

**Práctica Nro. 7
Tipos de Datos**

Objetivo: Interpretar y conocer las estructuras de datos que definen e implementan los distintos lenguajes.

Ejercicio 1: ¿Qué características encierra el concepto de “tipo de datos”?

Ejercicio 2: En todos los lenguajes ¿Una variable se liga estáticamente con su tipo? En caso de respuesta negativa, de al menos un ejemplo.

Ejercicio 3: Teniendo en cuenta el siguiente ejemplo, Enumere y explique las diferencias que existen entre el manejo de los tipos enumerativos entre Pascal y Ada.

```
type día=(lunes, martes, miércoles, jueves, viernes, sábado, domingo)
```

```
....
```

```
var nuevo-día, día-hoy: día;
```

```
can-horas-trabajadas: integer;
```

```
begin
```

```
.....
```

```
día-hoy:=nuevo-día;
```

```
case nuevo-día of
```

```
lunes..martes: can-horas-trabajadas:= can-horas-trabajadas + 4;
```

```
miércoles: can-horas-trabajadas:= can-horas-trabajadas + 2;
```

```
jueves-viernes: can-horas-trabajadas:= can-horas-trabajadas + 5;
```

```
end;
```

```
...
```

¿Qué diferencias hay entre subtipo y subrango para cada lenguaje?

Ejercicio 4: Manejo de punteros:

1. ¿Permite C tomar el l-valor de las variables? Ejemplifique. ¿Que sucede en otros lenguajes con esta operación como ser en PASCAL y en JAVA?
2. ¿Qué problemas podría acarrear utilizar este tipo de operación?
3. ¿Que otros problemas de seguridad existen en el manejo de punteros? Ejemplifique.

Ejercicio 5: ¿Los registros variantes en ADA son seguros? ¿Sucede lo mismo en C?. De un ejemplo en C de una unión discriminada.

Ejercicio 6:

a) En Python se habla de tipos de datos mutables e inmutables. ¿Para que se utiliza este tipo de datos? De al menos un ejemplo donde se lo puede utilizar. ¿Cómo?

b) ¿Sucede lo mismo para Ruby? Explique la forma en que un objeto mutable puede convertirse en inmutable.

Ejercicio 7: En la siguiente definición de un puntero en ADA:

<pre>with Ada.Text_IO; use Ada.Text_IO; procedure Hello is type Tdato is record StrDato: String(1..6); Valor: Integer; end record; type PointerTDato is access TDato; type PointerInteger is access integer; var1, var2: PointerTDato; pi: PointerInteger;</pre>	<pre>begin var1 := new Tdato; var2 := new Tdato'(Valor => 2, StrDato => "Hola22"); pi:= new integer'(10); var1.all.StrDato := "Hola11"; Put_Line(var1.StrDato); Put_Line(var2.StrDato); Put_Line(var1.all.StrDato); Put_Line(Integer'Image(var2.Valor)); Put_Line(Integer'Image(pi)); end Hello;</pre>
--	--

A que estamos referenciando con var1 y var1.all. Defina que funcionalidad tiene la operación .all. Pruebe este código en [compiler-online](#) y defina el resultado. ¿Qué sucede en particular con los registros? ¿Este código compila correctamente?

Ejercicio 8: De al menos dos definiciones de fuertemente tipado. Las definiciones deben ser extractadas de los libros dados en la bibliografía. Compárelas e indique cuál le parece mejor.

Ejercicio 9: Diferencie el la respuesta del ejercicio anterior con el concepto de estáticamente tipado.

Ejercicio 10: De un ejemplo de un lenguaje que sea fuertemente tipado y un lenguaje que no. Explicando en cada caso porque lo es y porque no lo es.

Ejercicio 11: ¿Que características debe cumplir una unidad para que sea un TAD?

Ejercicio 12: ¿Cómo logramos definir un TAD en ADA? ¿Tiene relación con éste tema las cláusulas PRIVATE y LIMITED PRIVATE?