

# **Programa 4**

## **Proceso Personal para el Desarrollo de Software**

Este material fue realizado en base a material del curso “Personal Software Process for Engineers: PartI”, dictado por The Software Engineering Institute (SEI).

# Proceso Personal para el Desarrollo de Software

## Instructivo para el Programa 4

### Descripción

---

#### Descripción

El presente instructivo cubre los siguientes temas

Sección	Página
Requerimientos del Programa 4	3
Tablas de tamaños relativos	4
Calcular una tabla de tamaños relativos utilizando desviación estándar	4
Ejemplo	6
Instrucciones	9
Criterios de Evaluación	10

---

## Programa 4 Requerimientos

### Requerimientos Programa 4

Usando PSP0.1, escribir un programa para calcular tamaño relativo para los rangos, muy pequeño, pequeño, medio, grande, y muy grande usando desviación estándar.

Probar a fondo el programa. Probar el programa usando los datos proporcionados en las tablas 1 y 2. Los valores previstos se incluyen en la tabla 3.

Nombre de Clase	LOC	Número de Métodos
each_char	18	3
string_read	18	3
single_character	25	3
each_line	31	3
single_char	37	3
string_builder	82	5
string_manager	82	4
list_clump	87	4
list_clip	89	4
string_decrementer	230	10
Char	85	3
Character	87	3
Converter	558	10

Tabla 1. LOC/Método

Capítulo	Páginas
Prefacio	7
Capítulo 1	12
Capítulo 2	10
Capítulo 3	12
Capítulo 4	10
Capítulo 5	12
Capítulo 6	12
Capítulo 7	12
Capítulo 8	12
Capítulo 9	8
Apéndice A	8
Apéndice B	8
Apéndice C	20
Apéndice D	14
Apéndice E	18
Apéndice F	12

Tabla 2. Pgs/Capítulo

	VS	S	M	L	VL
LOC/Método	4.3953	8.5081	16.4696	31.8811	61.7137
Pgs/Capítulo	6.3375	8.4393	11.2381	14.9650	19.9280

Tabla 3. Valores esperados

# Tabla de Tamaños relativos

## Usando tablas de tamaños

Las tablas relativas de tamaño se utilizan para dar un marco para juzgar el tamaño de nuevas piezas de software en productos previstos. Por ejemplo, si se conoce los tamaños de todas las piezas de cierto tipo, previamente desarrolladas, se puede entonces juzgar mejor el tamaño probable de una nueva pieza de ese tipo. El procedimiento de la desviación de estándar descrito en la sección siguiente permite balancear las estimaciones para que se ajusten aproximadamente a la distribución normal.

El rango medio (M) es el área a partir de -0.5 desviaciones estándar a +0.5 desviaciones estándar de la medio, según las indicaciones de la figura 1. Si se asume que los datos aproximan una distribución normal, el número probable de las piezas que están dentro de más o menos 0.5 desviaciones estándar del valor medio es 38.3%. Siguiendo una lógica similar, el área en porcentajes de los restantes rangos es:

- 6.68 % deberían ser muy pequeños
- 24.17% deberían ser pequeños
- 38.2% deberían ser medianos
- 24.17% deberían ser grandes
- 6.68% deberían ser muy grandes

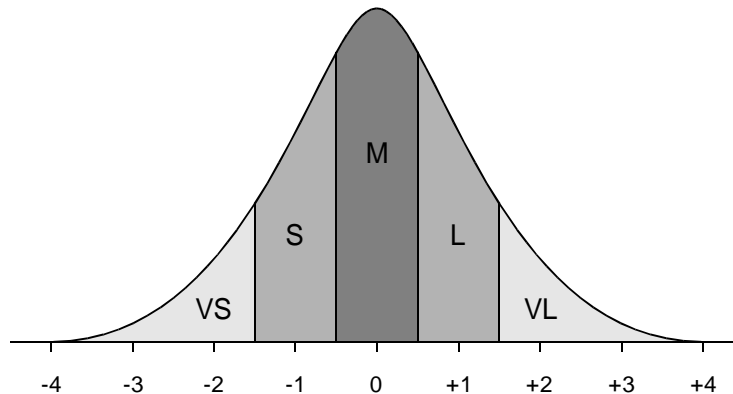


Figura 1. Ranges of standard deviations

## Calcular una tabla de tamaños relativos utilizando desviación estándar

Ciertos métodos de estimación dividen datos históricos de tamaño en categorías de clase. Una forma de hacer esto se basa en la desviación estándar. Primero, dividir los datos históricos en categorías funcionales donde cada una tiene por lo menos 6 a 8 miembros (cálculo, texto, y datos, por ejemplo). Para cada categoría se puede entonces calcular los rangos relativos de tamaño para VS, S, M, L, y VL siguiendo el procedimiento descrito a continuación

1. Dividir los tamaños de las piezas por el número de ítems en cada parte para determinar el tamaño por ítem, si fuera aplicable. Por ejemplo, puede no tenerse suficientes datos sobre las clases para desarrollar una tabla relativa de tamaño, pero tener suficientes datos sobre los métodos. Entonces, en lugar de utilizar el LOC total por clase, se puede utilizar LOC/método.

- 
2. Luego, se necesita transformar los datos utilizando el logaritmo natural. Esto es necesario porque no se pueden tener valores negativos de tamaño y los valores pequeños tienden a eso. Transformando los datos a su logaritmo natural permite graficar los datos alrededor de una media de cero. Para cada valor,  $x_i$ , tomar el logaritmo natural,  $\ln$ , obteniendo  $\ln(x_i)$ .

3. Calcular la media de estos  $n$  valores logarítmicos: 
$$avg = \frac{\sum_{i=1}^n \ln(x_i)}{n}.$$

4. Calcular la varianza de estos valores: 
$$var = \sigma^2 = \frac{\sum_{i=1}^n (\ln(x_i) - avg)^2}{(n-1)}.$$

5. Calcular la desviación estándar:  $\sigma = \sqrt{var}.$

6. Calcular los rangos logarítmicos :

$$\ln(VS) = avg - 2\sigma$$

$$\ln(S) = avg - \sigma$$

$$\ln(M) = avg$$

$$\ln(L) = avg + \sigma$$

$$\ln(VL) = avg + 2\sigma$$

7. Por último, convertir los valores en logaritmos naturales, de vuelta a su forma original, calculando el antilogaritmo. (calcular  $e$  elevado al valor del logaritmo) para obtener los valores medios de los rangos de tamaño

$$VS = e^{\ln(VS)}$$

$$S = e^{\ln(S)}$$

$$M = e^{\ln(M)}$$

$$L = e^{\ln(L)}$$

$$VL = e^{\ln(VL)}$$

---

## Ejemplo

### Ejemplo de cálculo de tablas relativas de tamaños

En este ejemplo, calcularemos los rangos relativos de tamaño de clases muy pequeñas, pequeñas, medias, grandes, y muy grandes usando la desviación estándar para los datos en la tabla siguiente.

Nombre de Clase	LOC	Número de Métodos	LOC/método
each_char	18	3	6.0000
string_read	18	3	6.0000
single_character	25	3	8.3333
each_line	31	3	10.3333
single_char	37	3	12.3333
string_builder	82	5	16.4000
string_manager	82	4	20.5000
list_clump	87	4	21.7500
list_clip	89	4	22.2500
string_decrementer	230	10	23.0000
Char	85	3	28.3333
Character	87	3	29.0000
Converter	558	10	55.8000

1. Dividir el tamaño de las piezas por el número de métodos en cada una, para determinar el tamaño por método, si es aplicable. En este caso, se calcula LOC/método para cada clase.
2. Para cada valor de tamaño,  $x_i$ , calcular el logaritmo natural,  $\ln$ , obteniendo  $\ln(x_i)$ .

Nombre de Clase	LOC/método	$\ln(x_i)$
each_char	6.0000	1.7918
string_read	6.0000	1.7918
single_character	8.3333	2.1203
each_line	10.3333	2.3354
single_char	12.3333	2.5123
string_builder	16.4000	2.7973
string_manager	20.5000	3.0204
list_clump	21.7500	3.0796
list_clip	22.2500	3.1023
string_decrementer	23.0000	3.1355
Char	28.3333	3.3440
Character	29.0000	3.3673
Converter	55.8000	4.0218
Total		36.4197

- 
3. Calcular la media de estos  $n$  valores logarítmicos:

$$avg = \frac{\sum_{i=1}^n \ln(x_i)}{n} = \frac{36.4197}{13} = 2.8015$$

4. Calcular la varianza de estos valores:

$$var = \sigma^2 = \frac{\sum_{i=1}^n (\ln(x_i) - avg)^2}{(n-1)} = \frac{5.2350}{12} = 0.4363$$

Nombre de Clase	LOC/método	$\ln(x_i)$	$(\ln(x_i) - avg)^2$
each_char	6.0000	1.7918	1.0196
string_read	6.0000	1.7918	1.0196
single_character	8.3333	2.1203	0.4641
each_line	10.3333	2.3354	0.2173
single_char	12.3333	2.5123	0.0836
string_builder	16.4000	2.7973	0.0000
string_manager	20.5000	3.0204	0.0479
list_clump	21.7500	3.0796	0.0773
list_clip	22.2500	3.1023	0.0905
string_decrementer	23.0000	3.1355	0.1115
Char	28.3333	3.3440	0.2943
Character	29.0000	3.3673	0.3201
Converter	55.8000	4.0218	1.4890
Total		36.4197	5.2350

5. Calcular la desviación estándar:  $\sigma = \sqrt{var} = \sqrt{0.4363} = 0.6605$

6. Calcular los rangos logarítmicos:

$$\ln(VS) = avg - 2\sigma = 2.8015 - 1.3210 = 1.4805$$

$$\ln(S) = avg - \sigma = 2.8015 - 0.6605 = 2.1410$$

$$\ln(M) = avg = 2.8015$$

$$\ln(L) = avg + \sigma = 2.8015 + 0.6605 = 3.4620$$

$$\ln(VL) = avg + 2\sigma = 2.8015 + 1.3210 = 4.1225$$

- 
7. Convertir los valores logarítmicos de nuevo a su forma original, calculando el antilogaritmo, e elevado al valor del logaritmo, para determinar los valores medios de estos rangos:

$$VS = e^{\ln(VS)} = e^{1.4805} = 4.3953$$

$$S = e^{\ln(S)} = e^{2.1410} = 8.5081$$

$$M = e^{\ln(M)} = e^{2.8015} = 16.4696$$

$$L = e^{\ln(L)} = e^{3.4620} = 31.8811$$

$$VL = e^{\ln(VL)} = e^{4.1225} = 61.7137$$

---



# Instrucciones

---

## Instrucciones

Antes de comenzar el programa 4, repasar el proceso PSP0.1 para asegurarse de comprenderlo. También, asegurarse de tener todas las entradas requeridas antes de comenzar con la fase del planificación.

---

## Entregables

Cuando complete la etapa de postmortem, arme un archivo .zip para enviar al instructor, conteniendo lo siguiente.

- El archivo .mdb con sus datos del ejercicio.
- Código fuente del programa.
- Impresión de pantalla de las pruebas realizadas mostrando el resultado de la mismas.
- Captura de pantalla de la salida del contador de LOC aplicada al ejercicio 4.
- Archivo de texto o documento que contenga la descripción de que clases/módulos/unidades del código construido contienen las distintas categorías (Added, Modified, Delete, etc). Utilizando el contador de LOC que muestra el tamaño de dichas unidades de código.

Ejemplo:

36 LOC Base -> Clase Base.java y muestre el tamaño de la clase utilizando el contador de LOCS aplicado a la clase.

3 LOC Deleted -> En la clase Base.java

10 LOC Modified -> En la clase Base.java

15 LOC Added -> En la clase Base.java

40 LOC Reused -> Clases LibUno.java, LibDos.java y muestre el tamaño de la las mismas utilizando el contador de LOCS aplicado ambas clases.

5 LOC New Reused -> agregadas en NuevaLib.java

### Tenga en cuenta que:

.- Deben programar de acuerdo a su estándar de codificación.

.- Deben utilizar el mismo proceso que en el ejercicio anterior, PSP 0.1. Esto quiere decir que en planificación van a estimar la cantidad de LOC que esperan generar y si parten de algún LOC Base van a indicar cuántas líneas son. Luego, en postmortem, van a usar el contador de LOC que construyeron para contar la cantidad de LOC del ejercicio 4.

---

## Criterios de evaluación para el programa 4

---

### Criterios de Evaluación

Su reporte debe ser

- completo

Los datos de su proceso deben ser

- exactos
  - precisos
  - consistentes
- 

### Sugerencias

Recuerde entregar su reporte en fecha.

Mantenga la simplicidad del programa.

Si no está seguro de algo, consulte a su instructor

Debe producir y reportar sus propias estimaciones y datos reales, desarrollar su propio código y llevar adelante su propio juego de pruebas.