

Programación 1 – Examen Julio 2022
Instituto de Computación - Facultad de Ingeniería
Soluciones

Solución Ejercicio 1

Parte a (10 pts)

```
function SonIguales (n1, n2 : Numero): boolean;  
var iguales : boolean;  
begin  
    iguales := false;  
    if n1.tipo = n2.tipo then  
        if n1.tipo = NumEntero then  
            iguales := n1.valorEntero = n2.valorEntero  
        else  
            iguales := n1.valorReal = n2.valorReal  
        end  
    end  
    SonIguales := iguales  
end;
```

Parte b (15 pts)

```
procedure SumarNumeros (n1, n2 : Numero; var suma : Numero);  
begin  
    if n1.tipo = NumEntero then  
        if n2.tipo = NumEntero then  
            begin  
                suma.tipo := NumEntero;  
                suma.valorEntero = n1.valorEntero + n2.valorEntero  
            end  
        else begin  
            suma.tipo := NumReal;  
            suma.valorReal = n1.valorEntero + n2.valorReal  
        end  
    else begin  
        suma.tipo = NumReal;  
        if n2.tipo = NumEntero then  
            suma.valorReal = n1.valorReal + n2.valorEntero  
        else  
            suma.valorReal = n1.valorReal + n2.valorReal  
        end  
    end  
end;
```

Parte c (20 pts)

```
function TodosIguales (act : ArregloConTope) : boolean;  
var i : integer;  
    iguales : boolean;  
begin  
    iguales := true;  
    if act.tope > 0 then  
        begin  
            i := 2;  
            while (i <= act.tope) and (SonIguales (act.arre[1], act.arre[i])) do  
                i := i + 1;  
            end  
            iguales := i > act.tope  
        end  
    end  
    TodosIguales := iguales  
end;
```

Parte d (15 pts)

```
procedure SumarTodos (act : ArregloConTope; VAR suma : Numero);
var i : integer;
begin
  suma.tipo := NumEntero;
  suma.valorEntero := 0;
  for i := 1 to act.tope do
    SumarNumeros (suma, act.arre[i], suma);
  end;
end;
```

Solución Ejercicio 2 (25 pts)

```
var p, aux, temp : Lista;
begin
  p := list;
  while (p <> nil) and (valor <> p^.dato) do
    p := p^.sig;
    if p <> nil then
      begin
        aux := p^.sig;
        p^.sig := nil;
        while aux <> nil do
          begin
            temp := aux;
            aux := aux^.sig;
            dispose(temp)
          end
        end
      end
    end;
end;
```

Solución Ejercicio 3 (15 pts)

Para x = 0, la salida es: 2, 0, 6, 2, 6
Para x = 1, la salida es: 2, 1, 8, 2, 8
Para x = 2, la salida es: 2, 2, 10, 2, 10
Para x = 3, la salida es: 2, 3, 12, 2, 12
Para x = 4, la salida es: 2, 4, 14, 2, 14
Para x = 5, la salida es: 2, 5, 16, 2, 16
Para x = 6, la salida es: 2, 6, 18, 2, 18
Para x = 7, la salida es: 2, 7, 20, 2, 20
Para x = 8, la salida es: 2, 8, 22, 2, 22
Para x = 9, la salida es: 2, 9, 24, 2, 24