

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

Ejercicio 1

Parte a

```
function docIguales (doc1, doc2: Documento): boolean;  
var i : integer;  
begin  
    i := 1;  
    while (i <= MaxDigito) and (doc1[i] = doc2[i]) do  
        i := i + 1;  
    docIguales := i > MaxDigito  
end;
```

Parte b

```
function fechasIguales (fec1, fec2: Fecha): boolean;  
var iguales : boolean;  
begin  
    iguales := false;  
    if fec1.anio = fec2.anio then  
        if fec1.mes = fec2.mes then  
            iguales := (fec1.dia = fec2.dia);  
    fechasIguales := iguales  
end;
```

Parte c

```
function estadoUsuario (doc: Documento; age: Agenda): Estado;  
var i : integer;  
begin  
    i := 1;  
    while (i <= age.tope) and not (docIguales(age.usuarios[i].documento, doc)) do  
        i := i + 1;  
    if (i <= age.tope) then  
        estadoUsuario := age.usuarios[i].est  
    else  
        estadoUsuario := Ausente  
end;
```

Parte d

```
procedure usuariosFecha (fec: Fecha; age: Agenda; var atendidos: Agenda);  
var i : integer;  
begin  
    atendidos.tope := 0;  
    for i := 1 to age.tope do  
        if (age.usuarios[i].est = Atendido) and  
            fechasIguales (age.usuarios[i].fechaAtendido, fec) then  
            begin  
                atendidos.tope := atendidos.tope + 1;  
                atendidos.usuarios[atendidos.tope] := age.usuarios[i]  
            end  
    end;
```

Ejercicio 2

Parte a

```
procedure agregaPrincipio (elem: char; var list: Lista);  
var p : lista;  
begin  
    new(p);  
    p^.elem := elem;  
    p^.sig := list;  
    list := p  
end;
```

Parte b

```
procedure soloAlfaNumericos (list: Lista; var resultado: Lista);  
var p : Lista;  
begin  
    resultado := NIL;  
    p := list;  
    while p <> NIL do  
        begin  
            if (esAlfaNumerico (p^.elem)) then  
                agregaPrincipio (p^.elem, resultado);  
            p := p^.sig  
        end  
    end  
end;
```