

Ejercicio 1: Sucesor de un DÍA de la semana

El tipo **DÍA** define por enumeración un día de la semana. En el ejercicio que determina el día del 1 de mayo de un año dado, también se ha especificado una función sucesor para un día de la semana. Falta dar una definición de esta función.

Dar una definición completa de la función sucesor para un día de la semana.

Todavía no disponemos de las herramientas que nos permitirán dar una definición «elegante» de esta función. Lo haremos más adelante.

*Algoritmo 3: Definición de **sucesor** en aritmética modulo 7*

```
sucesor(día : DIA) : DIA
  # El sucesor de `día' en la semana.

Precondición
   $0 \leq \text{día} \leq 6$ 

realización
  Resultado  $\leftarrow$  (día+1) modulo 7

postcondición
  Resultado = resto(día + 1, 7)

fin sucesor
```

La precondición es redundante porque **día** es una instancia del tipo **DIA** definido más alto en la versión 2. Además, ni la relación de orden ni la suma están definidas en el conjunto de los días. Por lo tanto, esta cláusula de precondición no es «legal».

Para seguir siendo coherente con la aritmética definida en los números enteros, podríamos definir el tipo de la siguiente manera:

```
tipo DIA estructura
  # versión 3

  j : ENTERO

  # (lunes, martes, miércoles, jueves, viernes, sábado, domingo)
  # ( 1,      2,      3,      4,      5,      6,      0 )

invariante
   $0 \leq j \leq 6$ 

fin DIA
```

El algoritmo de sucesor se convierte en:

*Algoritmo 4: Definición de **sucesor** - versión 3*

```

sucesor(día : DIA) : DIA
    # El sucesor de `día` en la semana.

realización
    Resultado ← (día.d + 1) modulo 7

postcondición
    Resultado = resto(día.d + 1, 7)

fin sucesor

```

Esta vez, las operaciones están completamente definidas porque el operando es `día.d` que es un **ENTERO**. La precondition y la primera cláusula de la postcondición son el invariante del tipo **DIA** que está implícito.

Ejercicio 2: Números, suma y producto

Dados dos números cualesquiera.

Clasificarlos respecto a su suma y a su producto.

Así, por ejemplo, dados $a = -15$ y $b = 6$, se obtiene $a \times b < a < a + b < b$ cuyos valores son, en orden: -90, -15, -9 y 6.

Algoritmo 2: Clasificar cuatro datos comparables

```

Algoritmo clasificar4
    # Clasifica `a`, `b`, `c` y `d` en orden creciente.

Entrada
    a, b, c, d : T → COMPARABLE

precondición
    VERDADERO

realización
    clasificar3(a, b, c)
    # a ≤ b ≤ c ; situar `d`

    si
        d < a
    entonces
        # a > d ; b ≤ c
        intercambiar(a, d)
        # a ≤ d ; b ≤ c
    fin si
    # a ≤ d ; b ≤ c

    clasificar3(b, c, d)
    # a ≤ b ≤ c ≤ d

postcondición
    a ≤ b ≤ c ≤ d

fin clasificar4

```

Ejercicio 3: Descuento

Un comerciante hace un descuento del 5 % en todas las compras con un importe comprendido entre 100 y 500 €, y del 8 % en los importes superiores.

Escribir el algoritmo de cálculo del importe del descuento en una compra dada.

Cálculo del importe del descuento

```
descuento(precio : REAL) : REAL
    # Importe del descuento acordado sobre `precio`.

Precondición
    precio ≥ 0

realización
    si
        precio < 100,00
    entonces
        # precio < 100,00 => no hay descuento
        Resultado ← 0,00
    si no si
        precio < 500
    entonces
        # 100 ≤ precio < 500 => descuento del 5 %
        Resultado ← precio x 0,05
    si no
        # precio ≥ 500 => descuento del 8 %
        Resultado ← precio x 0,08
    fin si

postcondición
    precio < 100,00    => Resultado = 0,00
    100 ≤ precio < 500 => Resultado = precio x 0,05
    500 ≤ precio      => Resultado = precio x 0,08

fin descuento
```

Ejercicio 4: Otra vez una media

En un sistema de calificaciones de 0 a 20, donde 20 es la nota más alta y 0 la más baja.

Un profesor quiere escribir un programa que calcula la media de las cuatro notas obtenidas por sus alumnos en los deberes del mes. Además, el programa deberá calcular una evaluación automática según la media del alumno. Dará «Alumno con talento» si la

media es superior a 15, «Con capacidad» si está comprendida entre 12 y 15, y, por último, «Debe reorientarse» si es inferior a 12.

Escribir un algoritmo que toma como entrada las cuatro notas de un alumno y que calcula la media y la evaluación correspondiente.

El problema anterior se puede resolver definiendo una estructura de datos que, para un alumno, agrupa su media y la evaluación. Un elemento de este tipo calcula el algoritmo solicitado.

Cálculo media de calificaciones

Algoritmo calificación

Entrada

N1, N2, N3, N4 : REAL # Calificaciones obtenidas
Med : REAL # Media obtenida

Resultado: REAL

Precondición

$Med \geq 0 \leq 20$

realización

Med $\leftarrow (N1+N2+N3+N4)/4$

si

$Med > 15$

entonces

calificación >15 => Alumno con talento

Resultado \leftarrow Alumno con talento

si no si

$Med \geq 12$ y $Med \leq 15$

entonces

$12 \geq Med \leq 15$ => Alumno con capacidad

Resultado \leftarrow Alumno con capacidad

si no

$Med < 12$ => Debe reorientarse

Resultado \leftarrow Debe reorientarse

fin si

postcondición

$calificación >15$ => **Resultado** \leftarrow Alumno con talento

$12 \geq Med \leq 15$ => **Resultado** \leftarrow Alumno con capacidad

$Med < 12$ => **Resultado** \leftarrow Debe reorientarse

fin calificación

Ejercicio 5: Con ADIF puedes

ADIF hace descuento a las familias que van al Parque Warner Madrid en función de la cantidad de niños que hay en la familia. Este descuento es del 10 % para 2 niños, 15 % para 3 niños y 18 % para 4 niños. A partir de 5 niños, el descuento es del 18 %, pero aumenta un 1 % por cada niño por encima de 4.

Establecer el algoritmo que calcula el importe del descuento al que tendrá derecho una familia dada.

Cálculo del importe del descuento

Algoritmo descuento

Entrada

```
niños : ENTERO # Número de niños
precio : REAL # Precio de la entrada sin descuento
desc : REAL # Descuento aplicable
Resultado: REAL # Importe del descuento acordado sobre `niños'.
```

Precondición

```
niños ≥ 0
```

realización

```
si
    niños < 2
entonces
    # niños < 2 => no hay descuento
    Resultado ← 0
si no si
    personas ≥ 2 y personas < 3
entonces
    # 2 ≤ niños < 3 => descuento del 10 %
    Resultado ← niños x 0,1
si no
    # 3 ≤ niños < 4 => descuento del 15 %
    Resultado ← niños x 0,15

    si no
        # 4 ≤ niños < 5 => descuento del 18 %
        Resultado ← niños x 0,18
```

De lo contrario

```
# niños > 4 => descuento del 18 % + 1% adicional por niño
Resultado ← niños * [(precio) * (18 + niños - 4)] / 100
fin si
```

```

postcondición
    niños < 2      => Resultado = 0
    2 ≤ niños < 3  => Resultado = precio x 0,1
    3 ≤ niños < 4  => Resultado = precio x 0,15
    4 ≤ niños < 5  => Resultado = precio x 0,18
    niños > 4 => Resultado del 18 % + 1% adicional por niño
fin descuento

```

Ejercicio 6: Descuento en los microprocesadores

La empresa UNTEL hace descuentos por la compra al por mayor de sus microprocesadores. Estos descuentos dependen de la cantidad de componentes pedidos y del cliente que los pide.

El descuento concedido es de un 10 % si la cantidad de componentes pedidos se encuentra entre 10 000 y 20 000, un 15 % si la cantidad se encuentra entre 20 001 y 40 000 y un 20 % para más de 40 000 componentes.

Además, si el cliente es COMMAQ, el descuento se reduce un 2 %. Por último, BEL disfruta de un descuento mejorado en un 1 %.

Establecer el algoritmo del cálculo del porcentaje de descuento concedido a un cliente dado para un pedido dado.

Cálculo del porcentaje del descuento

```

Algoritmo porcentaje_descuento

Entrada
    ump: ENTERO # Unidades compradas
    c: Entero # Tipo de cliente (Bel +1% y Commaq -2%)
                # Número de cliente: Bel 1, Commaq 2, resto 3)
    d: Entero # Porcentaje del descuento acordado sobre
                unidades y cliente

Precondición
    upm > 0 # Mayor que 0, ya que si es 0 no hay pedido

realización
    si
        10.000 ≤ upm ≤ 20.000

    entonces
        d=10

    si
        c=1

    entonces
        d=d-2 # Porcentaje de descuento aplicable para COMMAQ
    Resultado ← d=d-2

```

```

si no si
     $20.001 \leq upm \leq 40.000$ 
entonces
    d=15

si no si
    c=2
entonces
    d=d+1 # Porcentaje de descuento aplicable para BEL
    Resultado  $\leftarrow$  d=d+1
si no si
     $upm > 40.000$ 
entonces
    d=20
si no si
    c=3
entonces
    d=d # Porcentaje de descuento aplicable resto de clientes
    Resultado  $\leftarrow$  d=d

fin si

postcondición
     $upm < 0 \Rightarrow \text{Resultado} = 0,00$ 

fin porcentaje_descuento

```

Ejercicio 7: Viaje escolar

Un profesor planea organizar un viaje escolar. El coste del viaje depende de la cantidad de alumnos participantes.

El coste del trayecto es de 67,30 € por alumno hasta 25 alumnos y de 61,00 € si hay más de 25 alumnos. El coste de la comida es de 3,50 € por día y por alumno. Por último, el alojamiento es de 4,75 € por día y por alumno si la cantidad de alumnos es inferior a 30; 4,00 € para una cantidad de alumnos de entre 31 y 35, y 3,50 € si son más de 35.

Establecer el algoritmo de cálculo del precio de coste por alumno y del coste global del viaje en función de la cantidad de alumnos.

```

Algoritmo descuento

Entrada

    n : ENTERO # Número de alumnos
    d : ENTERO # Número de días
    c1 : REAL # Coste del trayecto
    c2 : REAL # Coste comida por día
    c3 : REAL # Coste alojamiento por día

Salida

    Ca : REAL # Coste por alumno
    Ct : REAL # Coste global

Resultado: REAL # Coste por alumno y coste global

```

```

realización
  si
     $0 < n \leq 25$ 
  entonces
     $c1 = 67,3$  # Coste del trayecto
     $c2 = 3,5$  # Coste comida por día
     $C3 = 4,75$  # Coste alojamiento por día

    Resultado  $\leftarrow ca = c1 + c2 * d + c3 * d$  # Coste viaje por alumno
    Resultado  $\leftarrow ct = n * ca$  # Coste total
  si no
     $0 > 25$ 
  entonces
     $c1 = 61$  # Coste del trayecto
     $c2 = 3,5$  # Coste comida por día

  si
     $25 < n \leq 30$ 
     $C3 = 4,75$  # Coste alojamiento por día
  si no
     $31 \leq n \leq 35$ 
     $C3 = 4$  # Coste alojamiento por día
  si no
     $C3 = 3,5$  # Coste alojamiento por día
    Resultado  $\leftarrow ca = c1 + c2 * d + c3 * d$  # Coste viaje por alumno
    Resultado  $\leftarrow ct = n * ca$  # Coste total

  fin si

fin descuento

```

Ejercicio 8: Prima anual

A final de año, la empresa LA CAMPANA paga una prima anual a sus empleados camioneros.

En principio, el conductor recibirá la prima anual completa si no ha tenido accidentes con una responsabilidad superior o igual al 20 % durante el año que termina. Si la responsabilidad es superior al 20 %, la empresa considera al conductor responsable del accidente. Si el conductor ha sido responsable de un accidente, solo recibe la mitad de la prima. Con dos accidentes, solo recibe un tercio. Con tres accidentes, la prima se reduce a un cuarto. Si supera los tres accidentes, la prima se anula.

- Esta prima es la suma de una prima de distancia y de una prima de antigüedad.
- La prima de distancia aumenta un céntimo por kilómetro recorrido durante el año, con un máximo de 400 €.

La prima de antigüedad solo se paga una vez transcurridos cuatro años de antigüedad y es de 200 €. Luego aumenta 20,00 € por año adicional.

Escribir el algoritmo de cálculo de la prima anual que se concederá a cada conductor.

Algoritmo prima_anual

```
Algoritmo prima_anual
    # Importe de la prima anual en función del número
    # de accidentes, de la distancia recorrida y de la antigüedad
    # del conductor.

entrada
    accidentes : ENTERO # Número de accidentes
    distancia : ENTERO # Distancia recorrida
    antigüedad : ENTERO # Antigüedad

Resultado: REAL

variable
    prima_antigüedad : REAL
    prima_distancia : REAL

realización
    si
        accidentes > 3
    entonces
        Resultado ← 0,00
    si no
        # Cálculo de la prima de antigüedad
        si
            antigüedad < 4
        entonces
            prima_antigüedad ← 0,00
        si no
            prima_antigüedad ← 200,00 +
                                REAL(antigüedad - 4) x 20,00
        fin si

        # cálculo de la prima de rendimiento
        prima_distancia ← inf(REAL(distancia) x 0,01, REAL(400))

        # Cálculo de la prima anual
        Resultado ← (prima_antigüedad + prima_distancia) /
                    REAL(accidentes + 1)
    fin si

postcondición
    ...
fin prima_anual
```