



**PRÁCTICA 5.**  
**FUNDAMENTOS DE PROGRAMACIÓN:**  
**Ejercicio sobre el sistema D'Hondt.**

**Arturo Hidalgo López**  
**Ángel Fidalgo Blanco**

*Noviembre, 2020*

## *Sistema electoral d'Hondt*

El sistema D'Hondt es un método utilizado para calcular el número de diputados que consigue cada partido tras unas elecciones. Debe su nombre al jurista belga Victor D'Hondt, que la creó en 1878.



Su funcionamiento es el siguiente: Después de unas elecciones, se cuentan los votos de cada uno de los partidos que se ha presentado en una circunscripción (en el caso español, la provincia o Ciudad Autónoma).

Supongamos que en una provincia en la que se han presentado 7 partidos se reparten 6 escaños

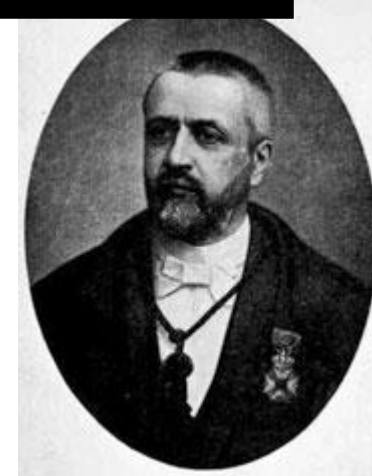
## *Sistema electoral d'Hondt*

Partidos	Votos
Partido A	500 000
Partido B	300 000
Partido C	250 000
Partido D	130 000
Partido E	80 000
Partido F	20 000
Partido G	15 000



## *Sistema electoral d'Hondt*

Partidos	Votos
Partido A	500 000
Partido B	300 000
Partido C	250 000
Partido D	130 000
Partido E	80 000
<del>Partido F</del>	<del>20 000</del>
<del>Partido G</del>	<del>15 000</del>



Se excluyen los partidos que tienen menos del 3% de los votos totales:

Suma votos= 1295000 ; 3% Suma Votos=38850

## *Sistema electoral d'Hondt*

	Votos
Partido A	500000
Partido B	300000
Partido C	250000
Partido D	130000
Partido E	80000



# *Sistema electoral d'Hondt*

## Primer escaño

	Votos
Partido A	500000
Partido B	300000
Partido C	250000
Partido D	130000
Partido E	80000



## *Sistema electoral d'Hondt*

### *Segundo escaño*

	Votos	Votos/2
Partido A	500000	250000
Partido B	300000	150000
Partido C	250000	125000
Partido D	130000	65000
Partido E	80000	40000



## Sistema electoral d'Hondt

### Tercer escaño

	Votos	Votos/2	Votos/3
Partido A	500000	250000	166667
Partido B	300000	150000	100000
Partido C	250000	125000	83333
Partido D	130000	65000	43333
Partido E	80000	40000	26667





## Sistema electoral d'Hondt

### Cuarto escaño

	Votos	Votos/2	Votos/3	Votos/4
Partido A	500000	250000	166667	125000
Partido B	300000	150000	100000	75000
Partido C	250000	125000	83333	62500
Partido D	130000	65000	43333	32500
Partido E	80000	40000	26667	20000



## Sistema electoral d'Hondt

### Quinto escaño

	Votos	Votos/2	Votos/3	Votos/4	Votos/5
Partido A	500000	250000	166667	125000	100000
Partido B	300000	150000	100000	75000	60000
Partido C	250000	125000	83333	62500	50000
Partido D	130000	65000	43333	32500	26000
Partido E	80000	40000	26667	20000	16000



## Sistema electoral d'Hondt

### Sexto escaño

	Votos	Votos/2	Votos/3	Votos/4	Votos/5	Votos/6
Partido A	500000	250000	166667	125000	100000	83333
Partido B	300000	150000	100000	75000	60000	50000
Partido C	250000	125000	83333	62500	50000	41667
Partido D	130000	65000	43333	32500	26000	21667
Partido E	80000	40000	26667	20000	16000	13333



**Partido A: 3 escaños.**

**Partido B: 2 escaños.**

**Partido C: 1 escaño.**





En este ejercicio, determinaremos el reparto de escaños entre los partidos políticos como si toda España fuese una misma circunscripción. Utilizaremos los resultados de las elecciones del 10 de noviembre de 2019.

No aplicaremos el criterio del 3%

Partido	Votos
PSOE	6752983
PP	5019869
VOX	3640063
UP	3097185
Cs	1637540
ERC	869934
MasPais	577055
JxCat	527375
PNV	377423
EH-Bildu	276519
CUP	244754
PACMA	226469
CC	123981
BNG	119597
Navarra+	98448
PRC	68580
Teruel Existe	19696

## Pseudo-código

### PRIMER PASO:

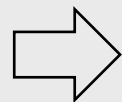
1. PROGRAMAR: `asignacion_escaños<-function(votos,npart,nesc) {`  
SALIDA: `escaños_unica` con los siguientes pasos:

- Inicializar los contadores: `k=1; cuenta_escan=1`
- Inicializar a 0 una matriz `A` de `npart` filas y `nesc` columnas.
- Inicializar el vector `escaños_unica = 0`.  
Además, hacer: `escaños_unica[i]=0, (i=1,...npart)`
- Mientras (`cuenta_escan<=nesc`) hacer:
  - Almacenar en la columna `k` de `A` la cantidad: `votos[i]/k (i=1,...,npart)`
  - Encontrar el mayor valor de la matriz `A` (filas `1:npart` ; columnas `1:k`) almacenando el valor en la variable `max`, su fila en `imax` y su columna en `jmax`.
  - Hacer `A[imax,jmax]=0` *para no volverlo a utilizar*.
  - Hacer `escaños_unica [imax]=escaños_unica [imax]+1; k=k+1;`  
`cuenta_escan=cuenta_escan+1`

Fin del bucle

`return (escaños_unica)`

Fin function }



*asignación de variable de salida*



## Pseudo-código

En el programa principal:

1. Introducir los DATOS.

### DATOS:

```
nesc=350; npart=17
votos=c(6752983, 5019869, 3640063, 3097185, 1637540, 869934, 577055, 527375,
377423, 276519, 244754, 226469, 123981, 119597, 98448, 68580, 19696)
Partido=c('PSOE', 'PP', 'VOX', 'PODEMOS', 'Cs', 'ERC', 'MasPais', 'JxCAT', 'PNV', 'BILDU',
'CUP', 'PACMA', 'CC', 'BNG', 'NAVARRA_SUMA', 'PRC', 'TeruelExiste');
escaños_reales=c(120, 89, 52, 35, 10, 13, 8, 8, 6, 5, 2, 0, 2, 1, 2, 1, 1);
```

2. Llamar a la función: `escañitos <- asignacion_escaños(votos, npart, nesc)`

3. Construir la tabla: `T = data.frame(Partido, escañitos, escaños_reales)`  
y Escribir: `T`



## Pseudo-código

$nvotos = \text{length}(\text{votos})$

```
Para i desde 1 hasta nvotos hacer
  Para j desde 1 hasta (nvotos-1) hacer
    Si (votos[j] < votos[j+1])
      Intercambiar:
      votos[j] con votos[j+1]
      Partido[j] con Partido[j+1]
      escaños_reales [j] con escaños_reales [j+1]
    Fin condición
  Fin bucle
Fin bucle
```

*Ordenamos los partidos de mayor a menor número de votos (Algoritmo de la burbuja)*



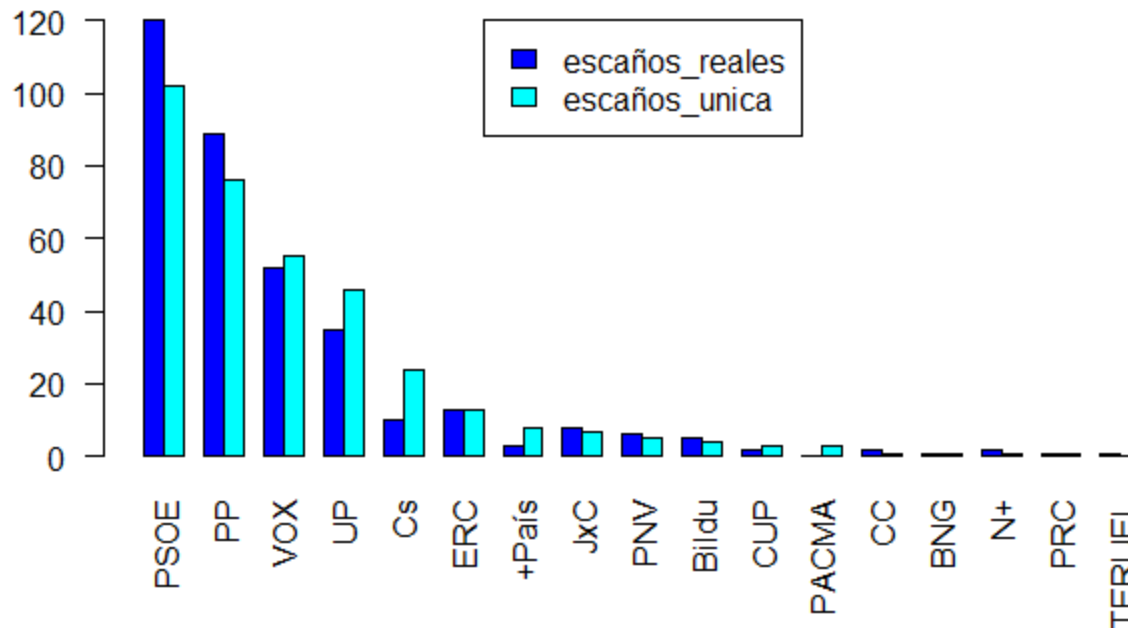
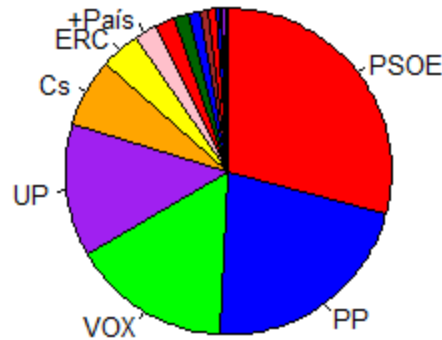


## Pseudo-código

### POST-PROCESO:

6. `par(mfrow=c(2,1),mar=c(5,4,0.01,1))` Para colocar 2 gráficos distribuidos en 2 filas y 1 columna.
7. `names(escañitos) <- c('PSOE', 'PP', 'VOX', 'UP', 'Cs', 'ERC', '+País')`
8. `pie(escañitos, clockwise=TRUE,col=c('red', 'blue', 'green', 'purple', 'orange', 'yellow', 'pink', 'red', 'dark green', 'blue', 'brown'),cex=0.8)` *#PARA GRÁFICO SECTORES CIRCULARES*
9. `names(escaños_reales) <- c('PSOE', 'PP', 'VOX', 'UP', 'Cs', 'ERC', '+País', 'JxC', 'PNV', 'Bildu', 'CUP', 'PACMA', 'CC', 'BNG', 'N+', 'PRC', 'TERUEL')`
10. `escanos <- rbind(escaños_reales, escañitos)`
11. `barplot(escanos, beside=TRUE, col=c(4,5), las=2)` *#PARA GRÁFICO BARRAS SUPERPUESTAS*
12. `legend(x='top', rownames(escanos), col=1:17, fill = c(4,5))`

*NOTA: para que las barras sean horizontales se pondría:  
horiz=1, las=1 (horiz gira las barras y las gira los nombres).*





ESCUELA TÉCNICA SUPERIOR  
DE INGENIEROS DE MINAS Y ENERGÍA

