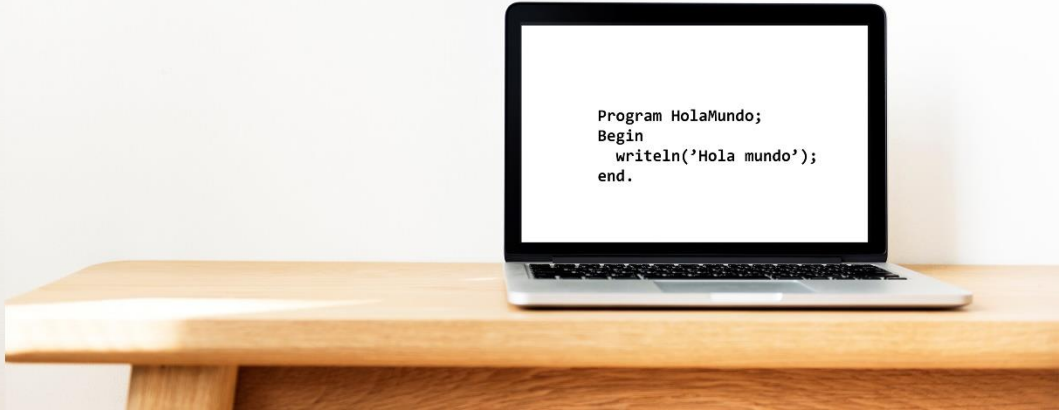


# Redictado Taller de Programación 2022

## CLASE 1

### Programación con Pascal

A silver laptop is placed on a light-colored wooden desk. The laptop screen displays a Pascal program. The background is a plain, light-colored wall.

```
Program HolaMundo;  
Begin  
  writeln('Hola mundo');  
end.
```

# Temas de la clase



1. Introducción a la Programación con Pascal.
2. Ejercitación con operaciones de Vector y Lista



# Actividades en Máquina

## ACTIVIDAD 1

a. Edite, compile y ejecute el programa **ProgramaNumAleatorio.pas**

```
program NumAleatorio;  
  
var num: integer;  
  
begin  
    Randomize;  
    num := random (100); {valores en el intervalo 0 a 99}  
    writeln ('El numero aleatorio generado es: ', num);  
    readln;  
end.
```

b. ¿Qué hace el programa?



# Actividades en Máquina

## ACTIVIDAD 2: Crear **programaLISTA.pas**

1. Crear un programa que contenga:

```
Program programaLista;  
  
Type  
  
    lista=^nodo;  
    nodo=record  
        datos: integer;  
        sig: lista;  
    end;  
  
Begin  
End.
```

2. Guardar como **programaLISTA.pas**



# Actividades en Máquina

3. Implemente el **módulo CrearListaAgregarAdelante** que genere valores aleatorios (entre 0 y 15) y los agregue *adelante* de la lista recibida por parámetro. Los valores se generan usando la función Random hasta que llega el valor 0. El módulo debe invocar al **procedure AgregarAdelante** que se muestra a continuación.

```
Procedure AgregarAdelante(var L:lista; elem: integer);  
Var nue:Lista;  
Begin  
    New(nue);  
    nue^.datos:=elem;  
    nue^.sig:=L;  
    L:=nue;  
End;
```

4. Implementar el módulo **ImprimirLista**

5. Invocar en el programa principal a ambos módulos para crear una lista e imprimirla. Compilar y ejecutar.



# Actividades en Máquina

6. Implementar el **módulo CrearListaAgregarAtrás** que genere valores aleatorios (entre 0 y 15) y los agregue *atrás* de la lista recibida por parámetro.

Los valores se generan usando la función Random hasta que llega el valor 0.

El módulo debe invocar al **procedure AgregarAtras** que se muestra a continuación.

```
procedure AgregarAtras (var pri, ult: lista; elem: integer);  
var  nue : lista;  
begin  
  new (nue);  
  nue^.datos:= elem;  
  nue^.sig := NIL;  
  if pri <> Nil then  
    ult^.sig := nue  
  else  
    pri := nue;  
  ult := nue;  
end;
```

7. Invocar en el programa principal al módulo CrearListaAgregarAtras para generar una nueva lista y luego muestre la lista generada (usar módulo ImprimirLista)



# Actividades en Máquina

8. Implementar el **módulo CrearListaOrdenada** que genere valores aleatorios (entre 0 y 15) y los inserte *en orden* en la lista recibida por parámetro.

El módulo debe invocar al **procedure InsertarEnLista** que se muestra a continuación.

```
Procedure InsertarEnLista ( var pri: lista; elem: integer);  
var ant, nue, act: lista;  
begin  
  new (nue);  
  nue^.datos := elem;  
  act := pri;  
  while (act<>NIL) and (act^.datos < elem) do begin  
    ant := act;  
    act := act^.sig ;  
  end;  
  if (act = pri) then pri := nue  
    else ant^.sig := nue;  
  nue^.sig := act ;  
end;
```

9. Invocar en el programa principal al módulo CrearListaOrdenada para generar una nueva lista y luego muéstrela en pantalla.



# Actividades en Máquina

## ACTIVIDAD 3: Crear **programaVECTOR.pas**

1. Crear un programa que contenga:

```
Program programaVector;  
  
Const  
  DimF = 10;  
Type  
  vector = Array [ 1..DimF] of integer;  
  
Begin  
  
End.
```

2. Guardar como **programaVECTOR.pas**





# Actividades en Máquina

3. Implementar el **módulo CrearVector** que cargue un vector de enteros (parámetro) con valores aleatorios (entre 0 y 15).

Los valores se generan usando la función Random hasta que llega el valor 0 o se completó el vector. El módulo debe seguir la especificación:

```
Procedure CrearVector (var v: vector; var dimL:integer);
```

4. Implementar el módulo **ImprimirVector** que reciba un vector y su dimensión lógica y muestre los elementos del vector de la siguiente forma:

```
Nros almacenados:
-----
19 | 07 | 42 | 76 | 96 | 26 | 19 | 70 |
-----
```

5. Invocar en el programa principal a los módulos **CrearVector** e **ImprimirVector** ya implementados.



# Actividades en Máquina

## ACTIVIDAD 4: Resolver

Una empresa desarrolladora de juegos para teléfonos celulares con Android dispone en papel de información de todos los dispositivos que poseen sus juegos instalados. De cada dispositivo se conoce la versión de Android instalada, el tamaño de la pantalla (en pulgadas) y la cantidad de memoria RAM que posee (medida en GB).

Realizar un programa que:

a. Lea cada dispositivo y lo guarde en una lista ordenada por versión de Android. La lectura finaliza con tamaño de la pantalla en 0.

b. Procese la lista para informar:

- La cantidad de dispositivos para cada versión de Android
- La cantidad de dispositivos con más de 3 GB de memoria y pantallas de a lo sumo a 5 pulgadas
- El tamaño promedio de las pantallas de todos los dispositivos.



# Actividades en Máquina

## ACTIVIDAD 5: Resolver

La Facultad de Informática debe seleccionar los 10 egresados con mejor promedio a los que la UNLP les entregará el premio Joaquín V. González. De cada egresado se conoce su número de alumno, apellido, y el promedio obtenido durante toda su carrera.

Implementar un programa que:

- Lea la información de todos los egresados, hasta ingresar el código 0, el cual no debe procesarse.
- Una vez ingresada la información de los egresados, se debe informar el apellido y número de alumno de los 10 egresados que recibirán el premio. La información debe imprimirse ordenada según el promedio del egresado (de mayor a menor).