numeral value that needs to taken from the user.
3. Assign the input value this variable N.
4. Declare another integer variable 'C.'
5. Multiply N thrice to itself and assign this expression to variable C.
6. Display the variable C with message of result of Cube.
7. End.
8. Write an algorithm that accepts the length and breadth of a rectangle and displays its area and perimeter.

Answer:

1. Start the algorithm.
2. Declare four integer variables for length, breath, Area and Perimeter as l, b, A, P respectively.
3. Capture values for length and breadth of rectangle from user inputs.
4. Calculate the Area by assigning the expression 'l \* b' to variable A.
5. Calculate the Perimeter by assigning the expression '2\*(length + breath)' to variable P.
6. Display the respective values of A and P on the screen.
7. End the algorithm.
8. Consider another example where the following remarks should be given to the students depending on the grades they have scored in the exams:

Grade A: Excellent



 Grade B: Very Good

 Grade C: Good

 If any other grade is entered by the user instead of A, B, or C, the message, Invalid Input should be displayed.

Answer:

1. Start the algorithm.
2. Get the grade in a character variable say y.
3. Use if else, else if and else statements to give conditions for filtered assessment of the grade held in y.
4. If y == A, display “Grade A: Excellent;” else if y==B, display “Grade B: Very Good;” else if y==C, display “Grade C: Good;” else, display “Invalid Input”.
5. End the algorithm.
6. Write an algorithm to display the sum of the first 10 natural numbers.

Answer:

1. Start the algorithm.
2. Take an variable named' SUM' of integer type and first, initialize it with value 0
3. Create a repetitive loop cycle and define within it another integer variable 'i' with value 1.
4. Keep the limit of repetition of the loop equalling 10 and increment i by one for each loop.
5. Write within the loop's body the new assigned expression for variable SUM that assigns it a value of singly incremented i like: [SUM +=i].
6. After the end of this loop display the value of SUM that will represent addition of the first 10 natural numbers.
7. End the algorithm.
8. Write a pseudocode to display the sum of the first 10 natural numbers.

Answer:

1. begin
2. numeric nSum, i
3. SET nSum = 0
4. FOR i = 1 to 10
5. nSum +=i
6. ENDFOR
7. Display nSum
8. end
9. Write an algorithm to display the table of 12.

Answer:

1. Start.
2. Declare four integer variables n, limit, count and Mul.
3. Initialize n with a value of 12.
4. Assign the limit value to 10.
5. Create an iteration cycle with count initialized with value 1.
6. The iteration upper limit being the 'limit' itself and the count incrementing by 1.
7. For each iteration take product of n and count.
8. Assign this product to the variable Mul for each loop.
9. Print within the loop the statement for taking values of [ n '\*' count '=' Mul].
10. Finish the loop
11. End.
12. Write a pseudocode to display the table of 12.

Answer:

1. begin
2. numeric nN, nLimit, nCount, nMul
3. SET nN = 12, nLimit = 10
4. Display 'The Table of 12: 'n\
5. FOR nCount = 1 to nLimt and nCount++
6. nMul = nNum \* nCount
7. Display nNum '\*' nCount '=' nMul n\
8. ENDFOR
9. end
10. Write an algorithm to display the factorial of any given number.

Answer:

1. Step1: Start.
2. Step2: Initializations of integer variables i=1 and Fact=1.
3. Step3: Accept and Read the input number by the user for integer variable n.
4. Step4: Repeat from Step4 through Step6 to stop until i=n
5. Step5: Fact = Fact \* i
6. Step6: Increment i by 1 like i = i + 1
7. Step7: Print Fact
8. Step8: Stop.
9. Write a pseudocode to display the factorial of any given number.

Answer:

1. begin
2. numeric nN, nI, nFact
3. set nI = 1 and nFact = 1
4. accept nN
5. FOR nI = 1 to nN and nI++
6. nFact = nFact \* nI
7. ENDFOR
8. Display 'The Factorial of ' nN + ':' nFact
9. end
10. Write a pseudocode that accepts a number and checks whether it's positive, negative, or zero.

Answer:

*Algorithm -*

1. Start.
2. Input a number from user in some integer variable say num.
3. use Switch Case for the user input.
4. Case 1(Check if num < 0) then, display number is negative.
5. If True, Break else Continue.
6. Case 2(Check if num > 0) then, display number is positive.
7. If True, Break else Continue.
8. Case 3(Check if num == 0) then, display number is zero.
9. End.

*Pseudocode -*

1. begin
2. accept numeric nN
3. SELECT CASE(nN)
4. Condition 1(nN<0):

display 'The number is Negative.'

break

1. Condition 2(nN>0):

display 'The number is Positive.'

break

1. Condition 3(nN==0):

display 'The number is Zero.'

break

1. end
2. Write a pseudocode that accepts a number and checks if the number is prime.

Answer:

*Algorithm -*

1. Step1: Start.
2. Step2: Initializations of integer variable N via accepting user input.
3. Step3: Define another integer variable 'I' with value 2.
4. Step4: Use Do While loop with Nested Selection(If-Then-Else) Statement.
5. Step5: check IF the condition (I <= N/2)
6. Step6: Do
   1. check IF the condition(N % I == 0)
      1. Display 'Number Entered is Not Prime'
      2. ENDIF
   2. ELSE for inner condition
7. WHILE (I = I +1)
8. Return back to Step5
9. ENDELSE
   1. ENDIF
10. Step7: ELSE for outer condition
11. Display 'Number Entered is Prime'
12. ENDELSE
13. Step8: Stop.

*Pseudocode -*

1. begin
2. numeric nN, nI
3. set nI = 2
4. accept nN
5. Condition 1 (I <= N/2)

Perform following Do statements

check second Condition 1a (N % I == 0)

Display 'Number Entered is Not Prime'

END Condition 1a

1. ELSE for Condition 1a

WHILE (I = I +1)

Return back to Condition 1

ENDELSE

1. END Condition1
2. ELSE for Condition1

Display 'Number Entered is Prime'

ENDELSE

1. end
2. Write a pseudocode that prints the following Fibonacci series:

# 0 1 1 2 3 5 8

Answer:

*Algorithm -*

1. Start.
2. Initialize three integer variables x, y and z with values 0, 1 and 0 respectively.
3. Display x and y.
4. WHILE z to 8

Assign z = x + y

Assign x = y

Assign y = z

Display z

1. End.

*Pseudocode -*

1. begin
2. set numeric x =0, y = 1, z = 0
3. Display x
4. Display y
5. REPEAT

z = x + y

x = y

y = z

Display z

UNTIL z != 9

1. end
2. Consider the example of a game in which a user can control **the movement of a character by using the arrow keys.**

* When the user presses the **1,** the character should be **moved to the left**.
*  When the user presses the **right arrow** key, the character should be **moved to the right**.
* If the user tries to move the character **beyond the left or right edge of the screen**, the character should turn in the opposite direction.

The **game should end** when the user presses the **spacebar**.

Answer:

*Algorithm -*

1. Start.
2. Initialize two byte variables and assign them with key press inputs.
3. First, capture the binary input of the number key 1 of keyboard and assign an overriding functionality for cursor moving left motion on single button press and release and on persistent button press.
4. Similarly, accept the default input functionality of right arrow button.
5. Initialize an integer variable and assign it value for screen character length.
6. Associate the functional behaviour wherein on pressing key 1 the screen length character goes for lowest negative count of reaching the minimum 0.
7. Associate the functional behaviour wherein on pressing Right Arrow key the screen length character goes for positive upper count of 60.
8. For both right and left motion functionality assign the reverse character motion flow on crossing the lower and upper limit respectively.
9. End.

*Pseudocode -*

1. begin
2. set byte L = @LeftMove(button key press 1), R = Arrow button press
3. Accept Screen Character Length integer ACL = Length(Screen\_Character\_per\_Line) say, 60
4. Condition1 (@ACL(L)<0)

Move Characters Back on ACL from 0 to 60

END Condition1

1. Condition2 (@ACL(L)>60)

Move Characters Back on ACL from 60 to 0

END Condition2

1. end
2. Write a pseudocode that accepts two numbers and displays their sum and product. Considering the preceding requirements, identify the variables and constants required for writing the pseudocode.

Answer:

*Pseudocode -*

1. begin
2. numeric nNum1, nNum2, nSum, nProduct
3. accept Num1
4. accept Num2
5. nSum = nNum1 + nNum2
6. nProduct = nNum1 \* nNum2
7. display 'The Sum of ' + nNum1 + 'and ' + nNum2 + 'is ' + nSum + '!!'
8. display 'The Product of ' + nNum1 + 'and ' + nNum2 + 'is ' + nProduct + '!!'
9. end
10. Write a pseudocode that accepts the product name and quantity as input and displays the total price using the unit prices already stored in the system. Considering the preceding requirements, identify the variables and constants required for writing the pseudocode.
11. Write a pseudocode that accepts the temperature in Celsius, converts it into Fahrenheit, and then, displays the result.

 Hint :- Temperature in Fahrenheit = (9/5)\*Temperature in Celsius+32

1. Write a pseudocode where two values have to be accepted from a user. The values have to be then compared, and a result indicating whether they are equal or not has to be displayed
2. Write a pseudocode to evaluate the following mathematical expression:

nResult = n\*(n-1)/n

1. Write a pseudocode, where the discount percentage on a TV needs to be decided on the basis of the type of TV. If the TV is CRT (C), the discount will be 5% of the selling price (SP). If the TV is LCD (L), then the discount will depend on the size of the TV screen. For 14 inches screen, the discount is 8% of the SP. For 21 inches screen, the discount is 10% of the SP. The following table summarizes the discount rates.
2. Write a pseudocode, where all candidates have to take two tests before appearing for an interview. A candidate is selected for the interview round, based on the scores of the two tests. The individual scorc in each test should be greater than 75, and the average score for the two tests should be a minimum of 80. A call letter for the interview is to be sent to candidates who have been selected, and a rejection letter is to be sent to the rest.
3. Write a pseudocode, where you need to write a pseudocode to accept two numbers and any one of the operators: +, - and /. Based on the operator entered, the pseudocode should add, subtract, multiply, or divide the numbers and display the result.
4. Write a pseudocode, scenario of automated telephone call transfer to various departments of the company, such as Marketing, Finance, Customer Care, Human Resource (HR), and Information.
5. Write a pseudocode, where you need to write a pseudocode that accepts a letter from the alphabet and checks whether it is a vowel.