

# Recursos Clase 0

Taller de Programación





# Generación de Números Aleatorios

```
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.
```



# Listas - Declaración

```
Program programaLista;
```

```
Type
```

```
    lista=^nodo;
```

```
    nodo=record
```

```
        datos: integer;
```

```
        sig: lista;
```

```
    end;
```

```
Begin
```

```
End.
```



# Listas – Agregar Adelante

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



# Listas – Agregar Atrás

```
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;
```



# Listas – Insertar Ordenado

```
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;
```



# Vectores - Declaración

```
Program programaVector;
```

```
Const
```

```
— DimF = 10;
```

```
Type
```

```
    vector = Array [ 1..DimF] of integer;
```

```
Begin
```

```
End.
```