

TEMA A

Ejercicio 1

Traducí las expresiones en formalismo básico de la tabla a lenguaje C y construí un programa que solicite al usuario el ingreso de los valores para las variables x , y , z e imprima el resultado dichas expresiones:

Expresión	$\llbracket x \rightarrow 3, y \rightarrow 6, z \rightarrow \text{True} \rrbracket$	$\llbracket x \rightarrow 0, y \rightarrow 3, z \rightarrow \text{False} \rrbracket$
$x = D_1 + D_2 \vee y = D_1 + D_2 \vee z$		
$(D_1 + D_2 + x) \bmod 3$		
$x * D_1 + y * D_2$		
$x \neq 0 \wedge 0 \leq (D_1 - D_2) / x \wedge \neg z$		

Antes de traducir, reemplazá las constantes D_1 y D_2 por tu última y penúltima cifra de tu número de DNI, respectivamente. Entonces si tu número de DNI es 40.123.789 las expresiones quedarían:

$x = 9 + 8 \vee y = 9 + 8 \vee z$
$(9 + 8 + x) \bmod 3$
$x * 9 + y * 8$
$x \neq 0 \wedge 0 \leq (9 - 8) / x \wedge \neg z$

Compilá el programa y completá los resultados de la tabla para los estados especificados.



Pensá si podés usar alguna de las funciones que usaste o alguna parte del código del Proyecto 3.

Ejercicio 2

Siendo D_1 y D_2 los dos últimos números de tu DNI ($XX.XXX.XD_1D_2$), programá en lenguaje C una traducción del siguiente programa:

```
var x, y, dni1, dni2 : Int;
 $\llbracket \sigma_0 : (x \mapsto 7, y \mapsto 4, \text{dni1} \mapsto D_1, \text{dni2} \mapsto D_2) \rrbracket$ 
if dni1 mod 2  $\neq$  0  $\wedge$  dni2 mod 2 = 0  $\rightarrow$  x, y := x + dni1, y + x
[] dni1 mod 2 = 0  $\vee$  dni2 mod 2  $\neq$  0  $\rightarrow$  x := y + x
fi
 $\llbracket \sigma_1 : (x \mapsto ?, y \mapsto ?, \text{dni1} \mapsto D_1, \text{dni2} \mapsto D_2) \rrbracket$ 
```

Asegurarse que los valores iniciales de x , y , dni1 , dni2 se correspondan con el estado σ_0 . El programa debe mostrar por pantalla los valores finales de x e y .

Ejercicio 3

Traducí a lenguaje C y compilá el siguiente programa. Luego, completá los estados σ^1_1 , σ^2_1 , σ^3_1 , σ^4_1 , σ^5_1 ; donde cada estado a completar es el resultado de realizar 1, 2, 3, 4 y 5 iteraciones del ciclo. Recordá, una iteración es la ejecución completa del cuerpo del ciclo.

```
var x, y, i, res : Int;
[[ $\sigma_0$  : (x  $\mapsto$  5, y  $\mapsto$  0, i  $\mapsto$  0, res  $\mapsto$  0)]]
i, y := 1, 0
do i  $\neq$  x + 1  $\rightarrow$ 
  i, y := i + 1, y + i
  if i mod 2 = 0  $\rightarrow$  y := y - 1
  [] i mod 2  $\neq$  0  $\rightarrow$  skip
fi
[[ $\sigma^1_1$ ,  $\sigma^2_1$ ,  $\sigma^3_1$ ,  $\sigma^4_1$ ,  $\sigma^5_1$ ]]
od
res := y * 2 + x mod 2
```

El valor inicial de la variable `x` se le debe solicitar al mediante la función:

```
int pedirEntero(void)
```

y se debe mostrar por pantalla el valor final de `res` usando la función

```
void imprimeEntero(int x)
```

ambas funciones definidas en el ejercicio 6 del Proyecto 3

Ejercicio 4*

Escribir un programa en C que sea equivalente al del Ejercicio 3 (en el estado final, la variable `res` de ambos programas deben tener el mismo valor) pero que no utilice ningún ciclo (ni recursión). El programa debe permitir al usuario ingresar los valores iniciales y mostrar el resultado por la pantalla.

TEMA B

Ejercicio 1

Traducí las expresiones en formalismo básico de la tabla a lenguaje C y construí un programa que solicite al usuario el ingreso de los valores para las variables x , y , z e imprima el resultado dichas expresiones:

Expresión	$\llbracket x \rightarrow 3, y \rightarrow 6, z \rightarrow \text{True} \rrbracket$	$\llbracket x \rightarrow 0, y \rightarrow 3, z \rightarrow \text{False} \rrbracket$
$x = D_1 + D_2 \vee y = D_1 + D_2 \vee z$		
$(D_1 + D_2 + x) \bmod 3$		
$x * D_1 + y * D_2$		
$x \neq 0 \wedge 0 \leq (D_1 - D_2) / x \wedge \neg z$		

Antes de traducir, reemplazá las constantes D_1 y D_2 por tu última y penúltima cifra de tu número de DNI, respectivamente. Entonces si tu número de DNI es 40.123.789 las expresiones quedarían:

$x = 9 + 8 \vee y = 9 + 8 \vee z$
$(9 + 8 + x) \bmod 3$
$x * 9 + y * 8$
$x \neq 0 \wedge 0 \leq (9 - 8) / x \wedge \neg z$

Compilá el programa y completá los resultados de la tabla para los estados especificados.



Pensá si podés usar alguna de las funciones que usaste o alguna parte del código del Proyecto 3.

Ejercicio 2

Siendo D_1 y D_2 los dos últimos números de tu DNI ($XX.XXX.XD_1D_2$), programá en lenguaje C una traducción del siguiente programa:

```
var x, y, dni1, dni2 : Int;
[[σ₀ : (x ↦ 7, y ↦ 4, dni1 ↦ D₁, dni2 ↦ D₂)]]
if dni1 mod 2 ≠ 0 ∧ dni2 mod 2 = 0 → x, y := x + dni1, y + x
[] dni1 mod 2 = 0 ∨ dni2 mod 2 ≠ 0 → x := y + x
fi
[[σ₁ : (x ↦ ?, y ↦ ?, dni1 ↦ D₁, dni2 ↦ D₂)]]
```

Asegurarse que los valores iniciales de x , y , $dn1$, $dn2$ se correspondan con el estado σ_0 . El programa debe mostrar por pantalla los valores finales de x e y .

Ejercicio 3

Traducí a lenguaje C y compilá el siguiente programa. Luego, completá los estados σ^1_1 , σ^2_1 , σ^3_1 , σ^4_1 , σ^5_1 ; donde cada estado a completar es el resultado de realizar 1, 2, 3, 4 y 5 iteraciones del ciclo. Recordá, una iteración es la ejecución completa del cuerpo del ciclo.

```
var x, y, i, res : Int;
[[ $\sigma_0$  : (x ↦ 3, y ↦ 8, i ↦ 0, res ↦ 0)]]
i := 0
do i ≠ x - y ∧ i ≠ y - x →
  if i mod 2 = 0 → x, y := x + 1, y + 1
  [] i mod 2 ≠ 0 → skip
fi
i := i + 1
[[ $\sigma^1_1$ ,  $\sigma^2_1$ ,  $\sigma^3_1$ ,  $\sigma^4_1$ ,  $\sigma^5_1$ ]]
od
res := (x + y) / 2
```

Los valores iniciales de las variables `x`, `y` se le deben solicitar al usuario usando la función:

```
int pedirEntero(void)
```

y se debe mostrar por pantalla el valor final de `res` usando la función

```
void imprimeEntero(int x)
```

ambas funciones definidas en el ejercicio 6 del Proyecto 3

Ejercicio 4*

Escribir un programa en C que sea equivalente al del Ejercicio 3 (en el estado final, la variable `res` de ambos programas deben tener el mismo valor) pero que no utilice ningún ciclo (ni recursión). El programa debe permitir al usuario ingresar los valores iniciales y mostrar el resultado por la pantalla.

TEMA C

Ejercicio 1

Traducí las expresiones en formalismo básico de la tabla a lenguaje C y construí un programa que solicite al usuario el ingreso de los valores para las variables x , y , z e imprima el resultado dichas expresiones:

Expresión	$\llbracket x \rightarrow 3, y \rightarrow 6, z \rightarrow \text{True} \rrbracket$	$\llbracket x \rightarrow 0, y \rightarrow 3, z \rightarrow \text{False} \rrbracket$
$x = D_1 + D_2 \vee y = D_1 + D_2 \vee z$		
$(D_1 + D_2 + x) \bmod 3$		
$x * D_1 + y * D_2$		
$x \neq 0 \wedge 0 \leq (D_1 - D_2) / x \wedge \neg z$		

Antes de traducir, reemplazá las constantes D_1 y D_2 por tu última y penúltima cifra de tu número de DNI, respectivamente. Entonces si tu número de DNI es 40.123.789 las expresiones quedarían:

$x = 9 + 8 \vee y = 9 + 8 \vee z$
$(9 + 8 + x) \bmod 3$
$x * 9 + y * 8$
$x \neq 0 \wedge 0 \leq (9 - 8) / x \wedge \neg z$

Compilá el programa y completá los resultados de la tabla para los estados especificados.



Pensá si podés usar alguna de las funciones que usaste o alguna parte del código del Proyecto 3.

Ejercicio 2

Siendo D_1 y D_2 los dos últimos números de tu DNI ($XX.XXX.XD_1D_2$), programá en lenguaje C una traducción del siguiente programa:

```
var x, y, dni1, dni2 : Int;
 $\llbracket \sigma_0 : (x \mapsto 7, y \mapsto 4, \text{dni1} \mapsto D_1, \text{dni2} \mapsto D_2) \rrbracket$ 
if dni1 mod 2  $\neq$  0  $\wedge$  dni2 mod 2 = 0  $\rightarrow$  x, y := x + dni1, y + x
[] dni1 mod 2 = 0  $\vee$  dni2 mod 2  $\neq$  0  $\rightarrow$  x := y + x
fi
 $\llbracket \sigma_1 : (x \mapsto ?, y \mapsto ?, \text{dni1} \mapsto D_1, \text{dni2} \mapsto D_2) \rrbracket$ 
```

Asegurarse que los valores iniciales de x , y , dni1 , dni2 se correspondan con el estado σ_0 . El programa debe mostrar por pantalla los valores finales de x e y .

Ejercicio 3

Traducí a lenguaje C y compilá el siguiente programa. Luego, completá los estados σ^1_1 , σ^2_1 , σ^3_1 , σ^4_1 , σ^5_1 ; donde cada estado a completar es el resultado de realizar 1, 2, 3, 4 y 5 iteraciones del ciclo. Recordá, una iteración es la ejecución completa del cuerpo del ciclo.

```
var x, y, i, res : Int;
[[ $\sigma_0$  : (x ↦ 5, y ↦ 3, i ↦ 0, res ↦ 0)]]
i := 0
do x ≠ 0 →
  if i < y - 1 → i, x := i + 1, x - 1
  [] i ≥ y - 1 → i, x := 0, x - 1
  fi
  [[ $\sigma^1_1$ ,  $\sigma^2_1$ ,  $\sigma^3_1$ ,  $\sigma^4_1$ ,  $\sigma^5_1$ ]]
od
res := i
```

Los valores iniciales de las variables `x`, `y` se le deben solicitar al usuario usando la función:

```
int pedirEntero(void)
```

y se debe mostrar por pantalla el valor final de `res` usando la función

```
void imprimeEntero(int x)
```

ambas funciones definidas en el ejercicio 6 del Proyecto 3

Ejercicio 4*

Escribir un programa en C que sea equivalente al del Ejercicio 3 (en el estado final, la variable `res` de ambos programas deben tener el mismo valor) pero que no utilice ningún ciclo (ni recursión). El programa debe permitir al usuario ingresar los valores iniciales y mostrar el resultado por la pantalla.

TEMA D

Ejercicio 1

Traducí las expresiones en formalismo básico de la tabla a lenguaje C y construí un programa que solicite al usuario el ingreso de los valores para las variables x , y , z e imprima el resultado dichas expresiones:

Expresión	$\llbracket x \rightarrow 3, y \rightarrow 6, z \rightarrow \text{True} \rrbracket$	$\llbracket x \rightarrow 0, y \rightarrow 3, z \rightarrow \text{False} \rrbracket$
$x = D_1 + D_2 \vee y = D_1 + D_2 \vee z$		
$(D_1 + D_2 + x) \bmod 3$		
$x * D_1 + y * D_2$		
$x \neq 0 \wedge 0 \leq (D_1 - D_2) / x \wedge \neg z$		

Antes de traducir, reemplazá las constantes D_1 y D_2 por tu última y penúltima cifra de tu número de DNI, respectivamente. Entonces si tu número de DNI es 40.123.789 las expresiones quedarían:

$x = 9 + 8 \vee y = 9 + 8 \vee z$
$(9 + 8 + x) \bmod 3$
$x * 9 + y * 8$
$x \neq 0 \wedge 0 \leq (9 - 8) / x \wedge \neg z$

Compilá el programa y completá los resultados de la tabla para los estados especificados.



Pensá si podés usar alguna de las funciones que usaste o alguna parte del código del Proyecto 3.

Ejercicio 2

Siendo D_1 y D_2 los dos últimos números de tu DNI ($XX.XXX.XD_1D_2$), programá en lenguaje C una traducción del siguiente programa:

```
var x, y, dni1, dni2 : Int;
[[σ0 : (x ↦ 7, y ↦ 4, dni1 ↦ D1, dni2 ↦ D2)]]
if dni1 mod 2 ≠ 0 ∧ dni2 mod 2 = 0 → x, y := x + dni1, y + x
[] dni1 mod 2 = 0 ∨ dni2 mod 2 ≠ 0 → x := y + x
fi
[[σ1 : (x ↦ ?, y ↦ ?, dni1 ↦ D1, dni2 ↦ D2)]]
```

Asegurarse que los valores iniciales de x , y , $dn1$, $dn2$ se correspondan con el estado σ_0 . El programa debe mostrar por pantalla los valores finales de x e y .

Ejercicio 3

Traducí a lenguaje C y compilá el siguiente programa. Luego, completá los estados σ^1_1 , σ^2_1 , σ^3_1 , σ^4_1 , σ^5_1 ; donde cada estado a completar es el resultado de realizar 1, 2, 3, 4 y 5 iteraciones del ciclo. Recordá, una iteración es la ejecución completa del cuerpo del ciclo.

```
var x, y, z, i : Int;
var res : Bool;
[[ $\sigma_0$  : (x ↦ 2, y ↦ 5, z ↦ 8, i ↦ 0, res ↦ False)]]
i := y
do (i ≠ x ∨ y ≠ z) ∧ i ≠ 0 ∧ z ≠ 0 →
  if i ≠ x → i := i - 1
  [] i = x → skip
  fi
  if y ≠ z → z := z - 1
  [] y = z → skip
  fi
  [[ $\sigma^1_1$ ,  $\sigma^2_1$ ,  $\sigma^3_1$ ,  $\sigma^4_1$ ,  $\sigma^5_1$ ]]
od
res := i * z ≠ 0
```

Los valores iniciales de las variables x, y, z se le deben solicitar al usuario usando la función:

```
int pedirEntero(void)
```

y se debe mostrar por pantalla el valor final de res usando la función

```
void imprimeEntero(int x)
```

ambas funciones definidas en el ejercicio 6 del Proyecto 3

Ejercicio 4*

Escribir un programa en C que sea equivalente al del Ejercicio 3 (en el estado final, la variable res de ambos programas deben tener el mismo valor) pero que no utilice ningún ciclo (ni recursión). El programa debe permitir al usuario ingresar los valores iniciales y mostrar el resultado por la pantalla.