

# Práctico 2.1: Ejercicio 2

...

Algoritmos y Estructuras de Datos II - 2020  
FaMAF - UNC

# Práctico 2.1: Ejercicio 2(c): Meses de más lluvia

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.

(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).

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

- Características del problema:
  - Arreglo multidimensional: Algunas dimensiones se indexan con tipos enumerados: `mes` y `clima`.
  - Resultado: Un arreglo de índices con años y valores con meses (`array[1980..2016] of mes`).

# Ejemplo (simplificado): Arreglo med

índices (posiciones)				valores
año	mes	día	clima	nat
1980	enero	1	Temp	30
			Prec	0
		2	Temp	29
			Prec	1
	febrero	1	Temp	20
			Prec	19
		2	Temp	16
			Prec	25
1981	enero	1	Temp	31
			Prec	11
		2	Temp	35
			Prec	12
	febrero	1	Temp	40
			Prec	9
		2	Temp	57
			Prec	7

med[1981, febrero, 1, Temp]



# Ejemplo (simplificado): Ejercicio 2(c)

índices (posiciones)				valores	Prec mensual (suma)
año	mes	día	clima	nat	
1980	enero	1	Temp	30	1
			Prec	0	
		2	Temp	29	
			Prec	1	
	febrero	1	Temp	20	44
			Prec	19	
		2	Temp	16	
			Prec	25	
1981	enero	1	Temp	31	23
			Prec	11	
		2	Temp	35	
			Prec	12	
	febrero	1	Temp	40	16
			Prec	9	
		2	Temp	57	
			Prec	7	

# Ejemplo (simplificado): Ejercicio 2(c)

índices (posiciones)				valores	Prec mensual (suma)	Máximo anual
año	mes	día	clima	nat		
1980	enero	1	Temp	30	1	44 (febrero)
			Prec	0		
		2	Temp	29		
			Prec	1		
	febrero	1	Temp	20	44	
			Prec	19		
		2	Temp	16		
			Prec	25		
1981	enero	1	Temp	31	23	23 (enero)
			Prec	11		
		2	Temp	35		
			Prec	12		
	febrero	1	Temp	40	16	
			Prec	9		
		2	Temp	57		
			Prec	7		

# Ejemplo (simplificado): Ejercicio 2(c)

índices (posiciones)				valores	Prec mensual (suma)	Máximo anual
año	mes	día	clima	nat		
1980	enero	1	Temp	30	1	44 (febrero)
			Prec	0		
		2	Temp	29		
			Prec	1		
	febrero	1	Temp	20	44	
			Prec	19		
		2	Temp	16		
			Prec	25		
1981	enero	1	Temp	31	23	23 (enero)
			Prec	11		
		2	Temp	35		
			Prec	12		
	febrero	1	Temp	40	16	
			Prec	9		
		2	Temp	57		
			Prec	7		

- Resultado:
 

febrero	enero
1980	1981

 (de tipo `array[1980..1981] of mes`)

# Construcción Incremental del Algoritmo

```
var res: array[1980..2016] of mes
var max_mes: mes

for año := 1980 to 2016 do
  {- acá calcular max_mes para este año -}

  res[año] := max_mes
od
```

# Construcción Incremental del Algoritmo

```
var res: array[1980..2016] of mes
var max_mes: mes
var max_prec_mes, prec_mes: nat
```

```
for año := 1980 to 2016 do
```

```
    max_prec_mes := 0
    for mes := enero to diciembre do
        {- acá calcular la suma prec_mes para este mes del año -}

        if prec_mes >= max_prec_mes:    <- ojo acá con el >= !
            max_prec_mes := prec_mes
            max_mes := mes
```

```
    res[año] := max_mes
od
```



# Construcción Incremental del Algoritmo

```
var res: array[1980..2016] of mes
var max_mes: mes
var max_prec_mes, prec_mes: nat

for año := 1980 to 2016 do
  max_prec_mes := 0
  for mes := enero to diciembre do
    prec_mes := 0
    for día := 1 to 28 do
      prec_mes := prec_mes + med[año,mes,día,Prec]
    od
    if prec_mes >= max_prec_mes:
      max_prec_mes := prec_mes
      max_mes := mes

  res[año] := max_mes
od
```

# Resultado Final

```
var res: array[1980..2016] of mes
var max_mes: mes
var max_prec_mes, prec_mes: nat

for año := 1980 to 2016 do
  max_prec_mes := 0
  for mes := enero to diciembre do
    prec_mes := 0
    for dia := 1 to 28 do
      prec_mes := prec_mes + med[año,mes,dia,Prec]
    od
    if prec_mes >= max_prec_mes:
      max_prec_mes := prec_mes
      max_mes := mes

  res[año] := max_mes
od
```

# Complejidad

```
var res: array[1980..2016] of mes
var max_mes: mes
var max_prec_mes, prec_mes: nat
```

```
for año := 1980 to 2016 do
  max_prec_mes := 0
  for mes := enero to diciembre do
    prec_mes := 0
    for día := 1 to 28 do
      prec_mes := prec_mes + med[año,mes,día,Prec]
    od
    if prec_mes >= max_prec_mes:
      max_prec_mes := prec_mes
      max_mes := mes

  res[año] := max_mes
od
```

<- 2016 - 1980 = 36 iteraciones (años)

<- x 12 iteraciones (meses)

<- x 28 iteraciones (días)

<- años x meses x días = 36 x 12 x 28

# Consideraciones Finales

- Arreglos multidimensionales + tipos enumerados:
  - Declaración: `med: array[1980..2016, enero..diciembre, 1..28, Temp..Prec] of nat`
  - Acceso: `med[1981, febrero, 1, Temp]`.
- Complejidad del algoritmo:  $|años| \times |meses| \times |dias|$ .
  - Lineal en la cantidad de años, meses y días.
  - No interviene el tipo `clima` en la complejidad.

**¡Gracias!**