

Programación Científica 2

Taller 5: Condiciones y Flujo de control

Profesor: Gilles Pieffet

1. El Último teorema de Fermat dice que no existen números enteros positivos a , b y c tal que:

$$a^n + b^n = c^n$$

para cualquier valor de n mayor a 2.

- a) Escriba una función `chequear_fermat` que recibe cuatro parámetros a , b , c y n y que chequee si el teorema de Fermat es válido. Si n es mayor a 2 y se encuentra que:

$$a^n + b^n = c^n$$

el programar debe imprimir en la pantalla “Oye, Fermat estaba equivocado!”. Si no, el programa debe imprimir “No, la igualdad no funciona”.

- b) Escribe una función que pide al usuario los valores a , b , c y n , los convierte en `int` y llama a `chequear_fermat` para ver si están en acuerdo con el teorema de Fermat.
 - c) Modifique la función `chequear_fermat` de tal manera que verifique que n es mayor a 2 (antes de chequear si el teorema de Fermat es válido).
2. Si tiene tres palitos, no es siempre posible organizarlos en un triángulo. Por ejemplo, si uno de los palitos tiene una longitud de 12 cm y los otros tienen una longitud de 3 cm cada uno, es muy claro que los dos pequeños palos no se van a poder tocar. Con tres longitudes cualquiera, hay un test muy sencillo para ver si pueden formar un triángulo:

Si una de las longitudes es mayor a la suma de las dos otras, no se puede formar un triángulo. De lo contrario, es posible formar uno (si la suma de dos longitudes es igual a la tercera, se forma lo que se llama un triángulo degenerado).

- a) Escribe una función llamada `es_triangulo` que recibe tres números enteros como argumentos, y que imprime “Si” o “No” dependiendo de si se puede formar un triángulo con tres palos con estas longitudes.
- b) Modifique el programa para tener en cuenta el caso en el cual el triángulo es degenerado. En este caso imprima “Si, pero es degenerado” en vez de simplemente “Si”. Escribe la expresión condicional de tal manera que chequee si dos lados juntos son iguales al tercero.
- c) Haga lo mismo que en (b) pero de tal manera que la expresión condicional chequee si dos lados juntos **NO** son iguales al tercero.

- d) Escribe una función que le pide al usuario las tres longitudes y utiliza `es_triangulo` para chequear si se puede formar un triángulo con tres palitos con estas longitudes.

3. **Adivina el número.** Tiene que escribir un juego en el cual el computador *piensa* un número de 1 a 20, y tiene 4 intentos para adivinar el número correcto. Abajo es lo que aparece cuando el programa está corriendo. Lo que entra el usuario está indicado con letras negrillas.

Hola! Cual es tu nombre?

Paloma

Bueno Paloma, estoy pensando en un número de 1 a 20.

Adivina cuál es.

10

No. Eso era el intento 1.

Adivina cuál es.

15

No. Eso era el intento 2.

Adivina cuál es.

7

Buen trabajo, Paloma. Encontraste mi número en 3 intentos.

Si no se encuentra el número en cuatro intentos, lo que aparece después del cuarto intento es:

No. Eso era el intento 4.

El número que tenía en la mente era el 14

Nota 1: Para que el computador escoja un número al azar tenemos que utilizar una función que genera un número aleatorio: la función `randint` del modulo `random`. Para utilizarla, se le pasa como argumentos el primero y el último número (los límites) del rango que nos interesa, y la función devuelve un número adentro de este rango.

`randint(1, 20) -> número entre 1 y 20`

Nota 2: Para salir de un bucle de manera anticipada (por ejemplo si se cumple una condición externa), se utiliza la instrucción `break`.

Nota 3: Cuando está probando el programa, con el fin de evitar la demora asociada con la entrada manual de datos, puede utilizar valores predefinidos o que se inicializan automáticamente.

4. **Adivina el número v2.** Para ayudar al jugador a encontrar el número correcto, indica para cada intento si su número es más grande o más pequeño que el número a encontrar:
Adivina cual es.
18
Su número es mas grande.