### Requerimientos del Programa 2

Utilizando el proceso **PSP1**, escribe un programa que:

1. Lea del teclado el nombre de uno o más archivos que contienen código fuente que compila exitosamente (el nombre del archivo puede tener cualquier extensión)
2. Lea del disco estos archivos y en cada uno cuente las Líneas de Código (LDC).
3. Utilizando las reglas de conteo descritas en el estándar de conteo, realiza los siguientes cálculos en cada parte:
   * Cuenta las LDC que tiene la parte (llamémosle T)
   * Utilizando las etiquetas de conteo calcula:
     + La cantidad de ítems que tiene la parte (llamémosle I)
     + Las LDC base (llamémosle B)
     + Las LDC borradas (llamémosle D)
     + Las LDC modificadas (llamémosle M)
   * Calcula las LDC agregadas (llamémosle A) con la siguiente fórmula: A = T - B + D
   * Basado en estos datos, clasifica la parte de acuerdo al siguiente criterio:

|  |  |
| --- | --- |
| **Tipo de parte** | **Criterio** |
| BASE | B>0 y (M>0 o D>0 o A>0) |
| NUEVA | B=0 y M=0 y D=0 y A>0 |
| REUSADA | B>0 y M=0 y D=0 y A=0 |

En cualquier otro caso existe un **error**

1. Durante todo el proceso de leer y contar las partes, lleva un contador global (diferente a los anteriores) que cuenta el total de LDC de ***TODOS*** los archivos (pueden haber LDC afuera de las partes).
2. Escriba en pantalla y en un archivo llamado “**ConteoLDC.txt**” la anterior información ***siguiendo*** el siguiente formato (“NNNN” es el nombre de la parte y “xx” es un entero ≥ 0):

PARTES BASE:

NNNN: T=xx, I=xx, B=xx, D=xx, M=xx, A=xx

NNNN: T=xx, I=xx, B=xx, D=xx, M=xx, A=xx



NNNN: T=xx, I=xx, B=xx, D=xx, M=xx, A=xx

--------------------------------------------

PARTES NUEVAS:

NNNN: T=xx, I=xx

NNNN: T=xx, I=xx



NNNN: T=xx, I=xx

--------------------------------------------

PARTES REUSADAS:

NNNN: T=xx, I=xx, B=xx

NNNN: T=xx, I=xx, B=xx



NNNN: T=xx, I=xx, B=xx

--------------------------------------------

Total de LDC=xx

Otras características que ***debe*** cumplir el programa:

1. No utilizará ningún GUI para operar (funcionará desde la consola)
2. Debe estar construido con programación orientada a objetos
3. Debe contar con al menos 3 clases “relevantes”, de las cuales al menos una es BASE (la clase que contiene el “main” se cuenta como una de estas 3 clases)
4. El ***único*** código que puede ser reutilizado es el de tu programa 1
5. Debe manejar apropiadamente ***todas*** las condiciones normales y anormales
6. Debe pasar exitosamente ***todos*** los casos de prueba:
   * Los diseñados por ti en la fase de diseño.
   * Además, los siguientes 2 casos de prueba (obligatorio):

|  |  |  |
| --- | --- | --- |
| **Objetivo de la prueba** | **Instrucciones y**  **datos de entrada** | **Resultados Esperados** |
| Probar un archivo con datos | Teclear en pantalla:  test1.src | PARTES BASE:  Cuenta: T=84, I=2, B=38, D=12, M=2, A=58  Cliente: T=43, I=4, B=21, D=10, M=5, A=32  --------------------------------------------  PARTES NUEVAS:  Banco: T=31, I=3  --------------------------------------------  PARTES REUSADAS:  --------------------------------------------  Total de LDC=162 |
| Probar dos archivos con datos | Teclear en pantalla:  Test2a.src  Test2b.src | PARTES BASE:  Lista: T=39, I=4, B=65, D=35, M=3, A=9  --------------------------------------------  PARTES NUEVAS:  --------------------------------------------  PARTES REUSADAS:  Nodo: T=25, I=1, B=25  --------------------------------------------  Total de LDC=69 |