

Estructuras de control de flujo de datos

Simple: El condicional simple *Si – Entonces – Fsi* ejecuta un conjunto de instrucciones si cumple la condición evaluada en el Si.

Sintaxis: If (<expresión lógica>) {
 <Bloque de instrucción S1>
}

Compuesto: Permite elegir entre dos opciones o alternativas posibles, en función de que se cumpla o no la condición expresada en el Si.

Sintaxis: If (<expresión lógica>) {
 <Bloque de instrucción S1>
} else {
 <Bloque de instrucción S2>
}

Anidado: Permite incluir dentro del cuerpo de la instrucción Si a otras instrucciones Si simples o compuestas. Se utiliza para elegir entre varias opciones o alternativas en función del cumplimiento o no de las diferentes condiciones que se van verificando en cada instrucción Si.

Selección Múltiple (Switch): La selección múltiple permite evaluar una condición o expresión que puede tomar muchos valores distintos. Se ejecutarán las instrucciones correspondientes al caso que se cumple. La principal ventaja de la estructura Selección es que permite crear algoritmos que sean legibles y evitar la confusión creada por el anidamiento de muchos bloques SI.

Estructuras de control de flujo de datos

For: Es una estructura iterativa que es controlada por una variable (llamada también variable índice), la cual se incrementa o decrementa hasta llegar a un valor límite o valor final que representa la condición de parada.

Sintaxis: for (i=0; i <=10; i++) {
 <Bloque de instrucción S1>
}

While: Es una estructura iterativa que permite verificar la condición de entrada al ciclo antes del cuerpo de instrucciones a repetir. Como la evaluación de la condición de entrada se realiza al inicio del bloque while, puede ocurrir que las instrucciones del ciclo no se realicen ni siquiera 1 vez, a diferencia del do...while, donde el bloque de instrucciones se realiza al menos 1 vez porque la condición de parada se verifica al final.

Sintaxis: While (<expresión lógica>) {
 <Bloque de instrucciones>
}

Do While: Ejecuta un bloque de instrucciones varias veces hasta que se cumple la condición que es verificada al final del bucle. Las instrucciones dentro del ciclo do...while se van a realizar mientras la condición de parada evaluada al final sea verdadera. Dicho de otro modo, el ciclo se va a detener cuando la condición de parada se haga falsa.

Sintaxis: Do {
 <Bloque de instrucciones>
} while(<expresión lógica>);

Procedimientos

La definición de procedimientos permite asociar un nombre a un bloque de instrucciones. Luego podemos usar ese nombre para indicar en algún punto de un algoritmo que vamos a utilizar ese bloque de instrucciones, pero sin tener la necesidad de repetirlas, sólo invocando al procedimiento por su nombre.

Acciones: Conjunto de instrucciones con un nombre que pueden ser llamadas a ejecución cuando sea necesario. No retornan valores.

Sintaxis:
Def<nombre> (<lista parámetros>) {
 <Bloque de instrucciones>
}

Funciones: Al igual que las acciones con conjuntos de instrucciones con un nombre, pero se caracterizan por retornar (enviar o devolver) un valor al algoritmo que la llama.

Sintaxis:
<tipo dato retorno> <nombre>(<lista parámetros>) {
 <Bloque de instrucciones>
 return(valor a retornar);
}