

## PROGRAMACIÓN

### Propuesta de resolución Trabajo Práctico N° 1 - Algoritmos

---

#### ALGORITMO TUNEANDO VERBOS

ENTRADA: CC: char, N: entero

SALIDA: CC: char

V. AUXILIARES: ante, ultima: char, i: entero (+)

1. LEER (N)
2. **Convertir verbos**
3. PARAR

#### 2. Convertir verbos

HACER N veces (i=1,...,N)

LEER (CC)

ante <- CC

LEER (CC)

ultima <- CC

LEER (CC)

MIENTRAS (CC <> MF)

ESCRIBIR (ante)

ante <- ultima

ultima <- CC

LEER (CC)

FIN MIENTRAS

SI ((ante="s"  $\wedge$  ultima="s")  $\vee$  (ante="s"  $\wedge$  ultima="h")  $\vee$   
(ante="c"  $\wedge$  ultima="h")  $\vee$  (ultima="x")  $\vee$  (ultima="o")) ENTONCES

ESCRIBIR(ante,ultima,"es")

SINO

SI (esVocal(ante)=0  $\wedge$  ultima="y") ENTONCES

ESCRIBIR(ante,"ies"),

SINO

ESCRIBIR(ante,ultima,"s")

FIN HACER

#### FUNCION esVocal

ARGUMENTO: car:char

SALIDA: entero

SI (car="a"  $\vee$  car="e"  $\vee$  car="i"  $\vee$  car="o"  $\vee$  car="u") ENTONCES

esVocal <- 1

SINO

esVocal <- 0

**NOTA:** Se da por supuesto que siempre entra como mínimo una palabra de dos caracteres.

## PROGRAMACIÓN

### Propuesta de resolución Trabajo Práctico N° 1 - Algoritmos

---

#### ALGORITMO TRENCITO DE NÚMEROS

ENTRADA: número: real (Secuencia de números reales)

SALIDA: suma: real, orden: entero (+)

V. AUXILIARES: total: real

##### 1. Inicializar

2. LEER(numero)

3. total <- número

##### 4. Sumar números

5. PARAR

##### 1. Inicializar

suma <- 0

orden <- 1

##### 4. Sumar números

LEER (numero)

MIENTRAS (suma+número <= total)

    suma <- suma + numero

    orden <- orden +1

    LEER (numero)

FIN

ESCRIBIR (suma, orden)

## PROGRAMACIÓN

### Propuesta de resolución Trabajo Práctico N° 1 - Algoritmos

---

#### ALGORITMO UBICANDO A MI AMIGO/A

ENTRADA: x, y: enteros

SALIDA: mensaje: char (cadena de caracteres), distancia: real (+)

V. AUXILIARES:

##### 1. Inicializar

##### 2. Procesar punto

3. ESCRIBIR(mensaje, distancia)

4. PARAR

##### 1. Inicializar

distancia <- 0

##### 2. Procesar punto

LEER(x,y)

SI(x>0 ^ y>0)ENTONCES

    mensaje <- "Está en el 1° cuadrante"

    distancia <- **calcularDistancia**(x,y)

SINO

    SI(x<0 ^ y>0)ENTONCES

        mensaje <- "Está en el 2° cuadrante"

    SINO

        SI(x<0 ^ y<0)ENTONCES

            mensaje <- "Está en el 3° cuadrante"

        SINO

            SI(x>0 ^ y<0)ENTONCES

                mensaje <- "Está en el 4° cuadrante"

        SINO

            SI(x=0 ^ y=0)ENTONCES

                mensaje <- "Está en 24 y 25"

        SINO

            SI(x>0 y=0)ENTONCES

                mensaje <- "Está sobre 24"

        SINO

            mensaje <- "Está sobre 25"

#### **FUNCION calcularDistancia**

ARGUMENTOS: x,y: enteros

SALIDA: real

calcularDistancia <- (x\*x + y\*y)^0.5

## PROGRAMACIÓN

### Propuesta de resolución Trabajo Práctico N° 1 - Algoritmos

---

#### ALGORITMO AYUDANDO AL PROFE HUGO

ENTRADA: p1, p2, efinal, tfinal: reales, alumno:char

SALIDA: nota: real, alumno\_menor: char

V. AUXILIARES: i,menor: enteros(+)

##### 1. Inicializar

##### 2. Calcular notas

3. ESCRIBIR(alumno\_menor)

4. PARAR

##### 1. Inicializar

menor <- 10 //máximo de nota posible

##### 2. Calcular notas

HACER 38 veces (i=1,...,38)

LEER(alumno, p1,p2,efinal, tfinal)

nota <- **calcularNota**(p1, p2, efinal, tfinal)

ESCRIBIR(nota)

SI(nota < menor)ENTONCES

menor <- nota

alumno\_menor <- alumno

FIN SI

FIN HACER

#### **FUNCION calcularNota**

ARGUMENTOS: p1, p2, efinal, tfinal: reales,

SALIDA: real

V.Auxiliar: promedio:real

promedio <- (p1+p2)/2

calcularNota <- 0,55\*promedio + 0,30\*efinal + 0,15\*tfinal