

# ALGORITMOS Y PROGRAMACIÓN II

---

Clase 5  
Programación  
orientada a objetos

# Temas de la clase

- Clase de objetos (Continuación)
- Uso de self



# Self

¿Cómo haemos para determinar si dos contadores manuales contaron la misma cantidad?

```
program Contadores;  
uses UContadorManual;  
var c1, c2: ContadorManual;  
begin  
  c1:= ContadorManual.create();  
  c2:= ContadorManual.create();  
  
  . . . { Uso de los contadores c1 y c2 }  
  
  if c1.getCantidad() = c2.getCantidad() then  
    writeln('Los contadores contaron lo mismo.');
```

{ ¿Cómo podríamos hacer para comparar los contadores desde el punto de vista de los objetos? }

```
end.
```



# Self

Desde el punto de vista de objetos, un contador debería saber si contó lo mismo que otro.

```
program Contadores;  
uses UContadorManual;  
var c1, c2: ContadorManual;  
begin  
  c1:= ContadorManual.create();  
  c2:= ContadorManual.create();  
  
  . . . { Uso de los contadores c1 y c2 }  
  
  if c1.sosIgual(c2) then  
    writeln('Los contadores contaron lo mismo.');
```

**end.**



# Self

**type**

```
ContadorManual = class
```

```
  private
```

```
    cantidad: integer;
```

```
  public
```

```
    ...
```

```
    function sosIgual(c: ContadorManual): boolean;
```

```
end;
```

```
function ContadorManual.sosIgual(c: ContadorManual): boolean;
```

```
begin
```

```
if self.getCantidad() = c.getCantidad() then
```

```
  sosIgual := true
```

```
else
```

```
  sosIgual := false;
```

```
end;
```

```
if c1.sosIgual(c2) then
```



# Actividades en Máquina

## ACTIVIDAD 1: Modificar el archivo **UContadorManual.pas**

Implemente en la clase ContadorManual el método:

- `function soslgual(c: ContadorManual): boolean;`

Modifique la actividad 2 de la clase 3 para imprimir si ambos contadores finalizaron con la misma cantidad.



# Comparando fechas

Los objetos **Date** podrían compararse contra otro **Date** para determinar si son iguales o uno es mayor o menor que el otro.

```
program Fechas;  
uses UDateTime;  
var f1, f2, f3: Date;  
begin  
  f1:= Date.create(10,6,2020);  
  f2:= Date.create(17,2,2021);  
  
  if f1.equals(f2) then  
    writeln('Las fechas son las mismas')  
  if f1.lessThan(f2) then  
    writeln('Las fecha 1 es anterior a fecha 2');  
  if f1.greaterThan(f2) then  
    writeln('Las fecha 1 es posterior a fecha 2');  
end.
```



# Actividades en Máquina

## ACTIVIDAD 2: Modificar el archivo **UDateTime.pas**

Implemente en la clase **Date** los métodos:

- `function equals(d: Date): boolean;`  
    // Devuelve true si ambas fechas son iguales
- `function greaterThan(d: Date): boolean;`  
    // Devuelve true si la fecha 'self' es posterior a la fecha 'd'
- `function lessThan(d: Date): boolean;`  
    // Devuelve true si la fecha 'self' es anterior a la fecha 'd'

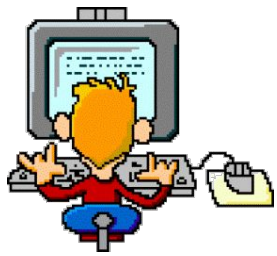




# Actividades en Máquina

## ACTIVIDAD 3: Crear el archivo **Fechas .pas**

- Implemente un módulo que cargue un vector de 15 fechas de nacimiento elegidas al azar.
- Implemente un módulo que reciba un vector de fechas y una fecha a buscar y devuelva si existe en el vector.
- Implemente un módulo que reciba un vector de fechas y devuelva la fecha de la persona más joven.
- Implemente un módulo que reciba un vector de fechas y lo devuelva ordenado de menor a mayor.
- Implemente un módulo que reciba un vector de fechas y las imprima por consola.
- Escriba un programa que invoque a todos los módulos implementados y compruebe el correcto funcionamiento del mismo.



# Actividades en Máquina

## ACTIVIDAD 4: Crear el archivo **VerduleriaPOO4.pas**

Modifique el programa de la actividad 4 de la clase 4 (VerduleriaPOO3.pas).

- Realice las modificaciones necesarias para que cada una de las cajas almacene sus tickets ordenados por día.
- Implemente un módulo que reciba las listas de tickets de las cajas y retorne una lista con los resúmenes de ventas por día. Para cada uno de los días se desea saber la cantidad de ventas, monto total facturado y el ticket de la compra de mayor monto. Nota: implemente una clase **Resumen** con los campos necesarios.
- Implemente un módulo que imprima los resúmenes de ventas por día.
- Escriba un programa que invoque a los dos módulos implementados y compruebe el correcto funcionamiento del mismo.