

Determinación del espacio muestral y los eventos

El espacio muestral y los eventos como cualquier conjunto se pueden determinar de dos maneras:

- Por extensión o
- por comprensión.

Un conjunto está definido por extensión si se enumeran sus elementos. Por ejemplo el conjunto de los colores primarios

$A = \{\text{'rojo'}, \text{' verde'}, \text{' azul'}\}$ Para enumerar los elementos de un espacio muestral dado por experimentos de tipo lanzamiento, por ejemplo monedas, dados etc., se puede usar una serie de ciclos for anidados.

Ejemplo 1. Escriba un código para determinar por enumeración el espacio muestral del experimento que consiste en lanzar una moneda dos veces.

Algoritmo 1. Enumeración de conjuntos mediante ciclos for

```
posibles = {'cara ', 'cruz '}
```

```
espacio_muestral = set ()
```

```
for tiro1 in posibles :
```

```
    for tiro2 in posibles :
```

```
        salida = (tiro1 , tiro2)
```

```
        espacio_muestral .add( salida )
```

```
print (espacio_muestral)
```

El resultado de la ejecución de este código es:

```
{( ' cara ', 'cara '), ( ' cara ', 'cruz '), ( ' cruz ', 'cruz '), ( ' cruz ', 'cara ')}
```

Análisis del código

- $\{e_1, e_2, \dots\}$ - define un conjunto por enumeración;
- `set()`- crea un conjunto vacío;
- `for x in X:-` x toma los valores de X iterativamente;
- `conjunto.add-` añade un elemento a conjunto.

El espacio muestral en este ejemplo corresponde al producto cartesiano del conjunto original por si mismo. Por ejemplo, en el lanzamiento de dos dados el producto cartesiano es el siguiente:

		<i>Segundo lanzamiento</i>					
		1	2	3	4	5	6
<i>Primer lanzamiento</i>	1	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)
	2	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)
	3	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)
	4	(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)
	5	(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)
	6	(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)

Cuadro 1.1: Producto cartesiano de un conjunto por si mismo

Si se quisiera lanzar la moneda tres veces o más veces se podría usar más ciclos for
for a in X:

for b in X:

.

.

for w in X:

Sin embargo, no sería muy eficiente. Una forma más conveniente es usar el módulo itertools

Ejemplo 2. Escriba un código para determinar por enumeración el espacio muestral correspondiente al lanzamiento triple de una pirinola de tres caras, cuyos rótulos son: 'A', 'B', 'C'.

Resolución

Algoritmo 1.7 Enumeración de conjuntos con el módulo itertools

```
from itertools import product
```

```
posibles = {'A' , 'B' , 'C'}
```

```
espacio_muestral = set(product(posibles ,  
repeat = 3))
```

```
print (espacio_muestral)
```

El resultado de la ejecución de este código es:

```
{('B', 'B', 'B'), ('B', 'C', 'B'), ('C', 'A', 'A'), ('C', 'B', 'A'), ('A', 'A', 'A'), ('C', 'C', 'A'), ('A', 'C', 'A'), ('C', 'A', 'C'), ('C', 'B', 'C'), ('A', 'B', 'A'), ('B', 'A', 'A'), ('A', 'A', 'C'), ('B', 'B', 'A'), ('C', 'C', 'C'), ('B', 'C', 'A'), ('A', 'C', 'C'), ('A', 'B', 'C'), ('B', 'A', 'C'), ('B', 'B', 'C'), ('B', 'C', 'C'), ('C', 'A', 'B'), ('C', 'B', 'B'), ('A', 'A', 'B'), ('C', 'C', 'B'), ('A', 'C', 'B'), ('B', 'A', 'B'), ('A', 'B', 'B')}
```

Análisis del código

- from módulo import función- importa solo una función de un módulo;
- itertools- módulo que contiene funciones iterativas;
- product- función que implementa el producto cartesiano;
- repeat = n- establece las veces que se ejecuta 'product'.

Ejemplo 3 . Escriba un código para determinar por enumeración el espacio muestral del lanzamiento triple de un dado.

El resultado (fragmento) de la ejecución de este código es: {(5 , 1, 6), (5, 3, 3), (5, 4, 2), (2 , 1, 6), (1, 6, 6), (2, 2, 5), (6 , 6, 4), ...}

Ejemplo 4. Escriba un código en Python para enumerar el evento $E = \{\text{puntaje menor que 4}\}$ a partir de su descripción formal en el lanzamiento de un dado

El resultado de la ejecución de este código es: $E = \{1, 2, 3\}$

Ejemplo 5. Escriba un código en Python para determinar los elementos de un evento si debe cumplir la condición que la suma de los puntos en dos lanzamientos de un dado no sea mayor que 7.

El resultado de la ejecución de este código es: {(2 , 4), (1, 2), (2, 1), (1, 5), (3, 1), (4 , 1), (1, 1), (5, 1), (4, 2), (1, 4), (2 , 3), (3, 3), (2, 2), (3, 2), (1, 3)}