

3. Listas: Tablas, matrices y vectores

Ejercicios resueltos

■ Ejercicio 1.

a. Dada la función $f(x) = x!$ generar una tabla que resulte al evaluar dicha función para los valores $x \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

b. Generar otra tabla que resulte de evaluar la función $g(x, y) = x^2 + y^2$, donde $x \in \{0, 1, 2, 3, 4\}$ e $y \in \{5, 6, 7, 8\}$.

Solución:

a. Introducimos la función f de la forma

```
f[x_] := x!
```

A continuación metemos en una lista los distintos valores de la variable x

```
lista = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
```

Ahora utilizamos la función `Table` para crear una lista con los resultados de evaluar f

```
tabla1 = Table[f[lista[[i]]], {i, 10}]
```

```
{1, 1, 2, 6, 24, 120, 720, 5040, 40320, 362880}
```

Por último, lo pondremos en forma de tabla usando `TableForm[]`

TableForm[tabla1]

```
1
1
2
6
24
120
720
5040
40320
362880
```

b. Introducimos la función g , teniendo en cuenta que tiene dos variables, de la forma

```
g[x_, y_] := x^2 + y^2
```

A continuación, metemos los distintos valores de las variables x e y en dos listas que llamaremos 1 y 2 respectivamente

```
lista1 = {0, 1, 2, 3, 4};
lista2 = {5, 6, 7, 8};
```

Ahora utilizamos la función `Table` para crear una tabla de doble entrada con los resultados de evaluar g en x e y

```
tabla2 = Table[g[lista1[[i]], lista2[[j]]], {i, 5}, {j, 4}]

{{25, 36, 49, 64}, {26, 37, 50, 65},
 {29, 40, 53, 68}, {34, 45, 58, 73}, {41, 52, 65, 80}}
```

Por último, lo pondremos en forma de tabla usando `TableForm[]`

TableForm[tabla2]

25	36	49	64
26	37	50	65
29	40	53	68
34	45	58	73
41	52	65	80

Observemos que la primera fila corresponde a los valores $i = 0, j = 5, 6, 7, 8$. La segunda sería para $i = 1$ y todas las posibilidades de j , y así sucesiva-

mente hasta llegar a $i = 4$ y $j = 5, 6, 7, 8$.

La generación de las tablas podrían haberse simplificado pues no es imprescindible el uso de las listas para evaluar las variables, sino que en este caso, los contadores i y j nos habrían bastado. Por ejemplo:

```
tabla3 = Table[g[i, j], {i, 0, 4}, {j, 5, 8}];
TableForm[tabla3]
```

25	36	49	64
26	37	50	65
29	40	53	68
34	45	58	73
41	52	65	80

■ Ejercicio 2.

a. Crear un vector cuyos coeficientes sean los números positivos menores que 100 y múltiplos de 5.

b. Crear un vector que contenga los cuadrados de los múltiplos de 15 menores de 1000.

Solución:

a. Llamamos $v1$ al vector que buscamos

```
v1 = Table[5 * i, {i, 100 / 5 - 1}]
```

```
{5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95}
```

Observemos que el contador va desde $i = 1$ hasta el valor del contador para el cual obtenemos el último múltiplo de 5 menor estricto que 100, por eso hemos incluido como final del bucle el número $(100/5) - 1$, esto es, 19.

b. Llamemos $v2$ al vector que vamos a calcular

```
v2 = Table[(15 * i)^2, {i, 2}]
```

```
{225, 900}
```

Observemos que para $i = 3$, $(15*3)^2$ es 2025 que se pasa de la cota que nos han marcado, por tanto, el contador tendrá como valor final $i = 2$.

■ Ejercicio 3.

Crear una matriz 10×10 que contenga un 1 en las componentes a_{ij} tales que $i + j$ es par, y un -1 en las que $i + j$ es impar.

Solución:

Observemos que tenemos que buscar una expresión que de 1 o -1 en las condiciones que nos piden. Así, por ejemplo, $(-1)^{i+j}$ verifica lo que pretendemos. Llamamos `tablaej3` a la matriz que buscamos

```
tablaej3 = Table[(-1)^(i + j), {i, 10}, {j, 10}]
```

```
{ {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1} }
```

También podríamos usar la función `Mod[i+j,2]` para determinar si $i+j$ es par o impar y la tabla podría definirse

```
tablaej3 = Table[-2 * Mod[i + j, 2] + 1, {i, 10}, {j, 10}]
```

```
{ {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1},
  {1, -1, 1, -1, 1, -1, 1, -1, 1, -1}, {-1, 1, -1, 1, -1, 1, -1, 1, -1, 1} }
```

Por último, la función `MatrixForm[]` pondrá la lista anterior en forma de matriz

MatrixForm[tablaej3]

$$\begin{pmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{pmatrix}$$

■ Ejercicio 4.

Crear una tabla que simbolice un tablero de ajedrez de manera que en las casillas negras aparezca un 0 y en las casillas blancas aparezca un 1.

Solución:

Llamamos tablaej4 a la lista que construiremos. Será una tabla 8×8 en la que se alternan los ceros y los unos que conseguimos como restos de dividir por 2:

tablaej4 = Table[Mod[i + j, 2], {i, 8}, {j, 8}]

```
{{0, 1, 0, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 0, 1, 0},
 {0, 1, 0, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 0, 1, 0}, {0, 1, 0, 1, 0, 1, 0, 1},
 {1, 0, 1, 0, 1, 0, 1, 0}, {0, 1, 0, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 0, 1, 0}}
```

Por último, la función TableForm[] pondrá la lista anterior en forma de tabla

TableForm[tablaej4]

0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0

Usando el 0 y el 1 a modo de índices para acceder a los valores de una lista, podríamos colocar en las componentes de la matriz caracteres que realmente simbolicen los recuadros del tablero de ajedrez. De esta forma observamos que las listas, tablas, matrices o vectores, además de números pueden contener también caracteres

```
caracteres = {"■", "□"};  
TableForm[Table[caracteres[[Mod[i + j, 2] + 1]], {i, 8}, {j, 8}]]
```

■	□	■	□	■	□	■	□
□	■	□	■	□	■	□	■
■	□	■	□	■	□	■	□
□	■	□	■	□	■	□	■
■	□	■	□	■	□	■	□
□	■	□	■	□	■	□	■
■	□	■	□	■	□	■	□
□	■	□	■	□	■	□	■