EXERCISES II

Structured programming

Solve the next problems with flowchart and pseudocode

1. Design a program to show all numbers between 1 and 100. If the number is a multiple of 7 you should show the message "The number xx is a multiple of 7". If the number is a multiple of 13 you should show the message "The number xx is a multiple of 13". If the number is a multiple of 7 and 13 you should show both messages.

2. Design a program for reading an integer between 0 and 10 and show the times table. To ask for the number you should show the next message "Enter one number between 0 and 10". If the number is out of the boundaries, the program should show the message "The number is out of the boundaries". If the number is valid, it should show the times table following the next format:

   7*0=0
   7*1=7
   …..
   7*10=70

3. Design a program that asks how many numbers the user wants to introduce. Then, the user would have to introduce the numbers one by one and the program should say if each one of the numbers is odd or even. If the user inputs 0 or a negative number, it is not valid, and the system should ask for another number. The messages are the following:
   "How many numbers do you want input?" to ask for the number of numbers.
   "Enter one number greater than 0:" to ask for a number.
   "The number is not valid, it should be greater than 0" to inform that the number is not valid.
   "The number XX is odd"
   "The number XX is even"

4. Design a program that reads a positive number N greater than 0 and show the result of the sum of the N numbers between 1 and N. If the number N is not valid, the program should ask for it again. The messages are the following:
   "Enter one number greater than 0:"
   "The number is not right, try again."
   "The sum of the N first numbers is XX."

5. Design a program that asks for numbers until the user inputs a negative one. When this happens, the program will show how many positive numbers have been introduced by the user. The messages are the following:
   "Enter a number (negative to finish):"
   "You have entered XX positive numbers."

6. Design a program that reads two integer numbers, for example numberA and numberB, and calculates the product of both numbers without use multiplication, but using the sum. The messages are the following:
   “Enter one number:”
   “The product is XX”

7. Design a program that asks how many numbers the user wants to write. After the user enters all numbers, the program should say the medium of the numbers. If the user inputs a wrong number, the program should ask for it again. The messages are the following:
   “How many numbers do you want input?” to ask for the number of numbers.
   “Enter one number greater than 0:” to ask for a number.
   “The number is not valid, it should be greater than 0” to inform that the number is not valid.
   “The medium is XX.XX” to show the result.

8. Design a program that asks for a set of numbers. After inputting each number, the program should ask “Do you want to enter more numbers (Y/N)?”. If the answer is “Y” the program asks for other numbers. When the user finishes to enter all the numbers, the program should say which one is the smallest. The messages are the following:
   “Enter one number:”
   “Do you want to enter more number (Y/N)?”
   “The smallest number is XX”

9. Design a program that reads an integer positive number greater than 0 and says if it’s a “perfect number”. A number is perfect if it is equal to the sum of all its divisors. The messages are the following:
   “Enter an integer positive number greater than 0:”
   “The number is not valid, try again.”
   “The number is perfect.”
   “The number is not perfect.”

10. Design a program that reads an integer positive number and says the “factorial” of the number. To calculate the factorial you should know that
    FACT(0)= 1
    FACT(1) =1
    FACT(N)= N*(N-1)*(N-2)*....1
    The messages are the following:
    “Enter an integer positive number:”
    “The number is not valid, try again.”
    “The factorial is XX”