Question One

Write an algorithm (Pseudocode) to find the average of unknown number of grades. Your algorithm should stop when -1 grade is entered.

- Set sum as 0
- Set counter as 0
- Ask the user to enter the first grade
- Read and save grade as gr
- While gr doesn't equal -1
    - Set sum as old sum plus gr
    - Increment counter
    - Ask the user to enter another grade
    - Read and save grade as gr
- End while
- If counter equal zero
    - Print "No grades were entered"
- End if
- Else
    - Set average as sum divided by counter
    - Print "The average is" average
- End if

Question Two

Write an algorithm (Pseudocode) to count the number of evens and the number of odds for an unknown number of integers that the user enters. Your algorithm should stop when the zero number is entered.

- Set evens equal to 0
- Set odds equal to 0
- Ask the user to enter the first number
- Read number and save as num
- While num doesn't equal zero
    - Divide num by ten and save remainder in rem
    - If rem equal to zero
        - Increment evens
    - End if
    - Else
        - Increment odds
    - End if
    - Ask the user to enter another number
    - Read number and save as num
- End while
- Print "Evens: " evens
- Print "Odds: " odds

Question Three

Write an algorithm (Pseudocode) that asks the user to enter 3 different integers, and sort them in ascending order.

- Ask the user to enter the first number
- Read number and save it as num1
- Ask the user to enter the second number
- Read number and save it as num2
- Ask the user to enter the third number
- Read number and save it as num3
- If num1 is greater than num2
    - Swap num1 and num2
- End if
- If num1 is greater than num3
    - Swap num1 and num3
- End if
- If num2 is greater than num3
    - Swap num2 and num3
- End if
- Print num1, num2, num3

Question Four

Write an algorithm (Pseudocode) to find the following for a given 3 digits integer

A. Sum of digits
B. Reverse the digits
C. Sort the digits in ascending order

A)

- Ask the user to enter a 3-digit number
- Read number and save as number
- Divide number by 10 and save remainder as rem1
- Divide number by 10 and save result in number
- Divide number by 10 and save remainder as rem2
- Divide number by 10 and save result in number
- Set result as rem1 plus rem2 plus number
- Print result

B)

- Ask the user to enter a 3-digit number
- Read number and save as number
- Divide number by 10 and save remainder as rem1
- Divide number by 10 and save result in number
- Divide number by 10 and save remainder as rem2
- Divide number by 10 and save result in number
- Multiply rem1 by 100 and save result in rem1
- Multiply rem2 by 10 and save result in rem2
- Set result as rem1 plus rem2 plus number
- Print result

C)

- Ask the user to enter a 3-digit number
- Read number and save as number
- Divide number by 10 and save remainder as rem1
- Divide number by 10 and save result in number
- Divide number by 10 and save remainder as rem2
- Divide number by 10 and save result in number
- Multiply rem1 by 100 and save result in rem1
- Multiply rem2 by 10 and save result in rem2
- If rem1 is greater than rem2
    - Swap rem1 and rem2
- End if
- If rem1 is greater than number
    - Swap rem1 and number
- End if
- If rem2 is greater than number
    - Swap rem2 and number
- End if
- Print rem1, rem2, number

Question Five

Write an algorithm (Pseudocode) to simulate a simple calculator for two numbers with +, -, /, * operations.

- Ask the user to enter a number
- Read number and save as x
- Ask the user to enter another number
- Read number and save as y
- Ask the user to enter an operation
- Read operation and save as operation
- Set result equal to zero
- If operation is equal to +
    - Set result as x plus y
- End if
- If operation is equal to –
    - Set result as x subtracted by y
- End if
- Else If operation is equal to *
    - Set result as x multiplied by y
- End if
- Else if operation is equal to /
    - If y is equal to zero
        - Print “Can’t divide by zero”
        - Exit program
    - End if
    - Else
        - Set result equal to x divided by y
    - End if
- End if
- Print result

Question Six

Write an algorithm (Pseudocode) to find the largest element between 5 given integers

- Ask the user to enter a number
- Read number and save as num1
- Ask the user to enter a number
- Read number and save as num2
- Ask the user to enter a number
- Read number and save as num3
- Ask the user to enter a number
- Read number and save as num4
- Ask the user to enter a number
- Read number and save as num5
- Set max equal to num1
- If num2 is greater than max
    - Set max equal to num2
- End if
- If num3 is greater than max
    - Set max equal to num3
- End if
- If num4 is greater than max
    - Set max equal to num4
- End if
- If num5 is greater than max
    - Set max equal to num5
- End if
- Print max

Question Seven

Write an algorithm (Pseudocode) to find the following for any given integer (with an unknown number of digits)

A. Sum of digits
B. Maximum digit
C. Minimum digit

- Ask the user to enter a number
- Read number and save as n
- Set min equal to 9
- Set max equal to 0
- Set sum equal to 0
- While n not equal zero
    - Divide n by ten and save remainder in rem
    - Divide n by ten and save result in n
    - Set sum equal to sum plus rem
    - If remainder is less than min
        - Set min equal rem
    - End if
    - If remainder is greater than max
        - Set max equal rem
    - End if
- End while
- Print sum, max, min

Question Eight

Write an algorithm (Pseudocode) to decide whether a number is a perfect number or not. A perfect number is a number which is equal to the sum of all its divisors (الأعداد التي يقبل القسمة عليها) excluding the number itself. (e.g. 6 is a perfect number because 1 + 2 + 3 = 6)

- Ask the user to enter a number
- Read number and save as x
- If x is less than 2
    - Print x “is not a perfect number”
    - Exit program
- End if
- Set counter equal to 1
- Set sum equal to zero
- While counter is less than x
    - Divide x by counter and save remainder as rem
    - If rem is equal to zero
        - Set sum as old sum plus counter
    - End if
    - Increment counter
- End while
- If x is equal to sum
    - Print x “ is a perfect number”
- End if
- Else
    - Print x “ is not a perfect number”
- End if