

Agregado de imágenes (entorno figure)

```
\includegraphic[y]{x}
```

En x va el nombre del archivo si está en la carpeta o el path en caso de que esté en otro lado. El nombre debe incluir la extensión.

En y se parametriza el tamaño:

scale=n Especifica el porcentaje de escala por sobre el tamaño de la imagen original.

width=medida Especifica el ancho de la imagen. Debe incluir la unidad de medida (pt, cm, in)

height=medida Especifica la altura de la imagen.

angle=grados Especifica el ángulo en que se posiciona la imagen.

Extensiones típicas de imágenes: jpg, png, eps.

Se puede configurar el *path* desde el preámbulo mediante el siguiente comando:

```
\graphicspath{ {path} }
```

Se puede incluir la figura en un entorno especial llamado figure:

```
\begin{figure}
```

```
...
```

```
\end{figure}
```

Agregando `\caption{}` se puede incluir un epígrafe para la figura.

Después de `\begin{figure}` se puede especificar entre corchetes la posición en la que queremos que aparezca la imagen.

h: la posición se da aproximadamente en donde está en el código.

t: la figura se coloca al principio de la página.

b: la figura va al final de la página.

p: las figuras se colocan en una página parte.

!: Sirve para que LaTeX ignore los parámetros que tiene para posicionar objetos flotantes.

Se puede hacer que la figura esté envuelta por el texto utilizando el paquete wrapfig y los siguientes comandos:

```
begin{wrapfigure}{r}{0.3\textwidth}
```

 Destina el 30% de la línea ubicado a la derecha para incluir la imagen.

```
begin{wrapfigure}{l}{0.3\textwidth}
```

 Destina el 30% de la línea ubicado a la izquierda para incluir la imagen.

Gráficos (paquete tikz)

```
\usepackage{tikz}
```

Las imágenes se enmarcan en el entorno tikz

```
\begin{tikzpicture}
```

```
...
```

```
\end{tikzpicture}
```

La lógica detrás de tikz es la de un diagrama con ejes cartesianos. Cada orden de dibujo se prefija como `\draw` y termina en punto y coma (;).

```
\draw (x,y) circle (medida);
```

círculo con centro en el punto x del eje horizontal e y del eje vertical

```
\draw (x1,y1) -- (x2,y2)
```

línea que une el punto definido por x1 e y1 con el punto definido por x2 e y2

```
\draw (x1,y1) rectangle (x1,y1);
```

```
\draw (x,y) ellipse (n and m);
```

en n va la medida de la altura y en m va la del ancho

```
\draw (x1,y1) parabola (x2,y1);
```

```
\draw (x1,y1) .. controls (x2,y2) and (x3,y3) .. (x4,y4);
```

Los dos extremos (acá (x1,y1) y (x4,y4)) representan el punto de salida y el de llegada.

Los controls sirven para atraer la línea hacia determinado punto para producir curvas.

Después del comando draw se pueden introducir algunos parámetros para personalizar la imagen:

color: grey, red...

Estilo de línea: dashed

Relleno: fill (se puede personalizar el color del relleno poniendo fill=grey/red/green...

```
\node (x,y) {texto};
```

Paquete tree-dvip

Para correrlo es necesario compilar usando `dvi<ps`

Entre otras cosas, nos permite agregar flechas. Para eso hay dos pasos:

Paso 1. Identificar los nodos que se quieren unir.

Paso 2. Dibujar la flecha entre esos nodos.

Para el paso 1, se usa el comando `\node{x}{y}`

x es el nombre que le vamos a poner al nodo e y es el texto que queremos que aparezca en ese nodo.

El paso 2 depende del tipo de flecha que se quiera dibujar

Para flecha combada: `\anodecurve[position-on]{source}[position-on]{target}{depth}`

Para flecha recta: `\anodeconnect[position-on]{source}[position-on]{target}`

Para flecha horizontal que una dos nodos en una línea abajo o arriba `\abarnodeconnect[distancia]{source}{target}`

Source y Target son los nombres que elegimos para los nodos. Position-on especifica la posición de la que parte o a la que llega la flecha. Puede tener los siguientes valores:

tl	t	tr
l	node	r
bl	b	br

Con depth se regula el la comba de la flecha. Probar con 0.1in

Si en lugar de anodecurve y anodeconnect se usa nodecurve y nodeconnect, solo se dibuja la línea (sin la flecha).

Paquete avm

Se lo invoca con el siguiente comando en el preámbulo

```
\usepackage{avm}
```

Es necesario tener el archivo avm.sty guardado en el path del archivo latex (si está en otro lado, especificar el path).

Las matrices con avm se enmarcan en el entorno avm.

```
\begin{avm}
```

```
\end{avm}
```

`\[\]` produce un corchete de apertura y uno de cierre

`\< \>` produce un corchete angular de apertura y uno de cierre

`\{ \}` produce una llave de apertura y una de cierre

`\\` sirve para cambiar de línea localmente (dentro de la “submatriz” en la que estoy)

`&` sirve para que se produzcan columnas alineadas