

2. Dados los tipos enumerados

```
type mes = enumerate
    enero
    febrero
    ...
    diciembre
end enumerate
```

```
type clima = enumerate
    Temp
    TempMax
    TempMin
    Pres
    Hum
    Prec
end enumerate
```

El arreglo `med:array[1980..2016,enero..diciembre,1..28,Temp..Prec]` **of nat** es un arreglo multidimensional que contiene todas las mediciones estadísticas del clima para la ciudad de Córdoba desde el 1/1/1980 hasta el 28/12/2016. Por ejemplo, `med[2014,febrero,3,Pres]` indica la presión atmosférica que se registró el día 3 de febrero de 2014. Todas las mediciones están expresadas con números enteros. Por simplicidad asumiremos que todos los meses tienen 28 días.

- (a) Dar un algoritmo que obtenga la menor temperatura mínima (TempMin) histórica registrada en la ciudad de Córdoba según los datos del arreglo.

```
fun menor_temp_hist(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat)
    ret res: nat

    var temp_min: nat

    temp_min := ∞

    for año := 1980 to 2016 do
        for mes := enero to diciembre do
            for día := 1 to 28 do
                if med[año,mes,día,TempMin] ≤ temp_min then
                    temp_min := med[año,mes,día,TempMin]
                fi
            od
        od
    od

    k := temp_min
end fun
```

- (b) Dar un algoritmo que devuelva un arreglo que registre para cada año entre 1980 y 2016 la mayor temperatura máxima (TempMax) registrada durante ese año.

```
fun mayor_temp_por_año(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat)
    ret res: array[1980..2016] of nat

    var temp_max: nat

    for año := 1980 to 2016 do
        temp_max := med[año,enero,1,TempMax]

        for mes := enero to diciembre do
            for día := 1 to 28 do
                if med[año,mes,día,TempMax] ≥ temp_max then
                    temp_max := med[año,mes,día,TempMax]
                fi
            od
        od
    od
end fun
```

```

        res[año] := temp_max
    od
end fun

```

- (c) Dar un algoritmo que devuelva un arreglo que registre para cada año entre 1980 y 2016 el mes de ese año en que se registró la mayor cantidad mensual de precipitaciones (Prec).

```

fun total_prec_del_mes(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat,
    a: nat, m: mes) ret res: nat
    res := 0

    for dia := 1 to 28 do
        res := res + med[año,mes,dia,Prec]
    od
end fun

fun mes_mayor_prec(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat)
    ret res: array[1980..2016] of mes

    var mayor_prec_del_año: mes
    var prec_del_mes: nat

    mayor_prec_del_año := enero

    for año := 1980 to 2016 do
        for mes := enero to diciembre do
            prec_del_mes := total_prec_del_mes(med,año,mes)

            if total_prec_del_mes(med,año,mayor_prec_del_año) ≤ prec_del_mes then
                mayor_prec_del_año := mes
            fi
        od
        res[año] := mayor_prec_del_año
    od
end fun

```

- (d) Dar un algoritmo que utilice el arreglo devuelto en el inciso anterior (además de med) para obtener el año en que ese máximo mensual de precipitaciones fue mínimo (comparado con los de otros años).

```

fun total_prec_del_mes(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat,
    a: nat, m: mes) ret res: nat
    res := 0

    for dia := 1 to 28 do
        res := res + med[año,mes,dia,Prec]
    od
end fun

fun mes_men_mayprec_hist(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat,
    a: array[1980..2016] of mes) ret res: nat

    var prec_min: nat
    var mes_mayor_prec: mes

    prec_min := ∞

    for año := 1980 to 2016 do

```

```

        mes_mayor_prec := total_prec_del_mes(med,año,a[año])
        if mes_mayor_prec ≤ prec_min then
            prec_min := mes_mayor_prec
            res := año
        fi
    od
end fun

```

(e) Dar un algoritmo que obtenga el mismo resultado sin utilizar el del inciso (c).

```

fun total_prec_del_mes(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat,
    a: nat, m: mes) ret res: nat
    res := 0
    for dia := 1 to 28 do
        res := res + med[año,mes,dia,Prec]
    od
end fun

fun comp_prec_del_año(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat,
    a: nat) ret res: mes
    var prec_del_mes: nat

    res := enero

    for mes := febrero to diciembre do
        prec_del_mes := total_prec_del_mes(med,año,mes)

        if total_prec_del_mes(med,año,res) ≤ prec_del_mes then
            res := mes
        fi
    od
end fun

fun mes_men_mayprec_hist(med: array[1980..2016,enero..diciembre,1..28,Temp..Prec] of nat)
    ret res: nat

    var minimo: nat
    var mes_mayor_prec: mes
    var prec_del_mes: nat

    minimo := ∞
    res := 0

    for año := 1980 to 2016 do
        mes_mayor_prec := comp_prec_del_año(med,año)
        prec_del_mes := total_prec_del_mes(med,año,mes_mayor_prec)

        if prec_del_mes ≤ minimo then
            minimo := prec_del_mes
            res := año
        fi
    od
end fun

```