

## **Guía 8 – Resolución de problemas 2 – SUSSINI PATRICIO**

### **8.1) Sumar los números del 1 al 10.**

- Usaría un método while.
- La condición de inicio es si mi acumulador es igual a 0, y debo inicializarlo en 0
- La condición de parada es cuando mi contador de iteraciones ya llego a 10.
- Necesito ambos, acumulador para hacer la suma y contador para las iteraciones, que deben ser 10 y en cada una sumar un numero diferente.

### **8.2) Mostrar los primeros 10 números de la serie de Fibonacci.**

- Usaría un FOR porque el número de iteraciones (8) es conocido después de imprimir los dos primeros números.
- La condición inicial es declarar  $a = 0$  y  $b = 1$  y mostrar estos números.
- La condición de parada es cuando el bucle se ejecutó 8 veces.
- Se usaría un contador en el FOR para iteraciones y dos acumuladores a y b para almacenar los últimos valores.

### **8.3) Encontrar cuántos días faltan para alcanzar una meta de ahorro semanal de \$500, partiendo de \$0 y ahorrando \$50 por día.**

- Primero usaría un WHILE porque no se de antemano cuantos días necesito para alcanzar la meta.
- Como condición inicial declaro ahorro en 0 y días en 0.
- Para la parada el bucle se detendrá cuando ahorro sea menor o igual a 500 y se alcance la meta.
- Necesito un contador de días para conocer la respuesta, un acumulador de ahorro que sume 50 por día hasta alcanzar la meta.

### **8.4) Contar cuántas veces aparece un carácter en una cadena de texto.**

- Usaría un FOR de ciclo para iterar directamente sobre cada carácter de la cadena.
- Como condición inicial declaro el contador en 0 para almacenar el numero de veces que aparece el dicho carácter.
- Condición de parada se da cuando se han recorrido todos los caracteres de la cadena.
- Se utilizaría un contador que se incrementa en uno cada vez que el carácter actual coincide con el carácter buscado. No se necesita acumulador.

**8.5)** Calcular cuántas filas completas pueden formarse con 150 sillas distribuidas en grupos de 10.

- Usaría un WHILE porque necesito verificar repetidamente si aún hay suficientes sillas para formar una fila completa.
- Mi condición inicial es declarar sillas igual a 150 y filas = 0 como contador de filas completas.
- El bucle de detiene cuando sillas < 10 es decir cuando ya no hay suficientes sillas para formar otra fila completa.
- Necesito un contador de filas que lleva el numero de filas formadas correctamente. Un decrementador de sillas que reduce el número de sillas disponibles en 10 cada vez que se forma una fila correctamente. No se necesita acumulador.

**8.6)** Imprimir todos los números impares entre 1 y 20.

- Uso un ciclo FOR porque el rango de números es conocido y puedo iterar directamente sobre él.
- Mi condición inicial es declarar un bucle que itere desde el 1 hasta el 20.
- La condición de parada es cuando se alcanza la iteración 20.
- Se usa un contador en el FOR que controla las iteraciones.

**8.7)** Determinar si un número ingresado por el usuario es perfecto.

- Usaría un ciclo FOR, porque necesito iterar sobre todos los posibles divisores del número, lo cual es un rango conocido de 1 a n-1.
- Mi condición inicial es que declaro la suma de divisores a 0 para acumular los divisores propios.
- La condición de parada es cuando se han verificado todos los números desde 1 hasta n-1.
- Se usa un acumulador que suma los divisores propios de n. no se usa contador ni decrementador pero si al final se compara si la suma de los divisores == n.- para saber si es un numero perfecto.

**8.8)** Calcular cuántos pasos son necesarios para reducir un número a 0 restando 5 en cada iteración.

- Usaría un ciclo WHILE porque no sabes de antemano cuántas veces necesito restar 5 hasta llegar a 0.
- La condición inicial es declarar un numero como el valor ingresado por el usuario y un contador en 0 para contar las iteraciones necesarias.
- La condición de parada es cuando el numero es menor o igual a 0, cuando el numero ya se ha reducido a 0 o menos.
- Se necesita un contador de pasos que lleva la cuenta de las veces que se ha restado 5. Un decrementador numero que se reduce en 5 en cada iteración. No hace falta un acumulador.

**8.9) Mostrar las tablas de multiplicar del 1 al 5.**

- Necesito usar dos ciclos FOR, para iterar sobre dos rangos conocidos, las tablas del 1 y del 5 y los multiplicadores de 1 al 10.
- Como condición inicial el primer bucle itera sobre las tablas del 1 al 5 y el segundo sobre los multiplicadores del 1 al 10.
- La condición de parada es cuando ambos bucles terminan y alcanzan sus límites superiores.
- No se necesita un acumulador y los mismos bucles hacen las veces de contadores de iteraciones si se quiere.

**8.10) Verificar si una palabra es un palíndromo comparando sus letras desde los extremos hacia el centro.**

- Usaría un WHILE porque necesitas comparar caracteres desde los extremos hacia el centro, y el número de comparaciones varía según la palabra ingresada
- La condición inicial declara dos índices inicio = 0 y fin que es el largo de la cadena -1 del último carácter.
- La condición de parada dice que el bucle se detiene cuando inicio es mayor o igual a fin, es decir cuando se han comparado todos los pares de caracteres o se ha llegado al centro de la palabra.
- No se necesita un acumulador o decrementador, los índices de inicio y fin funcionan como contadores para moverse hacia el centro de la palabra. Si los caracteres en inicio y fin no coinciden en cualquier momento, la palabra no es un palíndromo. Si termina sin encontrar diferencias si es.