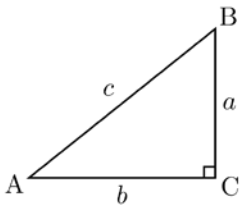
Exercise #6

- **Problem 1**.
  - (Sides of a Right Triangle) Write a function that reads three nonzero integers and determines whether they are the sides of a right-angled triangle. 輸入三邊長，判斷是否為直角三角形
  - The function should take three integer arguments and return 1 (true) if the arguments comprise a right-angled triangle, and 0 (false) otherwise. Use this function in a program that inputs a series of sets of integers.
  - In any right triangle, $a^2 + b^2 = c^2$



- **Problem 2.**
  - ***(Perfect Numbers)*** An integer number is said to be a *perfect number* if its factors, including 1 (but not the number itself), sum to the number. For example, 6 is a perfect number because 6 = 1 + 2 + 3.
  - Write a function **isPerfect** that determines whether parameter number is a perfect number.
  - Use this function in a program that determines and prints all the perfect numbers between 1 and 1000. Print the factors of each perfect number to confirm that the number is indeed perfect.

- **Problem 3.**
  - ***(Sum of Digits)*** Write a function that takes an integer and returns the sum of its digits. For example, given the number 7631, the function should return 17.
    - 7+6+3+1 = 17

- **Problem 4.** π estimation
  - Use the Leibniz formula for π to estimate the value of Pi. In main(), input a positive integer number n and pass n to the function LeibnizPi(n). The function LeibnizPi(n) calculates the value of Pi and returns to main(). Finally, main() outputs the Pi value. Repeat the program using a loop in main().Input 0 to end the program. **You should validate whether or not the input is a nonnegative integer.**
  - Leibniz formula for π:

$$\pi = 4 \sum_{k=1}^{n} \frac{(-1)^{k-1}}{2k-1}$$