**Các thành phần tham khảo**

Init() :

* là một phương thức, nó được gọi trong constructor
* chứa những câu lệnh khởi tạo thành phần hay giá trị của các biến

pack() : giúp frame có kích thước vừa đủ với nội dung của frame ngay cả khi chúng ta đặt kích thước frame lớn hơn.

jframe

**setLocationRelativeTo :** . Nếu muốn frame xuất hiện chính giữa màn hình

borderlayout

gridlayout :

* Lớp **GridLayout** sắp xếp các thành phần trong một lưới hình chữ nhật. Một thành phần được hiển thị trong mỗi hình chữ nhật.
* **GridLayout(int rows, int columns, int hgap, int vgap) : hàng , cột , dọc, ngang**

Emptyborder :

* EmptyBorder(top, left, bottom, right)

jCombobox

setlayout

serialization : đối tượng được biểu diễn như một dãy byte liên tục bao gồm dữ liệu của đối tượng , thông tin về kiểu đối tượng , kiểu dữ liệu được lưu trữ trong đối tượng

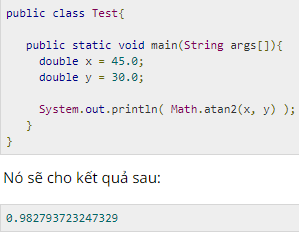
Ellipse2D.Float(float x , float y , float w , float h)

* X tọa độ góc trên bên trái của hình chữ nhật khung hình ellipse

Line2D.double : điểm bất đầu điểm kết thúc

* Creates a Line2D from (X1, Y1) to (X2, Y2).

math.atant2 :

* biến đổi tọa độ góc (x,y) thành tọa độ cực (r , theta) và trả về theta
* double atan2(double x , double y )
* 

setStroke : qui định kiểu vẽ cho đối tượng Graphics

Lớp BasicStroke là lớp định nghĩa một tập các cách vẽ khác nhau.

Cơ **chế vẽ của Java**

Bât cứ đoạn code nào bạn dùng để vẽ ra thứ gì cũng được đặt trong phương thức paintComponent(), phương thức này được override từ lớp JPanel. Phương thức này tự động được gọi bởi phương thức paint() (trong lớp java.awt.Component), ngoài phương thức paintComponent()còn có 2 phương thức khác được gọi cùng nữa là paintBorder() và paintChildren(). Nếu muốn bạn có thể override cả 2 phương thức đó cũng được nhưng thường thì cũng không cần thiết lắm. Trong series này thì chúng ta chỉ dùng đến paintComponent()

Graphics2d thừa kế từ Graphics cung cấp các phương thức cao cấp hơn

Repaint() : chương trình cập nhật lại vị trí của mỗi điểm

EULER :

* Trong đồ thị G :
  + được gọi là đường đi euler nó đi qua mỗi cạnh đúng 1 lần , có đúng 2 đỉnh bậc lẻ
  + Đường đi euler có đỉnh cuối trùng với đỉnh xuất phát gọi là chu trình euler
  + G vô hướng liên thông , mọi đỉnh có bậc chẳn => G có chu trình euler
  + Một đồ thị không có **chu trình Euler** nhưng vẫn có thể có **đường đi Euler**. Khi đó, đồ thị có đúng hai đỉnh bậc lẻ, tức là tổng các số cạnh xuất phát từ một trong hai đỉnh đó là số lẻ. Một **đường đi Euler** phải xuất phát từ một trong hai đỉnh đó và kết thúc ở đỉnh kia. Như vậy, thuật toán tìm **đường đi Euler** chỉ khác với thuật toán tìm **chu trình Euler** ở chỗ ta phải xác định điểm xuất phát của đường đi từ đỉnh bậc lẻ này và kết thúc ở đỉnh bậc lẻ khác.

Đồ thị liên thông

* Một đồ thị không liên thông gồm nhiều đồ thị con liên thông , các đồ thị con này gọi là các thành phần liên thông
* Đồ thị liên thông khi và chỉ khi có một thành phần liên thông

Thuật toán Fleury :

* Mỗi khi đi qua một cạnh nào đó thì xóa nó đi, sau đó xóa đỉnh cô lập (nếu có).
* Không bao giờ đi qua [cầu](https://vi.wikipedia.org/wiki/Thu%E1%BA%ADt_ng%E1%BB%AF_l%C3%BD_thuy%E1%BA%BFt_%C4%91%E1%BB%93_th%E1%BB%8B#C) trừ khi không còn cách đi nào khác.

\*Tạo một **mảng CE** để ghi đường đi và một **stack**để xếp các đỉnh ta sẽ xét. Xếp vào đó một đỉnh tuỳ ý **u** nào đó của đồ thị, nghĩa là đỉnh **u** sẽ được xét đầu tiên.   
\*Xét đỉnh trên cùng của ngăn xếp, giả sử đỉnh đó là đỉnh **v** và thực hiện:   
- Nếu **v** là đỉnh cô lập thì lấy **v** khỏi ngăn xếp và đưa vào **CE**;   
- Nếu **v** là liên thông với đỉnh **x** thì xếp **x** vào ngăn xếp sau đó xoá bỏ cạnh **(v, x)**;   
\*Quay lại bước 2 cho tới khi ngăn xếp rỗng. Kết quả chu trình Euler được chứa trong **CE**theo thứ tự ngược lại.

Boxlayout là một Layout cho phép chúng ta sắp xếp các control trên một hàng ngang hoặc trên một cột thẳng đứng.

Tltk :

* <http://fit.lqdtu.edu.vn/files/FileMonHoc/Ly%20Thuyet%20do%20thi.pdf>
* https://vi.wikipedia.org/wiki/%C4%90%E1%BB%93\_th%E1%BB%8B\_Euler

cầu : là 1 cạnh mà khi bỏ nó đi đồ thị không liên thông

https://vi.wikipedia.org/wiki/Thu%E1%BA%ADt\_ng%E1%BB%AF\_l%C3%BD\_thuy%E1%BA%BFt\_%C4%91%E1%BB%93\_th%E1%BB%8B#C