

## Ejercicio resuelto de la Práctica 2: Ciclo Mientras, contador, acumulador

**16-**DESARROLLE UN DIAGRAMA DE FLUJO QUE PERMITA CALCULAR EL PROMEDIO DE LAS NOTAS DE UN CURSO. EL ALGORITMO TERMINA CUANDO SE INGresa LA NOTA 0.

**Pasos para la resolución:** Como buena práctica, sugerimos hacer el diseño de la estrategia:

**1) Análisis del problema (QUE??):** Leer detenidamente el enunciado, e identificar:

**Datos:** Cuáles son los datos del problema: Conozco la nota de cada alumno del curso.

**Proceso:** Qué operaciones debo hacer: Tengo que calcular el promedio de notas del curso. Para ello deberé hacer cálculos auxiliares: contar cuántos alumnos son, y por cada nota de cada alumno ir acumulando ese valor, hasta tener la suma de todas las notas acumuladas.

¿Cuántos datos de notas son? No lo sé. Pero puedo identificar que terminarán con nota=0.

Con estos dos valores podré calcular el promedio.

**Resultados:** Qué resultados me piden: Mostrar el promedio de notas del curso.

**2) Diseño de la estrategia (COMO??):** Defino cómo voy a diseñar el algoritmo, y le doy un orden lógico a las tareas que voy a realizar:

**a) Entradas:** Ingreso la nota del alumno en la variable **nota**.

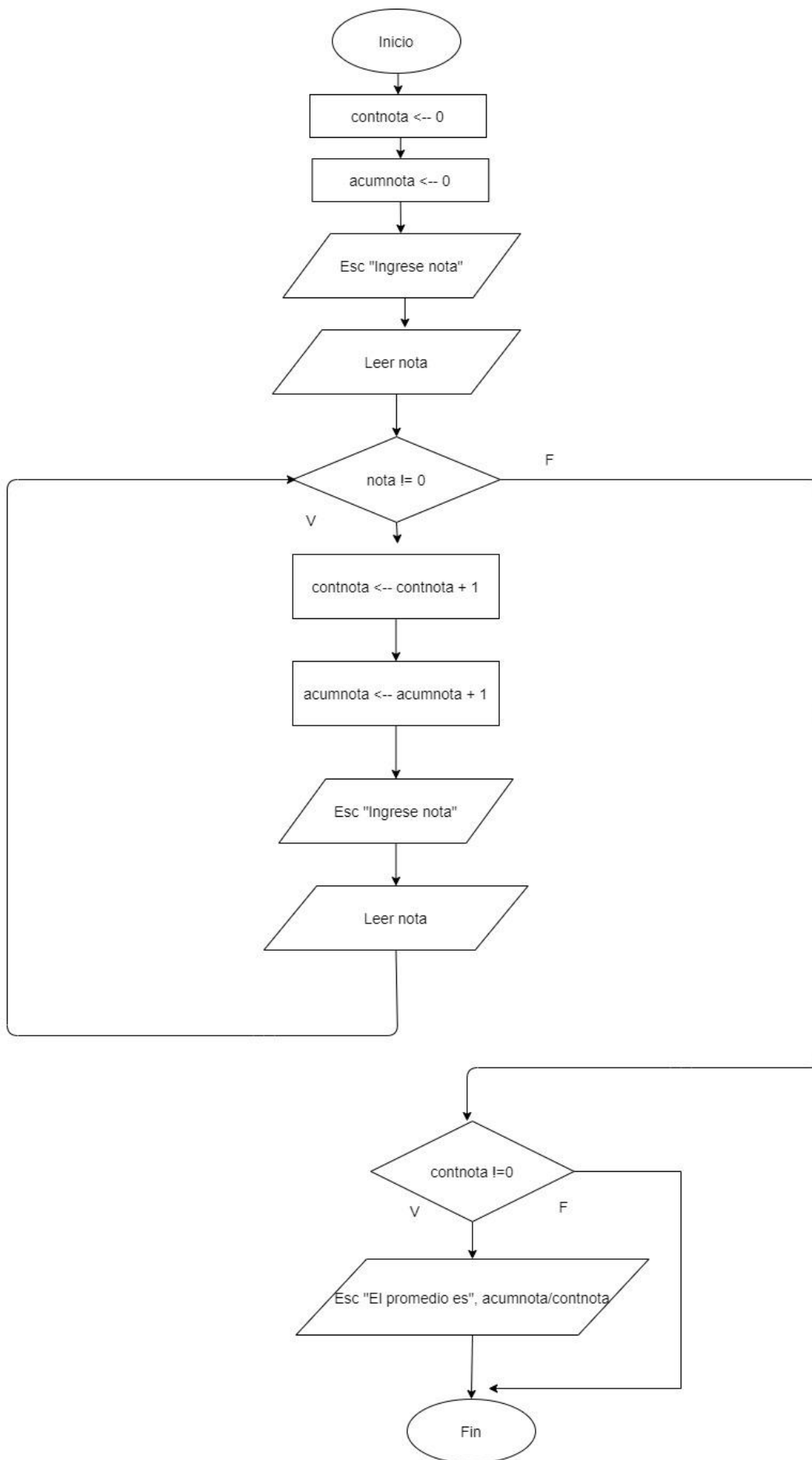
**b) Proceso:** Primero identifico **la condición de fin**: Mientras la nota sea distinta de cero voy a procesar una nota. Procesar significa: Por cada nota: voy a sumar 1 a un contador de notas y voy a sumar el valor de la nota en un acumulador de notas. Estas tareas las repito por cada nota.

**c) Salida:** Informo el promedio del curso: Calculo la división entre el valor acumulado de notas y el contador de notas.

Identificamos también, qué variables vamos a utilizar, y las enumeramos con su tipo: En este ejercicio las variables serán:

**nota, contnota, acumnota tipo entero**

**3) Diseño el algoritmo en base a la estrategia en Diagrama de Flujo:**



#### Observaciones de cómo se diseñó el Diagrama de Flujo:

a) La variable `contnota` y `acumnota` se inicializaron en cero. ¿Para qué? Las variables en la memoria no tienen valor inicial. Yo le asigno un valor inicial. La primera vez que ejecuto la sentencia `contnota ← contnota + 1`, a la variable `contnota` le asigno lo que tiene en ese momento la variable `contnota` (que la primera vez es 0) más 1. La vez siguiente ya tiene un valor definido, que es 1, y al sumarle 1 quedará el valor 2 y así sucesivamente. Lo mismo ocurre con `acumnota`. **Siempre que necesite contar o acumular valores, la variable que utilizo de contador o acumulador debe ser inicializada en 0 antes del proceso donde la utilizo.**

b) Armado del ciclo `mientras`: **Siempre debe haber una lectura adelantada al ciclo mientras de la variable o variables que se usan en la condición del mismo.** En este caso la variable `nota` tiene que tener valor cuando evalúo la condición del `mientras`. **Luego, dentro del ciclo mientras debe haber una lectura de la variable `nota`, para que cambie el valor de verdad de la condición del ciclo (la condición sea falsa) y pueda terminar.**

#### 4) Prueba de escritorio: Muy importante ejecutar el algoritmo con un juego de datos, para ver si realmente resuelve el problema:

Armamos una tabla con las variables del algoritmo, simulando lo que ocurre en la memoria de la computadora. Las variables del algoritmo van tomando valor si se ejecuta una instrucción de lectura o asignación. Se van ejecutando las instrucciones en el orden lógico desde el inicio al fin. Así queda la memoria al ejecutarse el algoritmo:

nota	contnota	acumnota
	0	0
10	1	10
6	2	16
7	3	23
9	4	32
0		

Muestra el cartel: **El promedio de notas es 8**

Probamos que nuestro algoritmo resuelve el problema.