



CIMAT

Informática Aplicada I

Marcela Morales Quispe

CIMAT

Septiembre 4, 2013

marcelamq@cimat.mx

- 1 Operadores aritméticos
- 2 Presidencia de operadores
- 3 Tomando decisiones
- 4 Estructura de repetición
- 5 Ejercicios

Operadores aritméticos

A manera de resumen se presenta la siguiente tabla que contiene los diferentes operadores aritméticos en C, todas éstas están definidas para los números enteros y flotantes, a excepción de la operación módulo ya que los operandos para éste deben de ser de tipo entero. Los operadores aritméticos son todos binarios.

Operación en C	Operador	Ejemplo	Expresión en C
Suma	+	$f + 7$	$f + 7$
Substracción	-	$b - a$	$b - a$
Multiplicación	*	ax	$a * x$
División	/	$\frac{x}{y}$	x / y
Módulo	%	$r \bmod s$	$r \% s$

Tabla 1: Operadores aritméticos de C.

Recuerda siempre que la *división de enteros* da como resultado también un entero.

Presidencia de operadores

Las reglas de presidencia de operadores son guías de acción, que le permiten a C calcular (evaluar) expresiones en el orden correcto. Cuando decimos que una evaluación o cálculo se realiza de izquierda a derecha, nos estamos refiriendo a la *asociatividad* de los operadores. En la tabla 2 se resumen estas reglas.

Operador(es)	Orden de cálculo (presidencia)
()	Se calculan primero. Si los paréntesis están anidados la expresión en el par más interno se evalúa primero. Si existen varios pares de paréntesis que no estén anidados, se evaluarán de izquierda a derecha.
*, /, %	Se evalúan en segundo lugar. Si existen varias, se calcularán de izquierda a derecha.
+, -	Se calculan al último. Si existen varios, serán evaluados de izquierda a derecha.

Tabla 2: Presidencia de operadores aritméticos.

Operadores de igualdad y relacionales

Esta sección introduce una versión simple de la *estructura de control if* de C, que permite a un programa tomar una decisión basada en la veracidad o falsedad de alguna declaración, conocida como *condición*. Las condiciones en las estructuras **if** se forman utilizando los *operadores de igualdad* y los *operadores relacionales* resumidos en la tabla 3.

Operadores algebraicos	Operadores de C	Ejemplo en C	Significado de la condición
=	==	x == y	x es igual a y
≠	!=	x != y	x no es igual a y
>	>	x > y	x es mayor que y
<	<	x < y	x es menor que y
≥	>=	x >= y	x es mayor que o igual a y
≤	<=	x <= y	x es menor que o igual a y

Tabla 3: Operadores de igualdad y relacionales.

Los operadores relacionales tienen un mismo nivel de presedencia y se asocian de izquierda a derecha. Los operadores de igualdad tienen un nivel de precedencia menor que los operadores relacionales y también se asocian de izquierda a derecha.

Sentencia **if**

```
if(condición)
{
    sentencia-1;
    sentencia-2;
    sentencia-n;
}
else
{
    sentencia-4;
    sentencia-5;
    sentencia-m;
}
```

Una buena práctica de programación es escribir las sentencias en cada línea con una adecuada indentación (sangría), con esto resulta fácil identificar que `sentencia-1;`, `sentencia-2;` y `sentencia-n;` se ejecutan si la condición es verdadera.

Sentencia **while**

Una *estructura de repetición* le permite al programador especificar que se repita una acción, en tanto cierta condición se mantenga verdadera.

while(condición)

```
{  
    sentencia-1;  
    sentencia-2;  
    sentencia-n;  
}
```

Eventualmente, la condición se hará falsa, llegado este punto, la repetición termina, y se ejecutarán las sentencias que siguen inmediatamente después de la estructura de repetición.

Observa que las sentencias dentro del cuerpo de **if**, **else** y **while** pueden ser nuevamente estas mismas sentencias, de tal forma que es posible construir sentencias anidadas y por ende, mucho más complejas.

- 1 Convetidor de temperatura (de grados centígrados a fahrenheit).
- 2 Dados dos enteros x, y , avisar si x divide a y .
- 3 Una clase de diez alumnos hizo un examen. Las calificaciones (enteros en el rango 0 a 100) correspondientes a este examen están a su disposición. Determine el promedio de la clase en este examen.
- 4 Desarrolle un programa de promedios de clase que pueda procesar un número arbitrario de calificaciones, cada vez que se ejecute el programa.
- 5 Dado un entero positivo n , regresar los $n - esimos$ números de Fibonacci.
- 6 Dado un entero positivo n , imprimir en pantalla el triángulo de Pascal de orden n .