

Principios y Herramientas de Programación

Dra. Jessica Andrea Carballido

jac@cs.uns.edu.ar

```
opcion;
printf("1. Capital de Argentina\n");
printf("2. Capital de España\n");
printf("3. 10000+58000 = ?\n");
printf("4. Capital de Uruguay\n");
scanf("%i",&opcion);
switch(opcion)
{
case 1:
printf("\n\nBuenos Aires");
break;
case 2:
printf("\n\nMadrid");
break;
case 3:
printf("\n\n68000");
break;
case 4:
printf("\n\nMontevideo");
break;
default:
printf("\n\nOpcion erronea. Intenta
```

Dpto. de Ciencias e Ingeniería de la Computación

UNIVERSIDAD NACIONAL DEL SUR

Datos

Los algoritmos combinan **datos** con **acciones**.

Los datos de entrada son utilizados y/o transformados mediante las acciones para así obtener los datos de salida, y en estos datos de salida se encuentra representada la solución buscada.

Algoritmo Perímetro de un Cuadrado

Dato de entrada: lado

Dato de salida: perimetro

Acción: perimetro = $\text{lado} * 4$ ← expresión



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Datos

Los datos pertenecen a distintos dominios (tipos) y se combinan mediante **EXPRESIONES**.

Las expresiones están compuestas básicamente de operandos y operadores.

De acuerdo al dominio al cual pertenecen estos elementos las expresiones pueden ser, en principio, de tipo ARITMETICAS o LOGICAS.

Algoritmo Area de rectangulo

Dato de entrada: base, altura

Dato de salida: area

Acción: $\text{area} = \text{base} * \text{altura}$ ← expresión



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Expresiones



Aritméticas: Los operandos que intervienen en ella son numéricos, el resultado es numérico y los operadores son aritméticos.

Lógicas: Su resultado es VERDADERO ó FALSO. Se construyen mediante los operadores relacionales y lógicos.

FALSE
true



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Expresiones Aritméticas



El resultado de la evaluación de la expresión es un valor numérico.

Operadores aritméticos más utilizados son: + , - , * , /.

<i>Operadores aritméticos en pseudocódigo:</i>	
+	Suma
-	Resta
*	Multiplicación
**	Potencia (también se utiliza el carácter <i>flecha arriba</i> ↑ o el carácter <i>acento circunflejo</i> ^)
/	División real
div	División entera (también se utiliza el carácter <i>barra invertida</i> \)
mod	Módulo (resto de la división entera)
<hr/>	
+	Signo más
-	Signo menos



Dra. J.

CONICET - DCIC (UNS)



ritmos



Expresiones Aritméticas



Precedencia de operadores

En una expresión puede aparecer más de un operador.

$11 + 9 \text{ div } 3$ (dos operadores)
 $-3 * 6 \text{ mod } 4$ (tres operadores)
 $-3.1 + 5 * 0.5$ (tres operadores)
 $3 ** 3 - 1$ (dos operadores)
 $+3 * -8$ (tres operadores)

Para poder evaluar correctamente las expresiones aritméticas del ejemplo, es necesario seguir un criterio de **precedencia de operadores.**



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Expresiones Aritméticas



Precedencia de operadores

La precedencia en el pseudocódigo suele ser como la definida para las expresiones matemáticas.

<i>Prioridad de los operadores aritméticos (de mayor a menor) en pseudocódigo:</i>	
<code>+ -</code>	Signos más y menos
<code>**</code>	Potencia
<code>* / div mod</code>	Multiplicación, división real, división entera y módulo
<code>+ -</code>	Suma y resta

© carlospes.com



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algunos

Expresiones Aritméticas



Precedencia de operadores

Sin embargo, al llegar el momento de programar, cada lenguaje puede definir su propia tabla de precedencia de operadores.

¡Tenerlo muy en cuenta al momento de aprender un nuevo lenguaje de programación!



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Expresiones Lógicas

El resultado de la evaluación de la expresión es un valor lógico (verdadero o falso).

Se combinan:

- Operadores relacionales
- Operadores lógicos



FALSE
true



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos



Expresiones Lógicas

FALSE
true

Un **operador relacional**
se utiliza para comparar
los valores de **dos expresiones del mismo tipo.**

Operadores relacionales en pseudocódigo:

<	Menor que
<=	Menor o igual que
>	Mayor que
>=	Mayor o igual que
=	Igual que
<>	Distinto que

Valores de tipo lógico (**verdadero** y **falso**) → **falso < verdadero.**

Valores de tipo carácter → orden del [ASCII](#) extendido.

Valores de tipo cadena → orden del [ASCII](#) extendido.



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Expresiones Lógicas

Por otra parte, un **operador lógico** actúa exclusivamente sobre valores de **expresiones lógicas**.

<expresión_1>	<expresión_2>	<expresión_1> y <expresión_2>
verdadero	verdadero	verdadero
verdadero	falso	falso
falso	verdadero	falso
falso	falso	falso

<expresión_1>	<expresión_2>	<expresión_1> o <expresión_2>
verdadero	verdadero	verdadero
verdadero	falso	verdadero
falso	verdadero	verdadero
falso	falso	falso

<expresión>	no <expresión>
verdadero	falso
falso	verdadero

FALSE
true

Operadores lógicos en pseudocódigo:

y Conjunción

o Disyunción

no Negación



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Finitos
Generales
Deterministas

Acciones:

- Asignación
- Lectura y escritura de datos

Se organizan con las estructuras de control:

- Secuencia
- Condicional
- Iteración

Se ejecutan de manera secuencial.



Asignación

Una asignación nos permite asociar un VALOR a un determinado DATO.

Por ejemplo:

Edad \leftarrow 15

Mediante esta acción de asignación estamos dándole al dato Edad el valor 15.



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Asignación

- A la izquierda de la asignación siempre aparece el nombre que identifica al dato.
- Del lado derecho de la asignación se utiliza un valor literal o una expresión en la que pueden intervenir otros datos o valores literales.
- Los dominios a los cuales pertenecen ambos lados de la asignación deben ser **compatibles**.

Ejemplo:

Precio ← PrecioContado - Descuento



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Secuencia

Las acciones se llevan a cabo de manera SECUENCIAL.

Precio Lista \leftarrow 160

Descuento \leftarrow 10

Precio Contado \leftarrow Precio Lista – Descuento

En este ejemplo se ven tres asignaciones, y las mismas se llevan a cabo en el orden en el que aparecen.



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Condicional

A veces hay que escoger entre una o mas alternativas de acción posibles.

La estructura de control **CONDICIONAL** nos permite tomar distintos cursos de acción de acuerdo a una condición, que se expresa mediante una expresión lógica.

Si paga de contado tendrá un descuento, sino no lo tendrá.



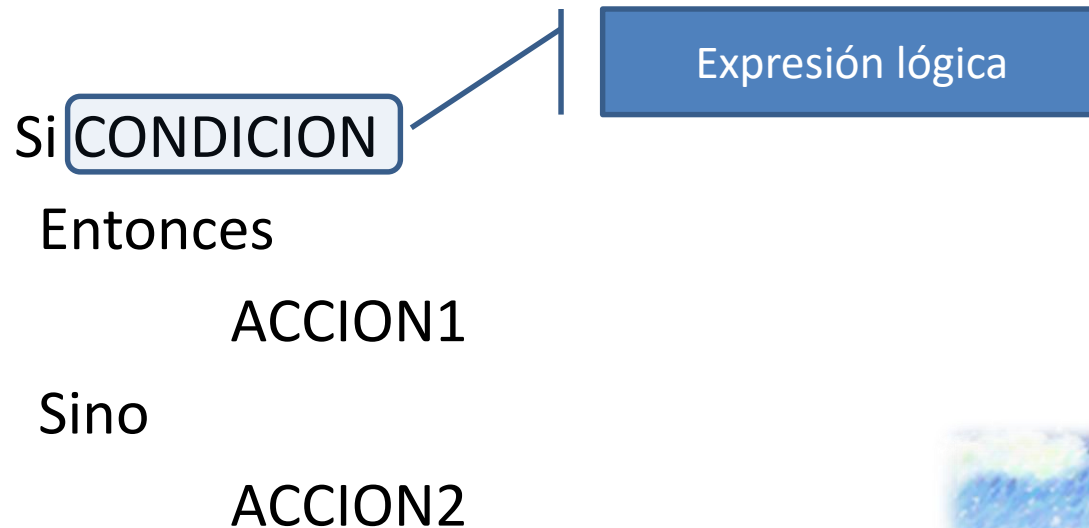
Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Condicional

Forma de expresarlo en pseudocódigo:



Condicional

Ejemplos:

...

Si $\text{precio} < 100$

Entonces

$\text{PorcDescuento} \leftarrow 0$

Sino

$\text{PorcDescuento} \leftarrow 5$



Condicional

Condicionales ANIDADOS

Si precio < 100

Entonces

PorcDescuento \leftarrow 0

Sino

Si precio < 1000

Entonces

PorcDescuento \leftarrow 5

Sino

PorcDescuento \leftarrow 10



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Condicional

Condicional con múltiples opciones

El valor de un DATO se compara con diferentes valores hasta encontrar el que coincide. Para cada valor se define una acción diferente.

En caso de DATO

Valor1: accion1

Valor2: accion2

...

ValorN: accionN



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Condicional

Condicional con múltiples opciones

Ejemplo:

En caso de EstadoCivil

“soltero”: $\text{solteros} \leftarrow \text{solteros} + 1$

“casado”: $\text{casados} \leftarrow \text{casados} + 1$

“divorciado”: $\text{divs} \leftarrow \text{divs} + 1$

“viudo”: $\text{viudos} \leftarrow \text{viudos} + 1$

“otro”: $\text{otro} \leftarrow \text{otro} + 1$



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)

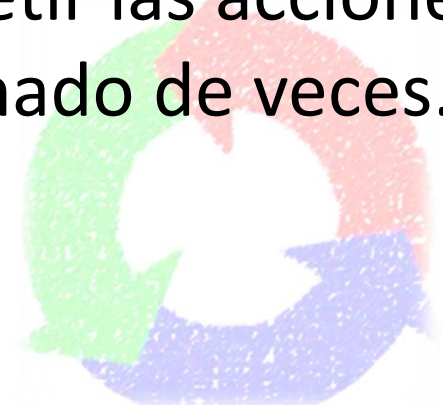


Iteración

Hasta ahora podríamos resolver problemas donde las acciones se ejecutan sólo una vez.

¿Qué pasa si necesito repetir las acciones?

La estructura de control ITERATIVA nos permite indicar que queremos repetir las acciones un número determinado o indeterminado de veces.



Iteración

Hay dos tipos de estructuras de repetición:

- La primera es aquella en donde el número de veces que un grupo de acciones se van a ejecutar depende de un valor numérico.
 - Repetir N veces ACCIONES



Repetir 6 veces
Agregar huevo



Iteración

Hay dos tipos de estructuras de repetición:

- La segunda es aquella en la cual el número de repeticiones es desconocido, se repetirá hasta que se cumpla o no cierta condición.

Dos alternativas:

- Mientras **CONDICION** hacer **ACCIONES**
- Repetir **ACCIONES** hasta **CONDICION**



Iteración

Ejemplos:

suma \leftarrow 0

Repetir 4 veces

suma \leftarrow suma + 2

¿Cual es el resultado de ejecutar estas acciones?



Iteración

Ejemplos:

Producto $\leftarrow 1$

Repetir 3 veces

Producto \leftarrow Producto * 10

¿Cual es el resultado de ejecutar estas acciones?



Iteración

Ejemplos:

Suma \leftarrow 0

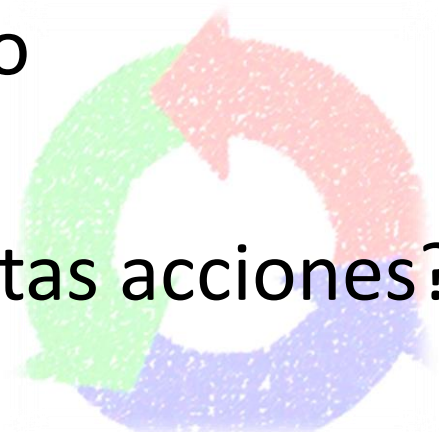
Numero \leftarrow 3

Mientras (Numero > 0) hacer

Suma \leftarrow Suma + Numero

Numero \leftarrow Numero - 1

¿Cual es el resultado de ejecutar estas acciones?



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Estructuras de control

Secuencia

Accion1
Accion2
...
AccionN

Condicional

Si CONDICION
Entonces
 Acciones en secuencia1
Sino
 Acciones en secuencia2

Iteración

Mientras CONDICION
 Acciones en secuencia

Repetir
 Acciones en secuencia
Hasta CONDICION

Repetir N veces
 Acciones en secuencia



Conceptos introducidos en esta clase:

- ✓ ALGORITMO
- ✓ DATO
- ✓ EXPRESION
- ✓ ASIGNACION
- ✓ ESTRUCTURAS DE CONTROL



¡HASTA LA PRÓXIMA CLASE!



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Extras

Para ver en casa



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Expresiones Aritméticas



Ejemplo 1:

El operador multiplicación (*) realiza la multiplicación de dos operandos numéricos.

$$5 * 2$$

$$3.1 * 2.5$$

Los resultados son:

10 (valor entero)

7.75 (valor real)



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Expresiones Aritméticas



Ejemplo 2:

El operador potencia (**) eleva el operando de la izquierda (número base) al operando de la derecha (exponente).

$$5 ** 2$$

$$3.1 ** 2.5$$

De estas expresiones, se obtienen los valores:

25 (valor entero)

16.920151 (valor real)



Expresiones Aritméticas



Ejemplo 3:

El operador división real (/) realiza la división real entre dos operandos numéricos.

$$5 / 2$$

$$3.1 / 2.5$$

Sus resultados son:

2.5 (valor real)

1.24 (valor real)



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Expresiones Aritméticas



Ejemplo 4:

El operador división entera (div) realiza la división entera entre dos operandos numéricos enteros.

5 div 2

3.1 div 2.5

Resultados:

2 (valor entero)

ERROR (no se puede evaluar; ambos operandos deben ser valores enteros)



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Algoritmos

Expresiones Aritméticas



Ejemplo 5:

El operador módulo (mod) realiza la división entera entre dos operandos numéricos enteros, devolviendo el resto de la misma.

$5 \text{ mod } 2$

$3.1 \text{ mod } 2.5$

Resultados:

1 (valor entero)

ERROR (tampoco puede operar con operandos reales)



Dra. Jessica Andrea Carballido
CONICET - DCIC (UNS)



Algoritmos

Expresiones Aritméticas



Ejemplo 6:

El operador signo menos (-) cambia el signo de un operando numérico. Así, de las expresiones

-11

$-(3.1)$

$-(-2.5)$

se obtienen los valores:

-11 (valor entero)

-3.1 (valor real)

2.5 (valor real)



Dra. Jessica Andrea Carballido

CONICET - DCIC (UNS)



Algoritmos

Iteración

Ejemplos:

Y si la condición fuera:
(Tope \leq 1) ??

Tope \leftarrow 0

Mientras (Tope $>$ 0) hacer

Tope \leftarrow Tope -1

¡Las acciones pueden no ejecutarse nunca!



Iteración

Ejemplos:

Y si la cantidad vale
100 al comenzar??

Cantidad \leftarrow 0

Repetir

Cantidad \leftarrow Cantidad + 1

Hasta Cantidad = 100

En el caso de esta estructura de control,
las acciones se ejecutan siempre por lo menos una vez.

