**Caminatas en graficas: los recorridos**

por *f6m*

Un *recorrido* (traducción del inglés trail) $R$ es una secuencia de vértices adyacentes de que no repite aristas y se dividen en recorridos abiertos (open trail) y cerrados (closed trail). Si el vértice inicial es distinto que el vértice final el recorrido es abierto, si el vértice inicial es igual al vértice final el recorrido es cerrado. Algunos autores indican “un $u - v$-recorrido” para indicar que el vértice inicial es $u$ y el vértice final es $v$.

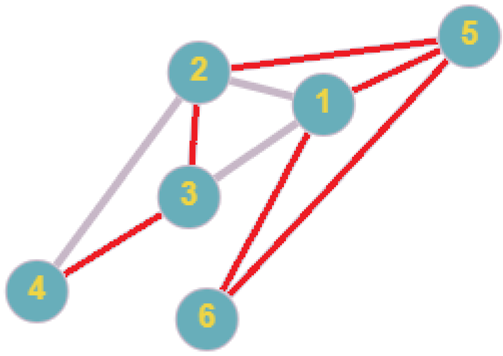



**Figura** 1: Un recorrido abierto en una gráfica. Inicia en el vértice 5 y termina en el vértice 4. La secuencia de vértices es R:5,6,1,5,2,3,4.

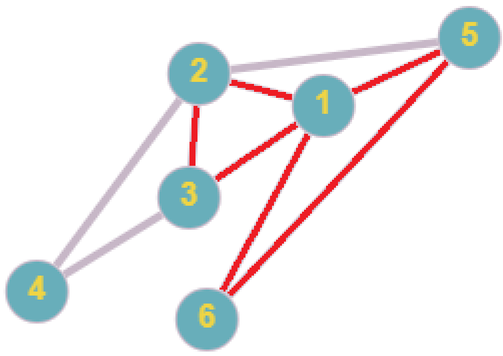



**Figura** 2: Un recorrido cerrado en una gráfica. Por ejemplo, inicia en el vértice 5 y termina en el vértice 4. La secuencia de vértices es R:5,6,1,3,2,1,5.

Si únicamente se indica “un recorrido” $R$ se entendería que es un recorrido abierto pues para recorridos cerrados tenemos el termino circuito (del inglés circuit).

Los circuitos tienen una estrecha relación con la historia de la teoría de gráficas, en especial con el problema de los puentes de Königsberg que Leonard Euler resolvió en 1736. En ese entonces no se indicaba bajo el termino actual “trail” o recorrido sino usando conjuntos. Actualmente debido al problema de los puentes de Königsberg y a la solución de Euler tenemos una clasificación de los recorridos en recorridos de Euler (o eulerianos) como aquellos que contienen todas las aristas de la gráfica donde se encuentran. Desde entonces ahora tenemos recorridos abiertos de Euler (si el vértice inicial y final son distintos) y recorridos cerrados de Euler o circuitos de Euler (o brevemente circuitos eulerianos).

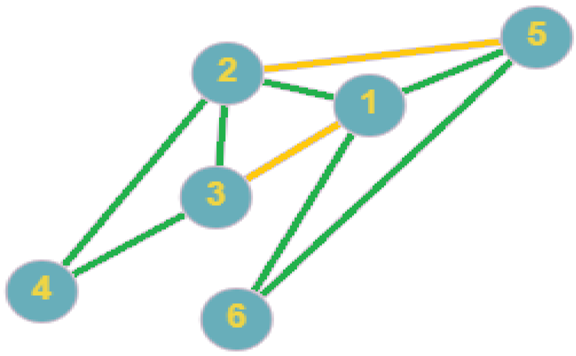



**Figura** 3: Un recorrido abierto de Euler. Inicia en el vértice 5 y termina en el vértice 3. Las aristas en color amarillo marcan el inicio del recorrido.

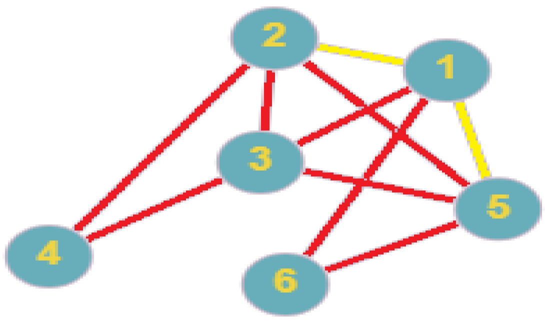



**Figura** 4: Un circuito de Euler. Inicia en el vértice 1 y termina en el vértice 1. La secuencia de vértices es $\mathrm{RE}: 1,2,3,4,2,5,3,1,6,5,1$. Las aristas en color amarillo marcan el inicio del circuito.

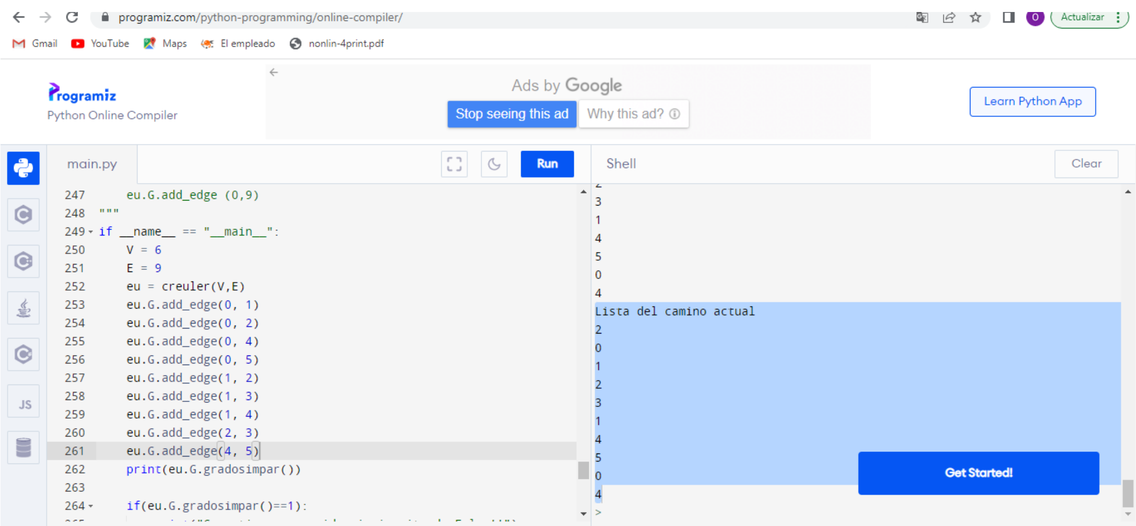
Pero si una gráfica tiene un recorrido o circuito de Euler ¿cómo determinarlo? La respuesta es mediante un algoritmo. Actualmente existen varios para este propósito y software libre o comercial para estos propósitos. Una implementación de uno de estos algoritmos para calcular el recorrido abierto o circuito de Euler de una gráfica la podemos encontrar en https://github.com/f6m/GraphAlgorithms

Vamos a utilizar el programa indicado para determinar el recorrido de Euler de la figura 3.

Primero introducimos la gráfica de la figura 3 dentro del programa, indicando orden y tamaño y aristas, así:

```
if __name__ == "__main__":
    V = 6
    E = 9
    eu = creuler(V,E)
    eu.G.add_edge(0, 1)
    eu.G.add_edge(0, 2)
    eu.G.add_edge(0, 4)
    eu.G.add_edge(0, 5)
    eu.G.add_edge(1, 2)
    eu.G.add_edge(1, 3)
    eu.G.add_edge(1, 4)
    eu.G.add_edge(2, 3)
    eu.G.add_edge(4, 5)
```

Segundo, cambiamos el código python y lo ejecutamos en línea (por ejemplo en [https://www.programiz.com/python-programming/online-compiler/](https://www.programiz.com/python-programming/online-compiler/)) y al final de la ejecución nos indica:




La ultima cadena indica la secuencia de vértices pero en el etiquetado de vértices 0,1, 2,…5 pero en nuestra imagen los vértices tienen las etiquetas 1,…,6 por lo que 2012314504 en nuestra imagen seria el recorrido 3123425615.

## Referencias


*f6m* (2021). Algoritmos en gráficas – Recorridos y circuitos de Euler (Versión 0.0.1) [Software]. Disponible en [https://github.com/f6m/GraphAlgorithms](https://github.com/f6m/GraphAlgorithms)