

Introducción a la Programación - Práctica 8

Registros y Variantes

# **CONSEJOS:**

* Leer el enunciado en su totalidad y pensar en la forma de resolverlo ANTES de empezar a escribir código
* Si un ejercicio no sale, se puede dejar para después y continuar con los ejercicios que siguen
* Los ejercicios están pensados para ser hechos después de haber mirado la teórica correspondiente
* Algunos de los ejercicios están tomados de las guías prácticas utilizadas en la materia de Introducción a la Programación de la Universidad Nacional de Quilmes por Pablo Ernesto “Fidel” Martínez López y su equipo. También Federico Aloi y Miguel Miloro, a su vez basada en las guías Ejercicios de Introducción a la Programación del CIU General Belgrano, elaboradas por Carlos Lombardi y Alfredo Sanzo, y Fundamentos de la Programación del Proyecto Mumuki.. Agradecemos a todos los que nos ayudaron con su inspiración.

# **VARIANTES:**

1. Declarar un tipo variante llamado **DíaDeLaSemana**, que sirva para identificar los días de la semana. Luego implementar las siguientes funciones (sin olvidar sus contratos):
   1. **díaSiguiente\_(unDía)**, que dado un día de la semana, devuelve el día siguiente.
   2. **díaPrevio\_(unDía)**, que dado un día de la semana, devuelve el día previo.
   3. **esDíaDeFinDeSemana\_(unDía)**, que indica si el día dado es uno del fin de semana.
2. Declare el tipo variante **PaloDeCartaEspañola** y el tipo registro **Carta** y escriba las siguientes funciones:
   1. anchoDeEspadas() que describe la carta 1 de Espadas.
   2. anchoDeBastos() que describe la carta 1 de Bastos
   3. laCarta\_de\_(número, palo) que describe la carta “número” del palo “palo”.
   4. esFigura(unaCarta) que dada una carta, describa Verdadero si la misma es una figura (las figuras son los 10s, los 11s y los 12s) y Falso en otro caso.
   5. esUnAncho(unaCarta) que describa Verdadero si la carta dada es un 1, y Falso en otro caso.
   6. tieneUnNúmeroMásGrande(unaCarta, otraCarta), que describa Verdadero cuando la primer carta tiene un número más grande que la segunda y Falso en otro caso.
   7. esDeOro(unaCarta) que describa Verdadero si la carta dada es de Oros, y Falso en otro caso.
   8. sonDelMismoPalo(unaCarta, otraCarta) que describa Verdadero si dadas dos cartas, estas tienen el mismo palo.
   9. valorParaEnvido(unaCarta) que describa el número que aporta la carta dada en el canto del envido. El número se corresponde al número de la carta, si la misma no es figura, y cero, si fuera figura.
   10. mayorValorEntre(unaCarta, otraCarta) que describe cual es el valor más grande para el envido entre las dos cartas. Por ejemplo, si las cartas son un 7 y un 12, describe 7.
   11. sumaParaElEnvido(unaCarta, otraCarta) que describa el número que tienen dos cartas para el envido. El envido se calcula como la suma los valores del envido más 20, si las cartas son del mismo palo, o como el mayor valor entre ellas, cuando son de distinto palo.
   12. sonMejores(miPrimerCarta, miSegundaCarta, suPrimerCarta, suSegundaCarta) que describe Verdadero cuando el envido de “mis” cartas es mejor al envido “sus” cartas. El envido es mejor si efectivamente el número es más grande o igual.
3. Dado el siguiente tipo de registro:

**type Celda is record {**

***/\* PROPÓSITO: modelar una celda del tablero***

***INV.REP.: los números son todos >=0***

***\*/***

**field cantidadDeAzules // un número**

**field cantidadDeNegras // un número**

**field cantidadDeRojas // un número**

**field cantidadDeVerdes // un número**

**}**

Se pide que realice las siguientes funciones y procedimientos:

* 1. celdaActual() que lee la información de la celda en donde está el cabezal y retorna un dato de tipo Celda.
  2. EscribirEnCelda(unaCelda) que dado el dato de una calda, escribe la información de este en el tablero.
  3. tienenMismaCantidadDeRojas(unaCelda, otraCelda) que dados dos datos del tipo celda, describe Verdadero si efectivamente ambos tienen la misma cantidad de bolitas rojas.

1. Declarar un tipo de registros llamado **Persona**, que contenga el número de DNI y el domicilio representados mediante Strings y un booleano indicando si la persona es de donante de órganos. Implementar las siguientes funciones:
   1. **sonConvivientes\_Y\_**, que dadas dos personas indique si comparten domicilio.
   2. **personaNacidaDe\_El\_**, que, dada una persona madre, cree una nueva persona que conviva con la madre, no tenga DNI asignado y en principio se indica que no es donante de órganos.  
      Para registrar el DNI sin asignar, escribir la función **sinAsignar**, que denote el string vacío y utilizarla adecuadamente.
   3. **persona\_RegistradaCon\_**, que dada una persona con DNI sin asignar y un DNI de registro, describe a la persona pero con el DNI asignado al dado.
   4. **persona\_ConDomicilioNuevoEn\_**, que recibe una persona y un nuevo domicilio y describe a la persona con el domicilio cambiado.
   5. **persona\_ConSituaciónComoDonanteCambiada**, que recibe una persona y retorna la persona con su situación como donante cambiada.
2. Declarar un tipo de registros llamado **Empleado** que contenga un nombre, el puesto dentro de la empresa, representado por un tipo variante llamado **Puesto** dado a continuación, y un sueldo modelado como un número en centavos.

**type Puesto is variant {  
 */\* PROPÓSITO: modelar los diferentes puestos que hay dentro   
 de una empresa de software  
 \*/* case GestorDeProyecto {}  
 case LíderDeProyecto {}   
 case Desarrollador {}  
 case Operador {}  
}**

Implementar las siguientes funciones:

* 1. **empleado\_ConSueldoActualizadoA\_**, que dado un empleado y un nuevo sueldo, describa al empleado con el sueldo actualizado.
  2. **empleado\_ConNuevoPuesto\_**, que dado un empleado y un nuevo puesto, describa al empleado con el puesto actualizado.
  3. **categoríaDelPuesto\_**, que dado un valor de tipo puesto devuelve su categoría según la siguiente tabla:

|  |  |
| --- | --- |
| **GestorDeProyecto** | 4 |
| **LíderDeProyecto** | 3 |
| **Desarrollador** | 2 |
| **Operador** | 1 |

* 1. **empleadoConMayorCargoEntre\_Y\_**, que dados dos empleados describa el empleado de mayor categoría entre ellos.
  2. **tienenIgualSueldo\_Y\_**, que dados dos empleados indique si ambas cobran lo mismo.
  3. **empleado\_ConAumentoEn\_PorBonoDeFinalizaciónDeProyecto**, que dado un empleado y un porcentaje de aumento describa al empleado con el sueldo actualizado en ese porcentaje. El porcentaje de aumento es un número del 0 al 100. El monto en el que se incrementa depende del porcentaje (i.e. si el porcentaje es 20%, cuando el sueldo es 100, el nuevo sueldo es 120, cuando el sueldo es 200 el nuevo sueldo es 240, y cuando el sueldo es 150, el nuevo sueldo es 180).