

PROCEDIMIENTOS Y FUNCIONES

AUTOR: WALTER MADRIGAL CHAVES

NOVIEMBRE: 2020



San Marcos

Contenido

INTRODUCCIÓN.....	2
PROCEDIMIENTOS	3
VENTAJAS	3
CONSTRUCCIÓN DE UN PROCEDIMIENTO.....	4
LLAMADO A UN PROCEDIMIENTO	5
FUNCIONES	6
CONSTRUCCIÓN DE UNA FUNCIÓN	6
MACRO ALGORITMO	7
VARIABLES GLOBALES Y LOCALES	8
CONCLUSIONES Y RECOMENDACIONES	10
REFERENCIAS BIBLIOGRÁFICAS.....	10



INTRODUCCIÓN

En la vida cotidiana la resolución de problemas de cualquier índole se facilita considerablemente si se dividen en problemas más pequeños, este razonamiento no es ajeno al desarrollo de sistemas, en donde los algoritmos se dividen en subalgoritmos dando origen a la programación modular.

Este paradigma de la programación busca ser más eficiente, ofreciendo más responsabilidades a pequeños bloques de código llamados módulos. Algo importante es que cada módulo contiene todo lo necesario para cumplir con su propia funcionalidad y se pueden editar o modificar sin que se vea afectado directamente el resto del proyecto.

Aplicar este tipo de programación resulta muy útil a la hora de programar grandes proyectos, ya que se crea una inmensa biblioteca de funciones y procedimientos, para tareas que se utilizan con frecuencia dentro del programa.

En esta lectura vamos a analizar los procedimientos y funciones que son la base principal en la programación modular.

PROCEDIMIENTOS

Según Herrera (2017), un procedimiento es un conjunto finito de instrucciones con un único propósito bien definido (responsabilidad) y que puede ser invocado (utilizado) por medio de un nombre que lo identifica de manera única.

Como bien lo indica el autor del libro, son subalgoritmos que se crean para resolver un problema específico, por lo general ese problema se presenta en varias ocasiones durante la vida del sistema, por lo cual, el procedimiento se puede invocar las veces que sea necesario. Otra característica importante de esta estructura es que no retorna un valor, a diferencia de su pariente cercano las funciones, esto se analizará más adelante.

VENTAJAS

Las siguientes son algunas de las ventajas de usar procedimientos:

- Al descomponer un algoritmo en partes de menor tamaño, será más fácil la trazabilidad, detectar errores y poner en práctica mejoras.
- La reutilización de código, esto implica que cada vez que necesitemos el procedimiento solamente lo invocamos y no es necesario escribirlo de nuevo.
- Por su independencia se pueden hacer cambios en su estructura sin afectar directamente el resto del programa.
- El código fuente del sistema será más ordenado y entendible.
- Facilita la división de las tareas entre un equipo de programadores.

CONSTRUCCIÓN DE UN PROCEDIMIENTO

Para la construcción de un procedimiento es necesario declararlo, así como lo haría con cualquier otra variable, esta declaración lo que hace es indicarle al algoritmo central cual es el bloque de instrucciones y donde están localizadas, para que sea invocado cuando sea necesario su ejecución.

El procedimiento tiene una estructura bien definida, algunos de sus partes son:

- **Nombre:** Lo ideal es que el nombre sea bien representativo de la acción o propósito del procedimiento. Por lo general se utiliza un verbo y algún complemento que da sentido al verbo, por ejemplo, imprimirNota, calcularImpuesto, ordenarDatos, etc.
- **Parámetros:** los procedimientos en muchas ocasiones necesitan de información externa para realizar su trabajo, estos datos son suministrados a través de los parámetros, que no son más que variables que solo existen dentro del procedimiento. Se pueden tener muchos parámetros o ninguno dependiendo la complejidad o necesidad.
- **Cuerpo:** Son el conjunto de código que se ejecutará dentro del procedimiento.

La estructura general de un procedimiento se vería de la siguiente manera:

Imagen 1 Estructura de un procedimiento

```

1  Procedimiento calcularSalario ([lista de parametros])
2      instrucción 1
3      instrucción 1
4      ....
5      instrucción final
6  FinProcedimiento
  
```

Fuente: Elaboración propia

LLAMADO A UN PROCEDIMIENTO

El llamado o invocación del procedimiento se puede realizar desde cualquier parte del algoritmo, lo que esto implica es que cuando el algoritmo llegue a la línea de invocación, se ejecutaran todas las instrucciones definidas dentro del procedimiento. La invocación de un procedimiento puede ser realizada incluso dentro de él mismo.

Al momento de hacer el llamado al procedimiento es necesario asignarle los datos reales mediante los parámetros. La instrucción quedaría de la siguiente manera:

Imagen 2 Procedimiento que calcula salario

```

1  Procedimiento calcularSalario (Cadena nombre, Real salario, Entero diasLaborados)
2      Real total
3      total=salario*diasLaborados
4      imprimir ("el salario de", nombre)
5      imprimir("es: ", total)
6  FinProcedimiento
7
8
  
```

Fuente: Elaboración propia

Imagen 3 Llamado al procedimiento calculaSalario

```

1  Procedimiento calcularSalario ("Ana", 15.000, 5)
2
  
```

Fuente: Elaboración propia

Como se muestra en la imagen 2, se creó un procedimiento que recibe tres parámetros, el primero de tipo cadena que representa el nombre, el segundo tipo real que representa el salario y el tercer tipo entero que representa los días laborados.

Dentro del procedimiento, se declara una variable local llamada "total" con la finalidad de que almacene el resultado de la multiplicación entre el salario y los días laborados, por último, se imprimen las variables cargadas con los datos.

En la imagen 3 se invoca al procedimiento calculaSalario que se acaba de analizar, nótese, que se le asigna los tres valores que llenaran los parámetros que el procedimiento solicita. Muy importante los parámetros de tipo cadena deben ir entre comillas.

FUNCIONES

Al igual que los procedimientos las funciones juegan un papel muy importante en desarrollo de sistemas, así lo indica Trejos (2017), indicando, ningún concepto a nivel de la programación es más importante que el concepto de función. Sin temor a equivocarme puedo garantizarle que la función es lo que podríamos llamar la gran "vedette" de la programación.

Las funciones son un tipo especial de procedimiento, por lo que conserva sus mismas características, estructura y propiedades, lo que diferencia a uno del otro es que la función tiene la capacidad de entregar un valor como respuesta.

CONSTRUCCIÓN DE UNA FUNCIÓN

La construcción de una función es muy parecida a la de procedimientos, se utiliza la palabra reservada "Funcion" (sin tilde), posteriormente se asigna una variable que permitirá el retorno de datos, luego el nombre, este se recomienda

que sea bastante representativo y al final los parámetros necesarios para recibir datos del entorno.

En la estructura también hay un pequeño cambio, es la creación de un nuevo elemento que sirve como variable retorno "return", viene implícita en la estructura de la función y sirve para indicar cual es el valor que se entrega como resultado. Este valor debe coincidir con el tipo de dato que se indicó en la función, justo antes de su nombre.

Imagen 4 Función para calcular salario

```

1 Funcion Real calcularSalario (Real salario, Entero diasLaborados)
2     Real total
3     total=salario*diasLaborados
4     return total
5 FinFuncion
  
```

Fuente: Elaboración propia

Nota: La invocación de las funciones es la misma utilizada en los procedimientos.

MACRO ALGORITMO

Para Trejos (2017), el macro algoritmo es un gran algoritmo dividido en unidades funcionales en donde cada una de ellas logra un pequeño objetivo dentro de toda la solución y en conjunto dichas unidades logran el objetivo general.

En esta técnica o paradigma se aplica el dicho divide y vencerás, en donde varias subrutinas se enfocan en pequeños problemas para solucionar un gran problema.



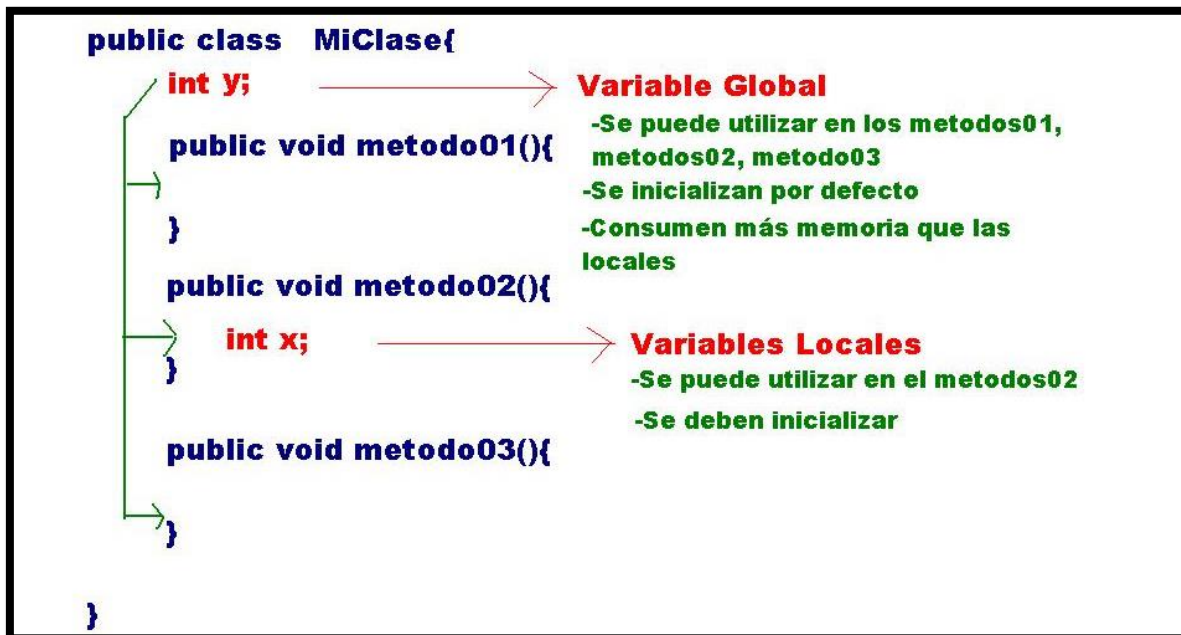
VARIABLES GLOBALES Y LOCALES

Con la programación modular y el uso de procedimientos y funciones, surge a la palestra un nuevo concepto, el ámbito de acción de las variables. Este término se refiere a la zona del algoritmo en donde se va a reconocer la influencia de la variable. Existen dos tipos, las variables locales y las globales.

Las variables locales tienen un ámbito de acción que se restringe a la función o procedimiento en la que se declaró. Esto implica que la variable sólo va a poder ser manipulada en dicha sección, y no se podrá hacer referencia en otra parte del algoritmo. Cualquier variable que se defina dentro de las llaves del cuerpo de una función se interpreta como una variable local a esa función.

Las variables globales son aquellas que se definen normalmente al principio del programa y fuera del cuerpo de cualquier función o procedimiento. El ámbito de una variable global son todas las funciones que componen el programa, cualquier función o procedimiento puede acceder a dichas variables para leer y escribir en ellas.

Imagen 5 Variables locales y globales



Fuente: <http://aprendiendoconjava.blogspot.com>

*Para ahondar más en el tema de programación modular deben realizar la lectura de las páginas 361 a la 410 del libro: **Lógica de programación (2017)** de Omar Iván Trejos Buriticá.*

CONCLUSIONES Y RECOMENDACIONES

- La programación modular es un paradigma ampliamente usado en el mercado, se considera que es la evolución de la programación estructurada.
- Se considera que las funciones y procedimientos son las estructuras más importantes del mundo de la programación, es gracias a ellos que se pueden aplicar el fraccionamiento de tareas, lo que conlleva a resolver grandes problemas, dividir recursos y maximizar esfuerzos.
- Durante la concepción y desarrollo de una función se debe pensar en forma genérica, entre más genérica se haga su estructura mejor se adaptará a necesidades futuras.
- Recuerde que lo visto en esta lectura es la concepción de una función y procedimiento en su forma general, cada lenguaje de programación tiene sus propias reglas y sintaxis.

REFERENCIAS BIBLIOGRÁFICAS

- Buriticá, O. T. (2017). *Lógica de programación*. Bogotá: Ediciones de la U.
- Herrera Morales, J., & Gutiérrez Posada, J. (2017). *Introducción a la lógica de programación*. Armenia: ELIZCOM S.A.A.
- Neira, A. (04 de 11 de 2020). *Apuntes de Programación* . Obtenido de <http://programacionjand.blogspot.com/p/arreglos-vectores-y-matrices-en-c.html>



www.usanmarcos.ac.cr

San José, Costa Rica