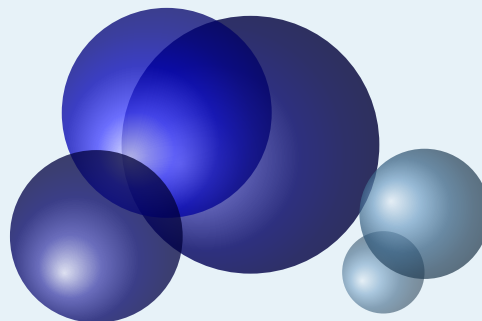
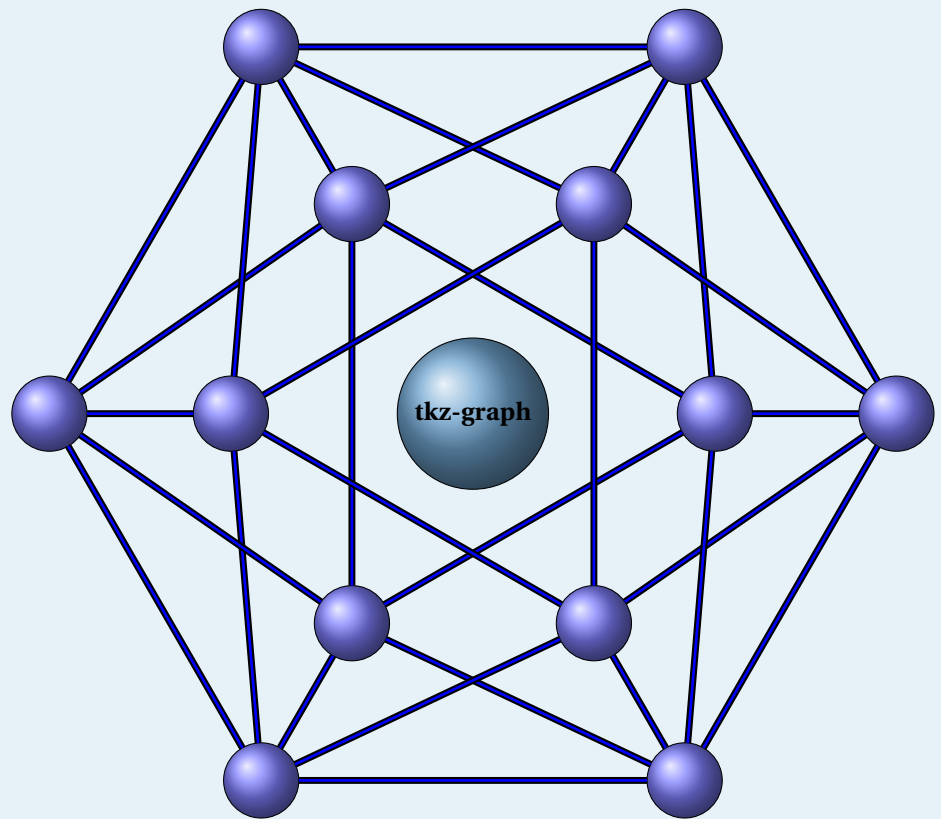


AlterMundus

tkz-graph 2.0c



Alain Matthes

20 janvier 2021 Documentation V.2.0c

<http://altermundus.fr>

tkz-graph

AlterMundus

Alain Matthes

Le package `tkz-graph.sty` est un package pour créer à l'aide de TikZ des graphes le plus simplement possible. Il fera partie d'une série de modules ayant comme point commun, la création de dessins utiles dans l'enseignement des mathématiques. La lecture de cette documentation va, je l'espère, vous permettre d'apprécier la simplicité d'utilisation de TikZ et vous permettre de commencer à le pratiquer. Il est accompagné du package `tkz-berge.sty` qui permet de tracer des graphes particuliers de la théorie des graphes.

☞ Je souhaite remercier **Till Tantau** pour avoir créé le merveilleux outil **TikZ**.

☞ Vous trouverez de nombreux exemples sur mon site : altermundus.fr

Vous pouvez envoyer vos remarques, et les rapports sur des erreurs que vous aurez constatées à l'adresse suivante : [Alain Matthes](mailto:Alain.Matthes@univ-lorraine.fr).

This file can be redistributed and/or modified under the terms of the LATEX Project Public License Distributed from CTAN archives in directory `CTAN://macros/latex/base/lppl.txt`.

Table des matières

1	Premiers graphes avec <code>tkz-graph.sty</code>	7
1.1	Exemple simple avec <code>tkz-graph</code>	7
1.2	Exemple classique avec <code>tkz-graph</code>	8
1.3	Modification du style	8
1.4	La ville de Königsberg avec <code>tkz-graph</code>	9
1.5	La ville de Königsberg avec TikZ mais sans <code>tkz-graph</code>	9
2	Vertex	11
2.1	<code>\Vertex</code>	11
2.1.1	Utilisation de coordonnées cartésiennes	11
2.1.2	Utilisation de coordonnées polaires	12
2.1.3	Option <code>Node</code> : utilisation d'une position référencée	12
2.2	Raccourcis pour placement relatif	13
2.2.1	Utilisation des raccourcis avec les valeurs par défaut	13
2.2.2	Modification de l'unité avec <code>\SetGraphUnit</code>	14
2.2.3	Modification des unités de TikZ : <code>x=2 cm,y=1 cm</code>	14
2.2.4	Exemple classique	14
2.2.5	Autre exemple classique	14
2.2.6	Modication locale de <code>unit</code> avec l'option	15
2.2.7	Modication locale de <code>unit</code> avec l'environnement <code>scope</code>	15
2.2.8	Modication locale de <code>unit</code> avec un groupe <code>TikZ</code>	15
3	Placement de sommets sur une forme géométrique	16
3.1	<code>\Vertices</code> à partir d'un sommet défini par des coordonnnées	16
3.2	<code>\Vertices</code> à partir d'une position donnée.	16
3.3	Exemples avec une direction	17
3.4	Placement sur un triangle	17
3.5	Utilisation d'un carré	17
3.6	Utilisation d'un cercle	18
3.7	Utilisation d'un cercle et positionnement des labels	18
3.8	Rotation et labels externes	18
3.9	Placement sur un cercle	18
4	Les labels	20
4.1	Options concernant les labels	20
4.1.1	Option <code>L</code>	20
4.1.2	Option <code>Math</code>	20
4.1.3	Suppression d'un label, Option <code>NoLabel</code>	20
4.1.4	Option <code>LabelOut</code> , <code>Lpos</code> et <code>Ldist</code>	20
4.2	<code>\SetVertexNoLabel</code>	21
4.2.1	Suppression des labels	21
4.3	<code>\SetVertexMath</code>	21
4.4	<code>\SetVertexLabel</code>	21
4.4.1	Labels supprimés puis autorisés.	21
4.4.2	Label en dehors du sommet <code>\SetVertexLabelOut</code>	22
5	Edge avec <code>tkz-graph</code>	23
5.1	Utilisation de <code>\Edge</code>	23
5.2	Arête particulière la boucle : <code>Loop</code>	24
5.2.1	Exemple avec <code>\Loop</code>	24
5.3	Multiple arêtes <code>\Edges</code>	25
5.3.1	Exemple avec <code>\Edges</code>	25

6	Modification des styles des sommets	26
6.1	Modification de <code>vstyle=Art</code>	29
6.2	Modification du style <code>VertexStyle</code> par défaut	30
6.3	Modification d'un style <code>VertexStyle</code>	30
6.4	Autre style <code>\SetVertexSimple</code>	31
6.5	<code>\SetVertexSimple</code> , <code>inner sep</code> et <code>outer sep</code>	31
6.6	Autre style <code>\SetVertexNormal</code>	32
6.7	<code>\SetUpVertex</code>	33
6.8	<code>\SetUpVertex</code> et <code>\tikzset</code>	33
7	Modification des styles des arêtes	34
7.1	Utilisation de l'option <code>style</code> de la macro <code>\Edge</code>	34
7.1.1	Exemple 1	34
7.1.2	Exemple 2	34
7.1.3	Exemple 3	34
7.2	Modification des styles par défaut <code>\SetUpEdge</code>	35
7.2.1	Utilisation de <code>\SetUpEdge</code> Exemple 1	35
7.2.2	Utilisation de <code>\SetUpEdge</code> Exemple 2	36
7.3	Arête avec label <code>LabelStyle</code>	36
7.4	Utiliser un style intermédiaire	36
8	Changement de couleurs dans les styles prédéfinis	38
8.1	<code>\SetGraphShadeColor</code>	38
8.1.1	Exemple	38
8.2	<code>\SetGraphArtColor</code>	39
8.2.1	Exemple	39
8.3	<code>\SetGraphColor</code>	40
8.3.1	Exemple avec <code>\SetGraphColor</code>	40
8.4	Variation I autour des styles	41
8.5	Variation II autour des styles	42
8.6	Variation III autour des styles	43
8.7	Variation IV autour des styles	44
8.8	Variation V autour des styles	45
9	Graphes probabilistes	46
9.1	La macro <code>\grProb</code>	46
9.2	Utilisation de <code>\grProb</code>	46
9.3	<code>\grProb</code> et le style par défaut	47
9.4	<code>\grProb</code> et le style « Simple »	47
9.5	Utilisation d'un style personnalisé	47
9.6	La macro <code>\grProbThree</code>	48
9.6.1	Graphe probabiliste d'ordre 3	48
10	Colorisation Welsh	49
10.1	La macro <code>\AddVertexColor</code>	49
10.2	Exemple d'utilisation	49
11	Annales.	53
11.1	Amérique du nord juin 2003	53
11.2	Antilles-Guyane juin 2003	54
11.3	Asie juin 2003	56
11.4	France juin 2003	58
11.5	Centres Étrangers juin 2003	60
11.6	Amérique du Nord juin 2004	63

11.7 Centres étrangers mai 2004	66
11.8 France juin 2004	68
11.9 La Réunion juin 2004	70
11.10Amérique du Sud Nov 2006	73
11.11Liban juin 2006	74
12 Dijkstra	77
12.1 Dijkstra exemple 1	77
12.2 Dijkstra exemple 2	78
Index	80

Liste des macros dans l'ordre d'apparition :

- `\SetVertexLabelOut`
- `\SetVertexLabelIn`
- `\SetVertexMath`
- `\SetVertexNoMath`
- `\SetUpVertex`
- `\Vertex`
- `\EA`
- `\WE`
- `\NO`
- `\SO`
- `\NOEA`
- `\NOWE`
- `\SOEA`
- `\SOWE`
- `\Vertices`
- `\SetUpEdge`
- `\Edge`
- `\Edges`
- `\Loop`
- `\grProb`
- `\SetGraphShadeColor`
- `\SetGraphArtColor`
- `\SetGraphColor`
- `\AddVertexColor`

1 Premiers graphes avec tkz-graph.sty

TikZ est un outil que je trouve très agréable à utiliser pour la création de graphes. J'ai trouvé si simple son utilisation que je me suis demandé si cela avait un sens de créer un package pour la création de graphes. Pas de théorie des graphes dans ce package, seulement des outils pour leur construction. Trois arguments peuvent intervenir pour soutenir mon effort :

1. Certains utilisateurs n'ont pas envie d'apprendre quoi que ce soit sur TikZ cela est respectable et une simplification du code par l'intermédiaire d'un package peut avoir une certaine utilité. La syntaxe n'est plus tout à fait celle de TikZ mais celle de \LaTeX .
2. Il est possible finalement de jouer avec les styles et d'optimiser certaines situations, ainsi la création d'un graphe sans la moindre coordonnée est possible. On peut obtenir des variantes du graphe, simplement en jouant avec les styles.
3. La création de ce que l'on peut appeler les graphes classiques de la théorie des graphes.
4. Et pour terminer, cela peut être une approche en douceur de l'utilisation de TikZ par l'intermédiaire des options.

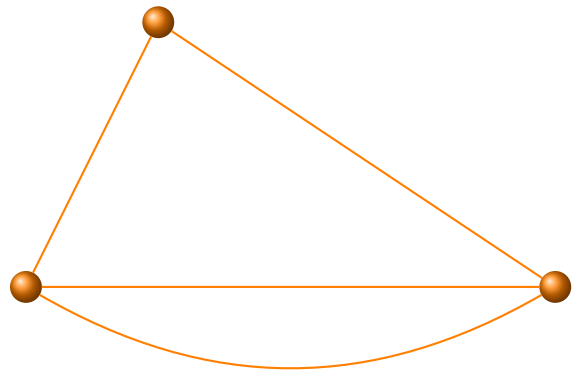
Que peut apporter `tkz-graph.sty`? Il facilite la gestion des styles des sommets et des arêtes, et également le positionnement de ceux-ci.

1.1 Exemple simple avec tkz-graph

Avant d'expliquer le fonctionnement des différentes macros, il est possible de tester si le package est bien installé avec l'exemple simple suivant. Le code complet est donné. Le préambule peut évidemment être modifié.

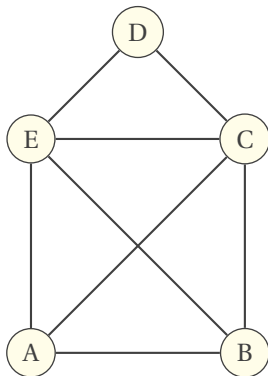
```
% Author : Alain Matthes
% Encoding : UTF8
% Engine : LuaLaTeX
\documentclass[border=3mm]{standalone}
\usepackage{tkz-graph}
\begin{document}

\begin{tikzpicture}[scale=1.75]
  \GraphInit[vstyle=Art]
  \Vertex{A}
  \Vertex[x=4,y=0]{B}
  \Vertex[x=1,y=2]{C}
  \Edge[style={bend left}](B)(A)
  \Edges(A,B,C,A)
\end{tikzpicture}
\end{document}
```



1.2 Exemple classique avec tkz-graph

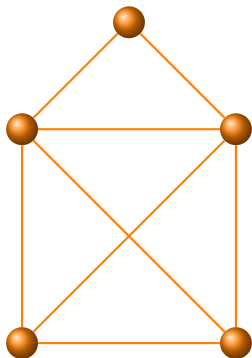
Voyons un exemple classique. Nous allons utiliser un style scolaire `vstyle=Normal` ainsi que les macros `\Vertices`, `\NOEA` et `\Edges` qui permet de créer une "chaîne" d'arêtes (edges). L'environnement `scope` fait partie de TikZ, il est utilisé ici afin d'appliquer une rotation.



```
\begin{tikzpicture}
  \GraphInit[vstyle=Normal]
  \SetGraphUnit{2}
  \begin{scope}[rotate=-135]
    \Vertices{circle}{A,B,C,E}
  \end{scope}
  \NOEA[unit=1.414](E){D}
  \Edges(A,B,E,D,C,E,A,C,B)
\end{tikzpicture}
```

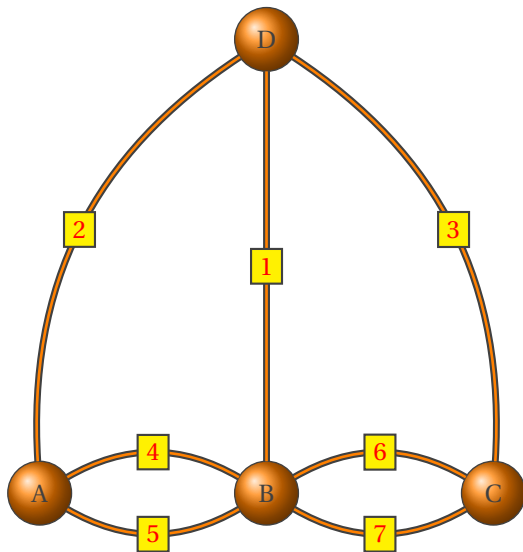
1.3 Modification du style

Un style plus esthétique peut être choisi avec `\GraphInit`. J'ai choisi `Art` parmi une liste que vous découvrirez plus tard.



```
\begin{tikzpicture}
  \GraphInit[vstyle=Art]
  \begin{scope}[rotate=-135]
    \Vertices[unit=2]{circle}{A,B,C,E}
  \end{scope}
  \NOEA[unit=1.414](E){D}
  \Edges(A,B,E,D,C,E,A,C,B)
\end{tikzpicture}
```


1.4 La ville de Königsberg avec tkz-graph



```

\begin{tikzpicture}
\SetGraphUnit{3}
\GraphInit[vstyle=Shade]
\tikzset{LabelStyle/.style= {draw,
                             fill = yellow,
                             text = red}}

\Vertex{A}
\EA(A){B}
\EA(B){C}
\SetGraphUnit{6}
% modifie la distance entre les nodes
\NO(B){D}
\Edge[label=1] (B) (D)
\tikzset{EdgeStyle/.append style = {bend left}}
\Edge[label=4] (A) (B)
\Edge[label=5] (B) (A)
\Edge[label=6] (B) (C)
\Edge[label=7] (C) (B)
\Edge[label=2] (A) (D)
\Edge[label=3] (D) (C)
\end{tikzpicture}

```

Ce dernier exemple était important sur un plan historique, mais il était un peu compliqué car on doit modifier des styles.

1.5 La ville de Königsberg avec TikZ mais sans tkz-graph

Voyons l'exemple précédent, sans l'utilisation du package `tkz-graph`. L'exemple peut être vu sur cet excellent site <http://www.texample.net/tikz/examples/bridges-of-konigsberg/>, voici le code complet. The result is on the next page. D'abord le préambule

```

% The seven bridges of Königsberg
% Author : Alain Matthes
% Encoding : UTF8
% Engine : PDFLaTeX
\documentclass[border=3mm]{standalone}
\usepackage{fullpage}
\usepackage{tikz}
\usetikzlibrary{arrows,shapes,positioning}
\begin{document}

```

Ensuite les styles principaux

```

\begin{center}
\begin{tikzpicture}
\useasboundingbox (-1,-1) rectangle (11,11);
\tikzset{VertexStyle/.style = {shape = circle,
                               ball color = orange,
                               text = black,
                               inner sep = 2pt,
                               outer sep = 0pt,
                               minimum size = 24 pt}}

\tikzset{EdgeStyle/.style = {thick,
                              double = orange,

```

```

double distance = 1pt}}
\tikzset{LabelStyle/.style = {draw,
                              fill      = yellow,
                              text      = red}}

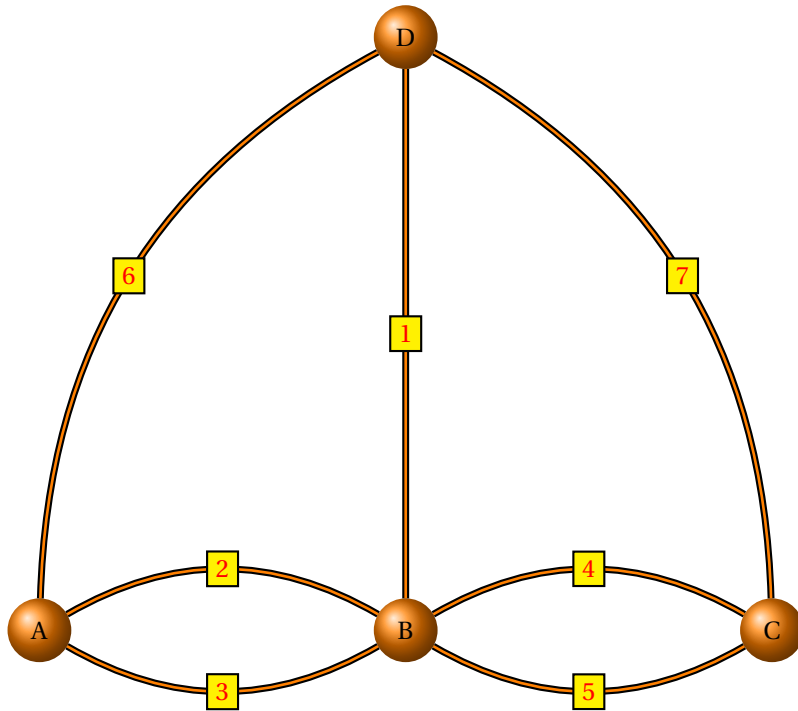
```

enfin, le tracé

```

\node[VertexStyle](A){A};
\node[VertexStyle,right=of A](B){B};
\node[VertexStyle,right=of B](C){C};
\node[VertexStyle,above= 7 cm of B](D){D};
\draw[EdgeStyle](B) to node[LabelStyle]{1} (D) ;
\tikzset{EdgeStyle/.append style = {bend left}}
\draw[EdgeStyle](A) to node[LabelStyle]{2} (B);
\draw[EdgeStyle](B) to node[LabelStyle]{3} (A);
\draw[EdgeStyle](B) to node[LabelStyle]{4} (C);
\draw[EdgeStyle](C) to node[LabelStyle]{5} (B);
\draw[EdgeStyle](A) to node[LabelStyle]{6} (D);
\draw[EdgeStyle](D) to node[LabelStyle]{7} (C);
\end{tikzpicture}
\end{center}
\end{document}

```



2 Vertex

C'est bien évidemment la macro essentielle qui permet de placer des sommets. Les sommets peuvent être placés avec un système de coordonnées rectangulaires ou bien polaires ou encore relativement les uns par rapport aux autres. Quelques dispositions particulières sont également possibles.

2.1 \Vertex

`\Vertex[⟨local options⟩]{⟨Name⟩}`

Un sommet se caractérise par :

- sa référence,
- sa position,
- son label,
- et le style.

Un argument non vide Name est obligatoire. Cet argument définit le nom de référence du node. C'est celui que l'on doit utiliser dans toute création de sommet (`\Vertex`) Il ne faut pas le confondre avec le `label` (étiquette) qui sera utilisé pour l'affichage. On peut vouloir afficher M_1 alors que le nom lui sera `M1`.

Des options sont utilisées pour définir les quatre premières caractéristiques. Les styles texte et graphique sont traités séparément.

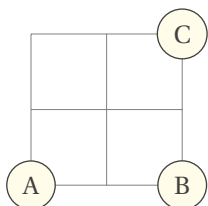
Options	Défaut	Définition
<code>x</code>	<code>{}</code>	abscisse
<code>y</code>	<code>{}</code>	ordonnée
<code>a</code>	<code>{}</code>	angle
<code>d</code>	<code>{}</code>	distance
<code>Node</code>	<code>false</code>	utilisation d'une référence déjà définie
<code>position</code>	<code>{}</code>	style qui permet un positionnement relatif
<code>dir</code>	<code>\EA</code>	direction pour un positionnement relatif
<code>empty</code>	<code>false</code>	booléen permettant de ne pas afficher le sommet
<code>NoLabel</code>	<code>false</code>	booléen supprime le label
<code>LabelOut</code>	<code>false</code>	booléen Label extérieur au node
<code>L</code>	<code>{}</code>	Le label
<code>Math</code>	<code>false</code>	booléen qui affiche le label en mode math
<code>Ldist</code>	<code>0cm</code>	distance du label au node
<code>Lpos</code>	<code>0</code>	position du label par rapport au node

Cette macro permet de définir un sommet qui a un nom `name` et un label.

Si `L=` alors `label = Name` sinon `label = L`.

2.1.1 Utilisation de coordonnées cartésiennes

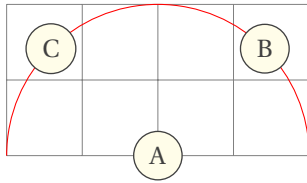
`\Vertex[x=⟨number⟩,y=⟨number⟩]{⟨name⟩}`. Coordonnées cartésiennes `x` et `y`.



```
\begin{tikzpicture}
  \GraphInit[vstyle=Normal]
  \draw[help lines] (0,0) grid (2,2);
  \Vertex{A} % par défaut x = 0 et y = 0
  \Vertex[x=2 , y=0]{B} \Vertex[x=2 , y=2]{C}
\end{tikzpicture}
```

2.1.2 Utilisation de coordonnées polaires

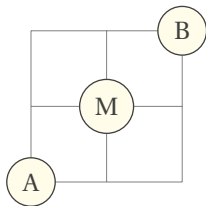
`\Vertex[a=(number),d=(number)]{<vertex>}` Les coordonnées polaires peuvent être aussi utilisées. J'ai utilisé une grille d'aide afin de constater le placement du sommet.



```
\begin{tikzpicture}
  \GraphInit[vstyle=Normal]
  \draw[help lines] (-2,0) grid (2,2);
  \draw[red] (2,0) arc (0:180: 2 cm);
  \Vertex{A}
  \Vertex[a=45 , d=2 cm]{B}
  \Vertex[a=135 , d=2 cm]{C}
\end{tikzpicture}
```

2.1.3 Option Node : utilisation d'une position référencée

Cette option permet de placer un sommet sur un Node déjà défini ou bien un objet du type « coordinate ».



```
\begin{tikzpicture}
  \GraphInit[vstyle=Normal]
  \draw[help lines] (0,0) grid (2,2);
  \Vertex{A} \Vertex[x=2 , y=2]{B}
  \coordinate (M) at ($ (A)!.5!(B) $){};
  \Vertex[Node]{M}
\end{tikzpicture}
```

2.2 Raccourcis pour placement relatif

Pour effectuer des placements relatifs, il est nécessaire de définir une distance unité entre deux sommets. La macro suivante permet de définir cette distance.

```
\SetGraphUnit{<nombre>}
```

Cette macro permet de définir la distance entre deux sommets. La distance se réfère aux centres de ces sommets et le nombre est exprimé en **cm**. Par défaut, l'unité est 1 cm.

utilisation : `\SetGraphUnit{2}`

```
\ShortCut[<local options>](<vertex A>){<vertex B>}
```

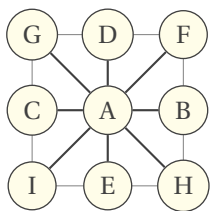
Ces raccourcis permettent de créer un **vertex B** relativement à un **vertex A**. La distance entre les deux sommets est déterminé par la valeur de **unit** et par les unités de TikZ. Horizontalement et verticalement la distance est définie par $\text{unit} \times x$ et $\text{unit} \times y$. La valeur de **unit** peut être redéfinie par la macro `\SetGraphUnit` ou bien avec l'option **unit**. Avec l'option la définition est locale; avec la macro, la définition est globale mais elle peut être locale si elle est interviert dans un groupe \TeX ou un environnement **scope**. Les raccourcis sont :

Raccourcis	Définition
<code>\EA</code>	à l'est
<code>\WE</code>	à l'ouest
<code>\NO</code>	au nord
<code>\SO</code>	au sud
<code>\NOEA</code>	au nord-est soit "nord" puis "est"
<code>\NOWE</code>	au nord-ouest soit "nord" puis "ouest"
<code>\SOEA</code>	au sud-est soit "sud" puis "est"
<code>\SOWE</code>	au sud-ouest soit "sud" puis "ouest"

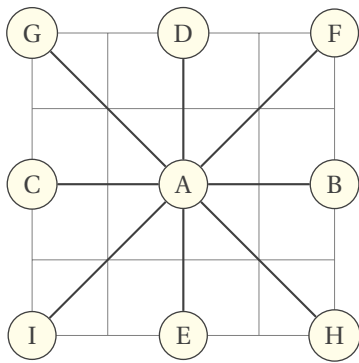
`\NOEA` est un raccourci pour `\NO\EA`. par défaut, la distance entre les sommets avec ce raccourci est $\sqrt{2} \times \text{unit} = \sqrt{2}$. Les options sont celles de la macro `\Vertex`.

Nous allons d'abord modifier la distance entre deux noeuds d'une façon générale avec `\SetGraphUnit{2}` sinon par défaut `unit = 1`.

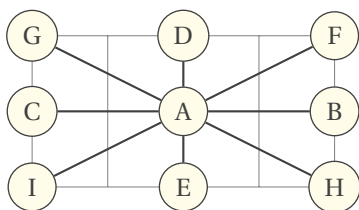
2.2.1 Utilisation des raccourcis avec les valeurs par défaut



```
\begin{tikzpicture}
  \draw[help lines] (-1,-1) grid (1,1);
  \GraphInit[vstyle=Normal]
  \Vertex{A}
  \EA(A){B} \WE(A){C} \NO(A){D} \SO(A){E}
  \NOEA(A){F} \NOWE(A){G} \SOEA(A){H} \SOWE(A){I}
  \foreach \v in {B,C,D,E,F,G,H,I}{\Edge(A)(\v)}
\end{tikzpicture}
```

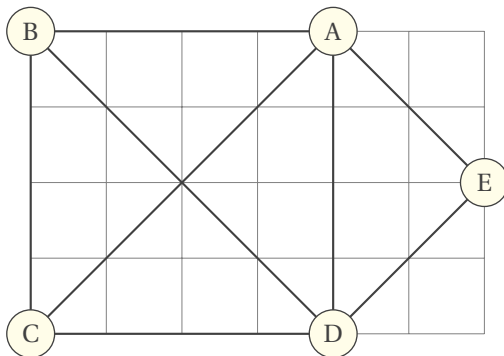
2.2.2 Modification de l'unité avec `\SetGraphUnit`

```
\begin{tikzpicture}
  \draw[help lines] (-2,-2) grid (2,2);
  \SetGraphUnit{2}
  \GraphInit[vstyle=Normal]
  \Vertex{A}
  \EA(A){B} \WE(A){C} \NO(A){D} \SO(A){E}
  \NOEA(A){F} \NOWE(A){G} \SOEA(A){H} \SOWE(A){I}
  \foreach \v in {B,C,D,E,F,G,H,I}{\Edge(A)(\v)}
\end{tikzpicture}
```

2.2.3 Modification des unités de TikZ : `x=2 cm,y=1 cm`

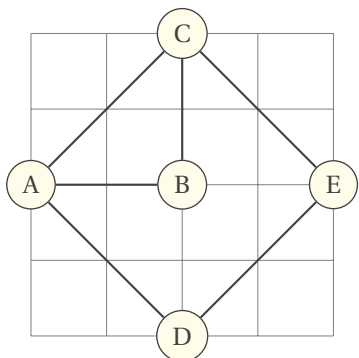
```
\begin{tikzpicture}[x=2 cm,y=1 cm]
  \draw[help lines] (-1,-1) grid (1,1);
  \GraphInit[vstyle=Normal]
  \Vertex{A}
  \EA(A){B} \WE(A){C} \NO(A){D} \SO(A){E}
  \NOEA(A){F} \NOWE(A){G} \SOEA(A){H} \SOWE(A){I}
  \foreach \v in {B,C,D,E,F,G,H,I}{\Edge(A)(\v)}
\end{tikzpicture}
```

2.2.4 Exemple classique



```
\begin{tikzpicture}
  \draw[help lines] (-2,-2) grid (4,2);
  \SetGraphUnit{2}
  \coordinate (O) at (0,0);
  \NOEA(O){A} \NOWE(O){B} \SOEA(O){D}
  \SOWE(O){C} \NOEA(D){E}
  \Edges(B,C,D,A,E,D,B,A,C)
\end{tikzpicture}
```

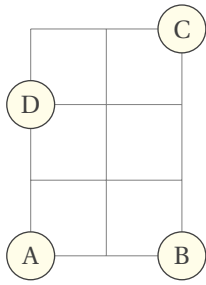
2.2.5 Autre exemple classique



```
\begin{tikzpicture}
  \draw[help lines] (0,-2) grid (4,2);
  \SetGraphUnit{2}
  \GraphInit[vstyle=Normal]
  \Vertex{A}
  \EA(A){B} \NO(B){C} \SO(B){D} \EA(B){E}
  \Edges(A,B,C,A,D,E,C)
\end{tikzpicture}
```

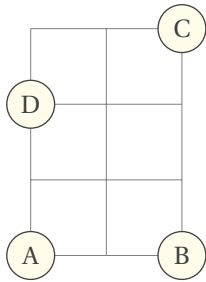
2.2.6 Modication locale de unit avec l'option

Le plus simple :



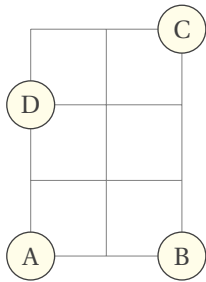
```
\begin{tikzpicture}
  \draw[help lines] (0,0) grid (2,3);
  \SetGraphUnit{2}
  \Vertex{A} \EA(A){B}
  \NO[unit=3](B){C}
  \NO(A){D}
\end{tikzpicture}
```

2.2.7 Modication locale de unit avec l'environnement scope



```
\begin{tikzpicture}
  \draw[help lines] (0,0) grid (2,3);
  \SetGraphUnit{2}
  \Vertex{A} \EA(A){B}
  \begin{scope}
    \SetGraphUnit{3} \NO(B){C}
  \end{scope}
  \NO(A){D}
\end{tikzpicture}
```

2.2.8 Modication locale de unit avec un groupe TeX



```
\begin{tikzpicture}
  \draw[help lines] (0,0) grid (2,3);
  \SetGraphUnit{2}
  \Vertex{A} \EA(A){B}
  {\SetGraphUnit{3} \NO(B){C}}
  \NO(A){D}
\end{tikzpicture}
```

3 Placement de sommets sur une forme géométrique

Il s'agit ici de placer un groupe de sommets suivant une direction donnée ou bien encore suivant une forme prédéfinie. Les sommets sont placés avec comme support une figure géométrique simple. La macro principale utilise une direction définie à l'aide de l'option `dir`, la version étoilée une forme particulière triangulaire, carrée etc...

`\Vertices[⟨local options⟩]{⟨type⟩}{⟨List of vertices⟩}`

Il y a donc plusieurs types de formes géométriques, droite, triangle, carrés et cercles. La macro `\SetGraphUnit` permet de modifier les longueurs. Pour les sommets alignés, ceux-ci sont placés suivant une direction donnée par `EA`, `WE`, `NO`, `SO`, `NOEA`, `NOWE`, `SOEA`, `SOWE`.

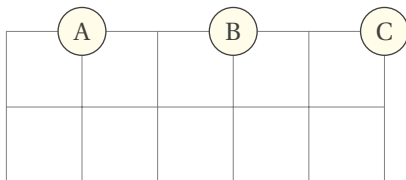
Premier Argument	Définition
<code>line</code>	Sommets alignés, une option détermine la direction
<code>tr1</code>	première forme de triangle
<code>tr2</code>	deuxième forme de triangle
<code>tr3</code>	troisième forme de triangle
<code>tr4</code>	quatrième forme de triangle
<code>square</code>	quatre sommets sur un carré
<code>circle</code>	sommets sur une cercle

Le second argument est une liste de noms pour les sommets.

Options	Défaut	Définition
<code>dir</code>	<code>EA</code>	permet de placer plusieurs sommets alignés

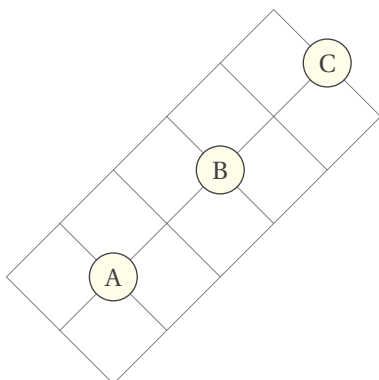
Les options sont celles d'un sommet (*Vertex*).

3.1 `\Vertices` à partir d'un sommet défini par des coordonnées



```
\begin{tikzpicture}
  \SetGraphUnit{2}
  \draw[help lines] (0,0) grid (5,2);
  \Vertices[x=1,y=2]{line}{A,B,C}
\end{tikzpicture}
```

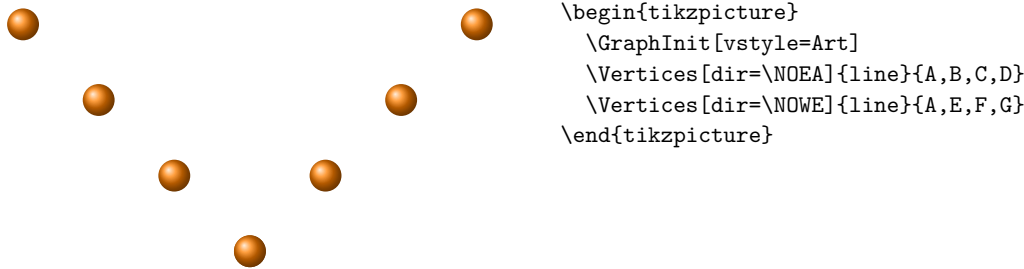
3.2 `\Vertices` à partir d'une position donnée.



```
\begin{tikzpicture}[rotate=45]
  \SetGraphUnit{2}
  \draw[help lines] (0,0) grid (5,2);
  \coordinate (A) at (1,1);
  \Vertices[Node]{line}{A,B,C}
\end{tikzpicture}
```

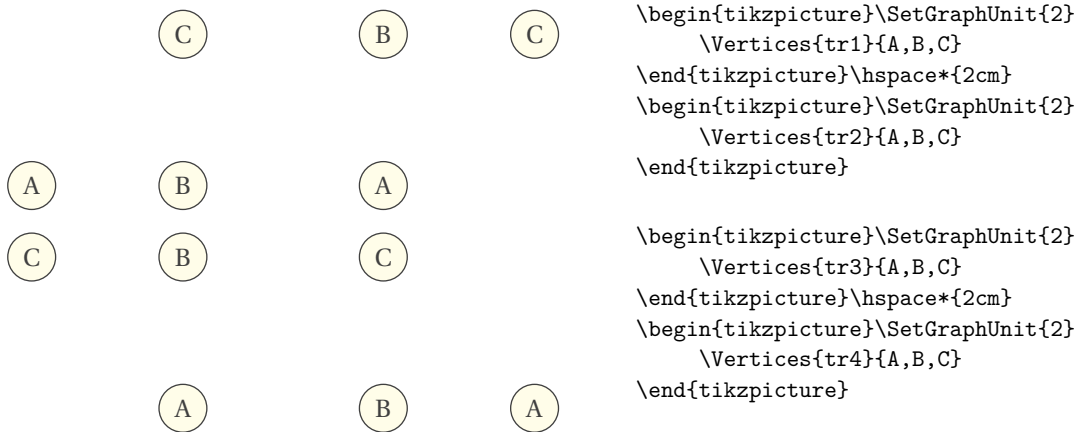

3.3 Exemples avec une direction

Il s'agit ici de placer une liste de sommets suivant une direction donnée, cette direction est définie à l'aide de l'option `dir`.



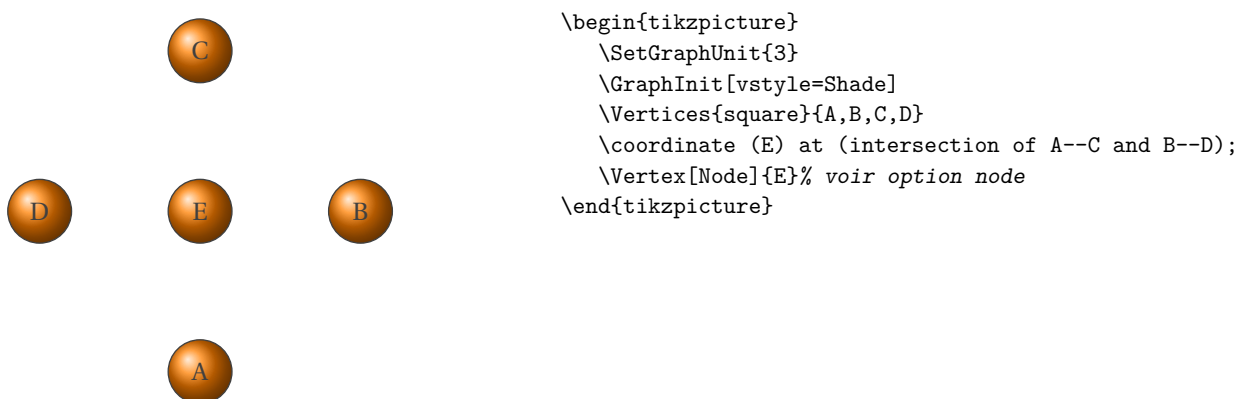
3.4 Placement sur un triangle

Il y a différentes possibilités avec une forme triangulaire, mais les triangles sont isocèles rectangles. Voici dans l'ordre les formes `tr1`, `tr2`, `tr3` et `tr4`



3.5 Utilisation d'un carré

Deux autres possibilités de placer un node. La première utilise un node obtenu à l'aide d'une intersection (voir le `pgfmanual`). Dans la première, j'ai redéfini la distance unité entre deux sommets à l'aide de `\SetGraphUnit`.



3.6 Utilisation d'un cercle



```
\begin{tikzpicture}
  \SetGraphUnit{2}
  \Vertices{circle}{A,B,C,D}
\end{tikzpicture}
```



3.7 Utilisation d'un cercle et positionnement des labels



```
\begin{tikzpicture} \SetGraphUnit{2}
  \GraphInit[vstyle=Classic]
  \Vertices{circle}{A,B,C,D,E,F}
\end{tikzpicture}
```



3.8 Rotation et labels externes

`Lpos = angle de la rotation`. Cela permet de faire une rotation du label autour du centre de chaque sommet et de suivre la rotation du graphe. Il suffit pour comprendre cette option de compiler l'exemple en l'omettant.

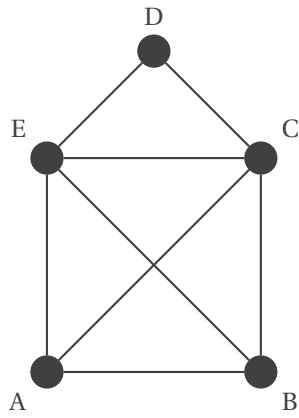


```
\begin{tikzpicture}[rotate=90]
  \GraphInit[vstyle=Classic]
  \Vertices[Lpos=90,unit=2]{circle}{A,B,C,D,E,F}
\end{tikzpicture}
```



3.9 Placement sur un cercle

Avec des labels externes, il faut procéder avec précaution.



```
\begin{tikzpicture}[scale=.5]
  \SetGraphUnit{4}
  \GraphInit[vstyle=Classic]
  \begin{scope}[rotate=45]
    \Vertices[Lpos=45]{circle}{C,E,A,B}
  \end{scope}
  \NOEA[Lpos=90,unit=2.828](E){D}
  \Edges(A,B,E,D,C,E,A,C,B)
\end{tikzpicture}
```

4 Les labels

Rappel : Si aucun label n'est donné alors l'affichage du label est celui de la référence du `vertex`. Il est possible de modifier localement le comportement des labels

4.1 Options concernant les labels

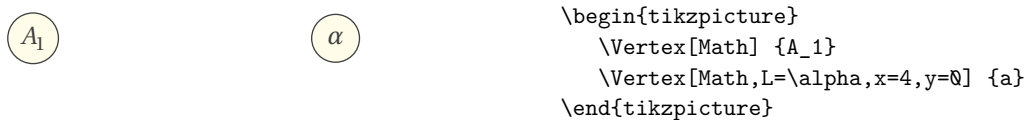
L'option suivante permet de définir un label, celui-ci peut être en mode texte ou bien en mode math.

4.1.1 Option L



4.1.2 Option Math

Le label est en mode math. Il est inutile de placer L en mode math si l'option est utilisée.



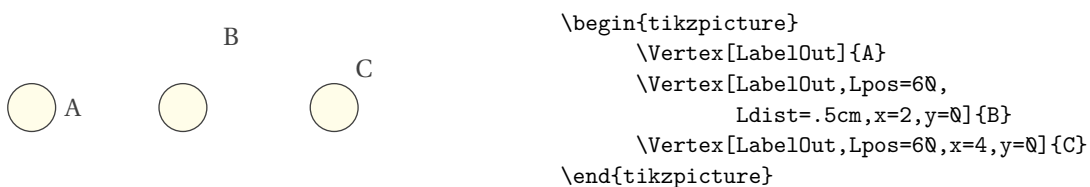
4.1.3 Suppression d'un label, Option NoLabel

Cette option supprime l'affichage du label. Il est préférable d'utiliser `SetVertexNoLabel` si on veut généraliser à tous les sommets.



4.1.4 Option LabelOut, Lpos et Ldist

La première option permet de placer le label hors du node, la deuxième positionne le label autour du sommet et la dernière spécifie la distance entre le label et le sommet.



On peut souhaiter appliquer une option pour tous les sommets.

4.2 `\SetVertexNoLabel`

On peut souhaiter ne pas avoir de label pour tous les sommets avec un style prédéfini.

`\SetVertexNoLabel`

Cette macro permet de supprimer les labels sur tous les sommets. Elle agit globalement sur tous les sommets. Elle correspond à l'option `NoLabel`.

4.2.1 Suppression des labels



```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \SetVertexNoLabel
  \Vertex{A}\EA(A){B}
\end{tikzpicture}
```

4.3 `\SetVertexMath`

`\SetVertexMath`

Cette macro permet d'appliquer l'option `Math` à plusieurs sommets. Elle agit globalement sur tous les sommets. Elle correspond à l'option `Math`.



```
\begin{tikzpicture}
  \SetVertexMath
  \Vertex {A_1} \EA[unit=3](A_1){A_2}\texttt{}
\end{tikzpicture}
```

4.4 `\SetVertexLabel`

`\SetVertexLabel`

Cette macro autorise les labels. Elle agit globalement sur tous les sommets.

4.4.1 Labels supprimés puis autorisés.

Dans l'exemple qui suit, les labels sont supprimés puis autorisés.



```
\begin{tikzpicture}
  \SetVertexNoLabel
  \SetGraphUnit{2}
  \Vertex {A} \EA(A){B}
  \SetVertexLabel \EA(B){C}
\end{tikzpicture}
```

4.4.2 Label en dehors du sommet `\SetVertexLabelOut``\SetVertexLabelOut`

`\SetVertexLabelOut` Dans les exemples précédents, les sommets sont des petits disques colorés, généralement en noir et dans ce cas par défaut le label est à l'extérieur. On peut contrôler la position à l'aide des labels avec `Ldist` et `Lpos`.

`\SetVertexLabelIn`

`\SetVertexLabelIn` permet d'écrire le label dans le sommet.

Cette macro permet d'appliquer l'option à plusieurs sommets. `\SetVertexLabelIn` annule l'effet.



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \SetVertexLabelOut
  \Vertex {A} \EA(A){B}
  \SetVertexLabelIn \SO[unit=3] (B){C}
\end{tikzpicture}
```



5 Edge avec tkz-graph

```
\Edge[⟨local options⟩](⟨Vertex A⟩)(⟨Vertex B⟩)
```

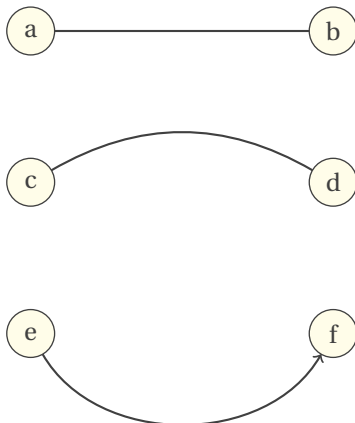
options	défaut	définition
local	false	booléen désactive EdgeStyle
color	\EdgeColor	couleur de l'arête
lw	\EdgeLineWidth	épaisseur de l'arête.
label	{}	le label
labeltext	black	couleur du texte
labelcolor	white	couleur du fond du label
labelstyle	{}	modification du style du label
style	pos=.5	modification du style général

Cette macro permet de tracer une arête entre deux sommets. Dans les exemples et dans le chapitre sur les styles, l'usage des styles est expliqué.

5.1 Utilisation de \Edge

On peut remarquer qu'il y a deux sortes d'arêtes au niveau de la forme : les segments et les arcs. De plus, ces arêtes peuvent avoir un label. La notion de style est importante car on peut définir pour toutes les arêtes un même style dès le début.

par défaut :



```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \Vertex{a}
  \EA(a){b}
  \SO[unit=2](a){c}
  \EA(c){d}
  {\SetGraphUnit{2}
  \SO(c){e}}
  \EA(e){f}
  \Edge(a)(b)
  \tikzset{EdgeStyle/.style = {-,bend left}}
  \Edge(c)(d)
  \tikzset{EdgeStyle/.style = {-
  >,bend right=60}}
  \Edge(e)(f)
\end{tikzpicture}
```

5.2 Arête particulière la boucle : Loop

```
\Loop[⟨local options⟩](⟨Vertex⟩)
```

options	défaut	définition
color	black	
lw	0.8pt	
label	{}	
labelstyle	{}	
style	{}	

5.2.1 Exemple avec \Loop



```
\begin{tikzpicture}
\useasboundingbox (-1,-2) rectangle (8,2);
\SetVertexSimple
\SetGraphUnit{5}
\Vertex{A}
\EA(A){B}
\Edge[style={->}] (A) (B)
\Loop[dist=3cm,dir=EA,style={thick,->}] (B)
\Loop[dist=5cm,dir=WE,style={thick,->}] (A)
\end{tikzpicture}
```

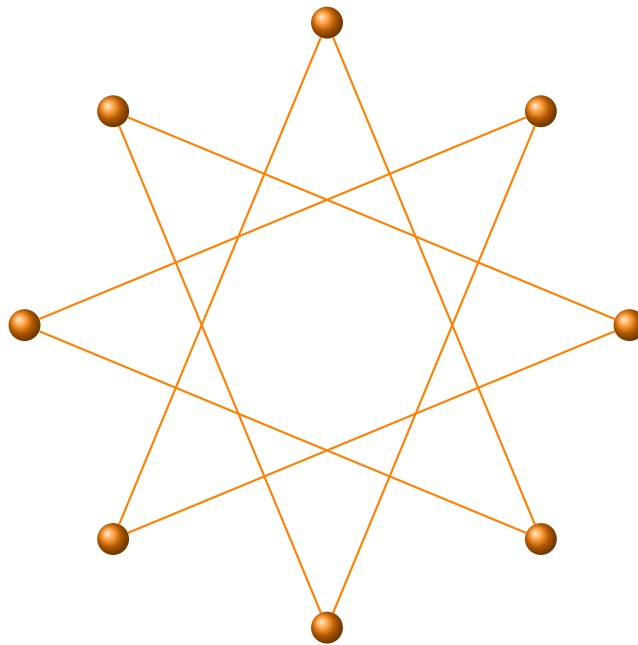

5.3 Multiple arêtes \Edges

```
\Edges[(local options)]((Vertex A,Vertex B,...))
```

options	défaut	définition
color	black	
lw	thick	
label	{}	
labelstyle	{}	
style	{}	

Cette macro permet de définir une série d'arêtes en une seule fois.

5.3.1 Exemple avec \Edges



```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \GraphInit[vstyle=Art]
  \Vertices{circle}{a0,a1,a2,a3,a4,a5,a6,a7}
  \Edges(a0,a3,a6,a1,a4,a7,a2,a5,a0)
\end{tikzpicture}
```

6 Modification des styles des sommets

Différentes méthodes sont possibles mais il faut distinguer une utilisation globale ou locale.

Les trois principaux styles sont `VertexStyle`, `EdgeStyle` et `LabelStyle`. Le dernier est attaché aux étiquettes que peuvent avoir les arêtes.

1. `\GraphInit` permet de choisir un style prédéfini et il est possible de retoucher ces styles en modifiant les valeurs choisies par défaut.
2. Les styles des sommets, des arêtes et étiquettes peuvent être personnalisés avec `VertexStyle`, `EdgeStyle` et `LabelStyle`. On peut redéfinir ces styles avec `\tikzset{VertexStyle/.append style = { ... }}` ou bien `\tikzset{VertexStyle/.style = { ... }}`. La première méthode modifie un style existant alors que la seconde définit un style.
3. On peut utiliser les anciennes macros : `\SetVertexSimple`, `\SetVertexNormal`, `\SetUpVertex` et `\SetUpEdge`.

Il est possible de mélanger tout cela en sachant que la dernière définition d'un style l'emporte.

<code>\GraphInit[(local options)]</code>		
Options	Défaut	Définition
<code>vstyle</code>	Normal	

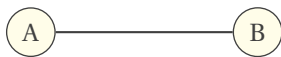
Les possibilités pour `vstyle` sont :

1. Empty,
2. Hasse,
3. Simple,
4. Classic,
5. Normal,
6. Shade,
7. Dijkstra
8. Welsh,
9. Art,
10. Shade Art.

Il y a pour le moment 10 styles pré-définis. Il est possible de modifier les valeurs par défaut.

Utilisation des styles pré-définis

1. `GraphInit` par défaut



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Normal]
  \Vertex{A}\EA(A){B}
  \Edge(A)(B)
\end{tikzpicture}
```

2. `GraphInit` et `vstyle=Empty`



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Empty]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

3. GraphInit et vstyle=Hasse



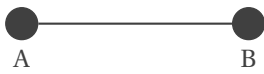
```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Hasse]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

4. GraphInit et vstyle=Simple



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Simple]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

5. GraphInit et vstyle=Classic

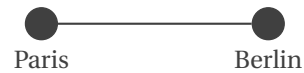


```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Classic]
  \Vertex[Lpos=-90]{A}
  \EA[Lpos=-90](A){B}\Edge(A)(B)
\end{tikzpicture}
```

6. GraphInit et vstyle=Normal



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Normal]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```



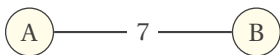
```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Classic]
  \Vertex[Lpos=-90]{Paris}
  \EA[Lpos=-90](Paris){Berlin}
  \Edge (Paris)(Berlin)
\end{tikzpicture}
```

7. GraphInit et vstyle=Shade



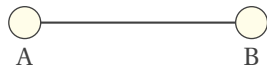
```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Shade]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

8. GraphInit et vstyle=Dijkstra



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Dijkstra]
  \Vertex{A}\EA(A){B}\Edge[label=7](A)(B)
\end{tikzpicture}
```

9. GraphInit et vstyle=Welsh



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Welsh]
  \Vertex[Lpos=-90]{A}
  \EA[Lpos=-90](A){B}\Edge(A)(B)
\end{tikzpicture}
```

10. GraphInit et vstyle=Art



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Art]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

11. GraphInit et vstyle=Shade Art

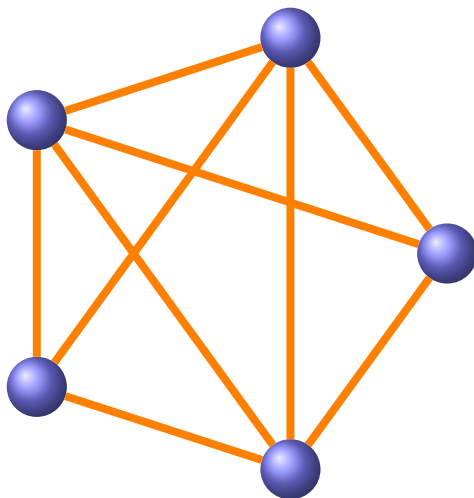


```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Shade Art]
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

vstyle est basé sur les macros suivantes qui peuvent être redéfinies.

Commandes pour les styles	utilisation
<code>\newcommand*{\VertexInnerSep}{0pt}</code>	
<code>\newcommand*{\VertexOuterSep}{0pt}</code>	
<code>\newcommand*{\VertexDistance}{3cm}</code>	
<code>\newcommand*{\VertexShape}{circle}</code>	
<code>\newcommand*{\VertexLineWidth}{0.8pt}</code>	
<code>\newcommand*{\VertexLineColor}{black}</code>	
<code>\newcommand*{\VertexLightFillColor}{white}</code>	
<code>\newcommand*{\VertexDarkFillColor}{black}</code>	
<code>\newcommand*{\VertexTextColor}{black}</code>	
<code>\newcommand*{\VertexFillColor}{black}</code>	
<code>\newcommand*{\VertexBallColor}{orange}</code>	
<code>\newcommand*{\VertexBigMinSize}{24pt}</code>	
<code>\newcommand*{\VertexInterMinSize}{18pt}</code>	
<code>\newcommand*{\VertexSmallMinSize}{12pt}</code>	
<code>\newcommand*{\EdgeFillColor}{orange}</code>	
<code>\newcommand*{\EdgeArtColor}{orange}</code>	
<code>\newcommand*{\EdgeColor}{black}</code>	
<code>\newcommand*{\EdgeDoubleDistance}{1pt}</code>	
<code>\newcommand*{\EdgeLineWidth}{0.8pt}</code>	

6.1 Modification de vstyle=Art



```

\begin{tikzpicture}
  \SetGraphUnit{3}
  \GraphInit[vstyle=Art]
  \renewcommand*{\VertexInnerSep}{8pt}
  \renewcommand*{\EdgeLineWidth}{3pt}
  \renewcommand*{\VertexBallColor}{blue!50}
  \Vertices{circle}{A,B,C,D,E}
  \Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}

```

6.2 Modification du style VertexStyle par défaut

Il est possible de redéfinir le style `\SetVertexSimple`.

Par défaut :

```
\tikzset{VertexStyle/.style = {
    shape      = circle,
    fill       = black,
    inner sep  = 0pt,
    outer sep  = 0pt,
    minimum size = 8pt,
    draw]
```

maintenant si on utilise ceci :



```
\begin{tikzpicture}
  \SetVertexSimple
  \tikzset{VertexStyle/.style = {
    shape      = rectangle,
    fill       = red,%
    inner sep  = 0pt,
    outer sep  = 0pt,
    minimum size = 10pt,
    draw}}
  \SetGraphUnit{3}
  \Vertex{A}\EA(A){B}
\end{tikzpicture}
```

6.3 Modification d'un style VertexStyle

C'est le style par défaut pour les sommets mais on peut le modifier. Voici quelques exemples utilisés plus tard dans ce document

par défaut :



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \tikzset{VertexStyle/.style = {%
    shape      = circle,
    shading    = ball,
    ball color = Orange,
    minimum size = 20pt,draw}}
  \SetVertexNoLabel
  \Vertex{A}\EA[unit=3](A){B}
\end{tikzpicture}
```

ou bien encore :



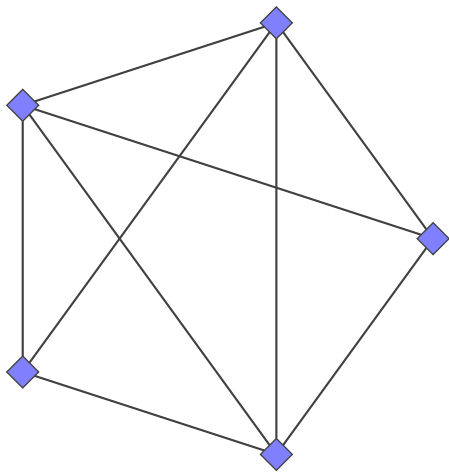
```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \tikzset{VertexStyle/.style = {%
    shape      = circle,
    shading    = ball,
    ball color = green!40!black,%
    minimum size = 30pt,draw}}
  \SetVertexNoLabel
  \Vertex{A}\EA[unit=3](A){B}
\end{tikzpicture}
```

```
\SetVertexSimple[local options]
```

Il est possible de modifier les styles prédéfinis. La macro `\SetVertexSimple` permet d'affiner le style « Simple » des

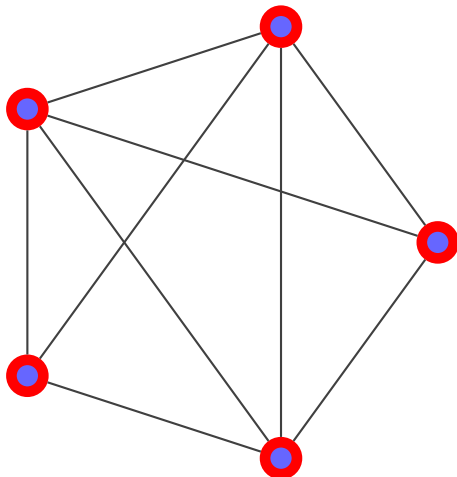
	options	default	definition
sommets.	Shape	<code>\VertexShape</code>	
	MinSize	<code>\VertexSmallMinSize</code>	
	LineWidth	<code>\VertexLineWidth</code>	
	LineColor	<code>\VertexLineColor</code>	
	FillColor	<code>\VertexFillColor</code>	

6.4 Autre style `\SetVertexSimple`



```
\begin{tikzpicture}
\SetVertexSimple[Shape=diamond,
                 FillColor=blue!50]
\Vertices[unit=3]{circle}{A,B,C,D,E}
\Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}
```

6.5 `\SetVertexSimple`, inner sep et outer sep



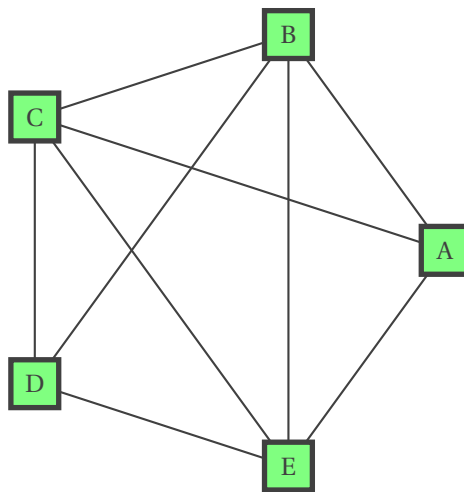
```
\begin{tikzpicture}
\SetGraphUnit{3}
\SetVertexSimple[MinSize    = 12pt,
                 LineWidth   = 4pt,
                 LineColor   = red,%
                 FillColor   = blue!60]
\tikzset{VertexStyle/.append style =
  {inner sep    = 0pt,%
   outer sep    = 2pt}}
\Vertices{circle}{A,B,C,D,E}
\Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}
```

```
\SetVertexNormal[⟨local options⟩]
```

Options	Défaut	Définition
color	\EdgeColor	
label	no default	
labelstyle	no default	
labeltext	\LabelTextColor	
labelcolor	\LabelFillColor	
style	no default	
lw	\EdgeLineWidth	

Macro semblable à la précédente.

6.6 Autre style \SetVertexNormal



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \SetVertexNormal[Shape = rectangle,%
                  LineWidth = 2pt,%
                  FillColor = green!50]
  \Vertices{circle}{A,B,C,D,E}
  \Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}
```

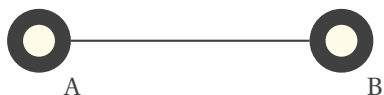

`\SetUpVertex[⟨local options⟩]`

Options	Défaut	Définition
Lpos	-90	position label externe
Ldist	0cm	distance du label
style		permet d'affiner le style
NoLabel	false	supprime le label
LabelOut	false	Label externe

Cette macro permet de modifier les options précédentes.

6.7 `\SetUpVertex`

```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \SetUpVertex[Lpos=-60,LabelOut]
  \Vertex{A}\EA(A){B}
\end{tikzpicture}
```

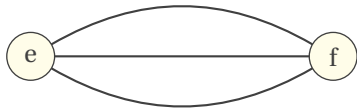
6.8 `\SetUpVertex` et `\tikzset`

```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \SetVertexLabel
  \SetUpVertex[Lpos=-60,LabelOut]
  \tikzset{VertexStyle/.append style =
    {outer sep = .5\pgflinewidth}}
  \renewcommand*\VertexLineWidth{6pt}
  \Vertex{A}\EA(A){B}\Edge(A)(B)
\end{tikzpicture}
```

7 Modification des styles des arêtes

7.1 Utilisation de l'option `style` de la macro `\Edge`

7.1.1 Exemple 1

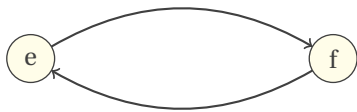


```

\begin{tikzpicture}
  \SetGraphUnit{4}
  \Vertex{e}
  \EA(e){f}
  \Edge(f)(e)
  \Edge[style={bend left}](f)(e)
  \Edge[style={bend right}](f)(e)
\end{tikzpicture}

```

7.1.2 Exemple 2

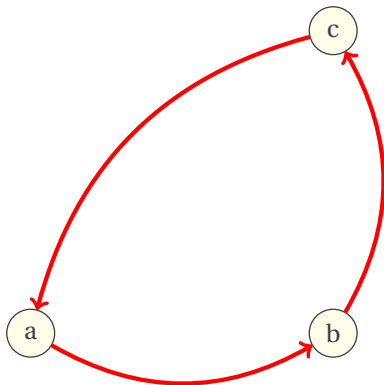


```

\begin{tikzpicture}
  \SetGraphUnit{4}
  \Vertex{e}
  \EA(e){f}
  \Edge[style={->,bend left}](f)(e)
  \Edge[style={<-,bend right}](f)(e)
\end{tikzpicture}

```

7.1.3 Exemple 3



```

\begin{tikzpicture}
  \SetGraphUnit{4}
  \Vertex{a}
  \EA(a){b}
  \NO(b){c}
  \SetUpEdge[style={->,bend right,ultra thick},
             color=red]
  \Edge(a)(b)
  \Edge(b)(c)
  \Edge(c)(a)
\end{tikzpicture}

```

7.2 Modification des styles par défaut `\SetUpEdge`

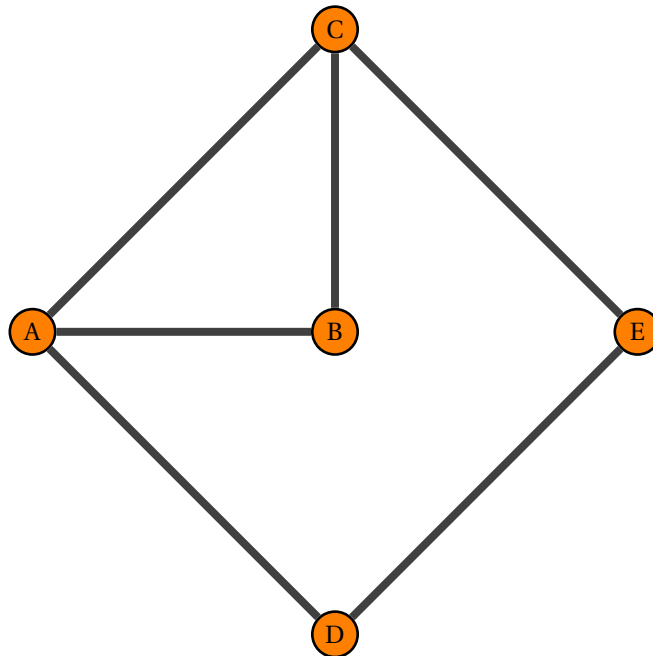
Cette macro a une action globale et permet de rédéfinir un style.

`\SetUpEdge[⟨local options⟩]`

Options	Défaut	Définition
<code>lw</code>	<code>-90</code>	position label externe
<code>color</code>	<code>\EdgeLineWidth</code>	position label externe
<code>label</code>	<code>0cm</code>	distance du label
<code>labelstyle</code>		permet d'affiner le style
<code>labeltext</code>	<code>false</code>	supprime le label
<code>style</code>	<code>false</code>	Label externe

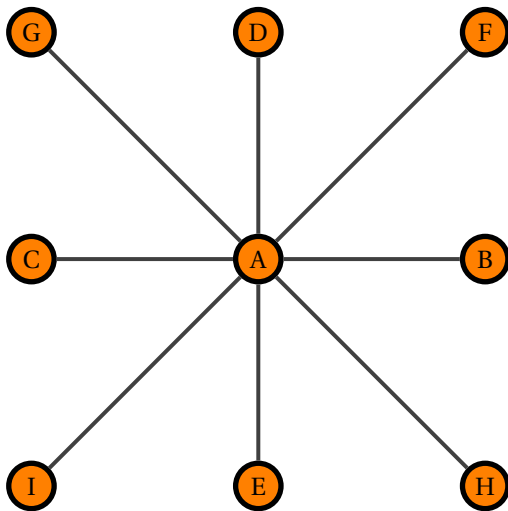
Cette macro permet de modifier les options précédentes.

7.2.1 Utilisation de `\SetUpEdge` Exemple 1



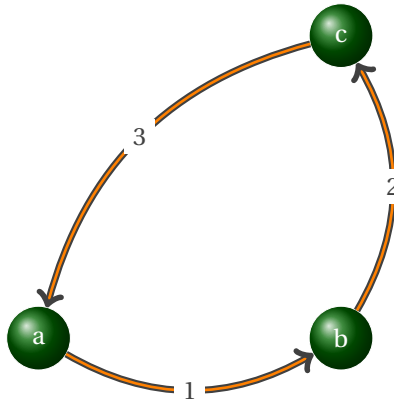
```
\begin{tikzpicture}
  \SetGraphUnit{4}    \SetUpEdge[lw=3pt]
  \Vertex{A}
  \EA (A){B}    \NO (B){C}
  \SO (B){D}    \EA (B){E}
  \Edges(A,B,C,A,D,E,C)
\end{tikzpicture}
```

7.2.2 Utilisation de \SetUpEdge Exemple 2



```
\begin{tikzpicture}
  \SetGraphUnit{3}
  \SetUpEdge[lw=1.5pt]
  \Vertex{A}
  \EA(A){B} \WE(A){C} \NO(A){D}
  \SO(A){E} \NOEA(A){F} \NOWE(A){G}
  \SOEA(A){H} \SOWE(A){I}
  \foreach \v in {B,C,D,E,F,G,H,I}{%
    \Edge(A)(\v)}
\end{tikzpicture}
```

7.3 Arête avec label LabelStyle



```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \tikzset{VertexStyle/.style =
    {draw,
      shape = circle,
      shading = ball,
      ball color = green!40!black,
      minimum size = 24pt,
      color = white}}
  \tikzset{EdgeStyle/.style =
    {->,bend right,
      thick,
      double = orange,
      double distance = 1pt}}
  \Vertex{a}
  \EA(a){b}
  \NO(b){c}
  \tikzset{LabelStyle/.style =
    {fill=white}}
  \Edge[label=$1$](a)(b)
  \Edge[label=$2$](b)(c)
  \Edge[label=$3$](c)(a)
\end{tikzpicture}
```

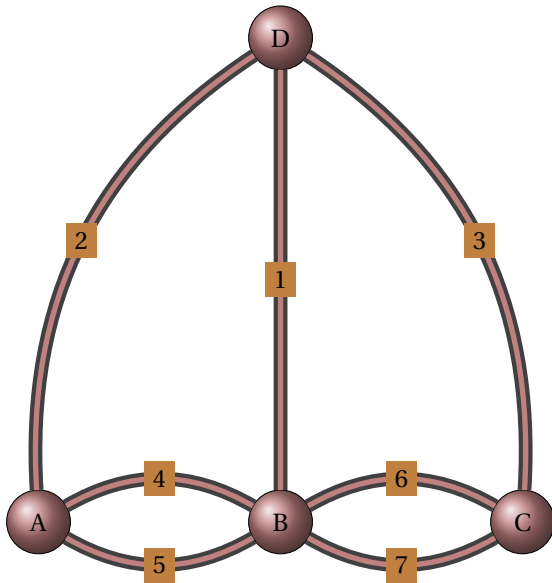
7.4 Utiliser un style intermédiaire

```
\SetGraphUnit{4}
\tikzset{VertexStyle/.style = {shape = circle,
  shading = ball,
  ball color = Maroon!50,
  minimum size = 24pt,
  draw}}
\tikzset{TempEdgeStyle/.style = {ultra thick,
```

```

double           = Maroon!50,
double distance = 2pt}}
\tikzset{LabelStyle/.style = {color           = brown,
                              text=black}}

```



```

\begin{tikzpicture}[scale=.8]
  \Vertex{A}
  \EA(A){B} \EA(B){C}
  \SetGraphUnit{8}
  \NO(B){D}
  \tikzset{EdgeStyle/.style = {TempEdgeStyle}}
  \Edge[label=1](B)(D)
  \tikzset{EdgeStyle/.style = {TempEdgeStyle,bend left}}
  \Edge[label=4](A)(B) \Edge[label=5](B)(A)
  \Edge[label=6](B)(C) \Edge[label=7](C)(B)
  \Edge[label=2](A)(D) \Edge[label=3](D)(C)
\end{tikzpicture}

```

8 Changement de couleurs dans les styles prédéfinis

Trois macros sont proposées

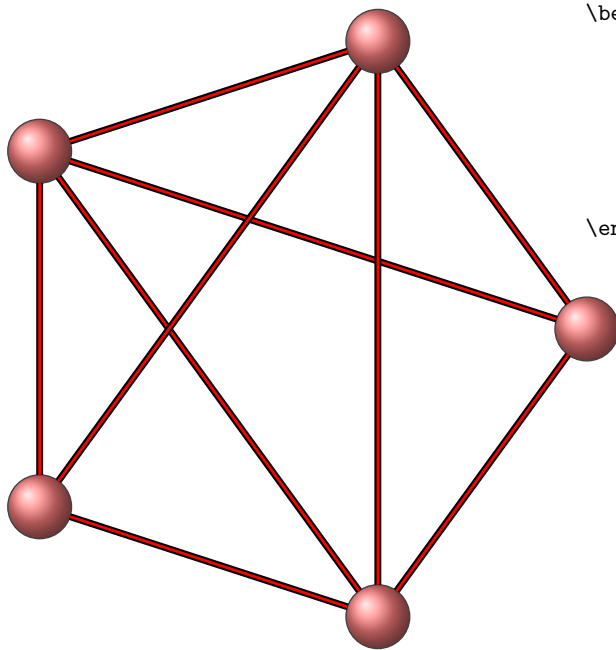
8.1 `\SetGraphShadeColor`

```
\SetGraphShadeColor{<ball color>}{<color>}{<double>}
```

`\SetGraphShadeColor` permet de modifier les couleurs pour le style `Shade`.

8.1.1 Exemple

Cet exemple utilise une macro de `tkz-berge`



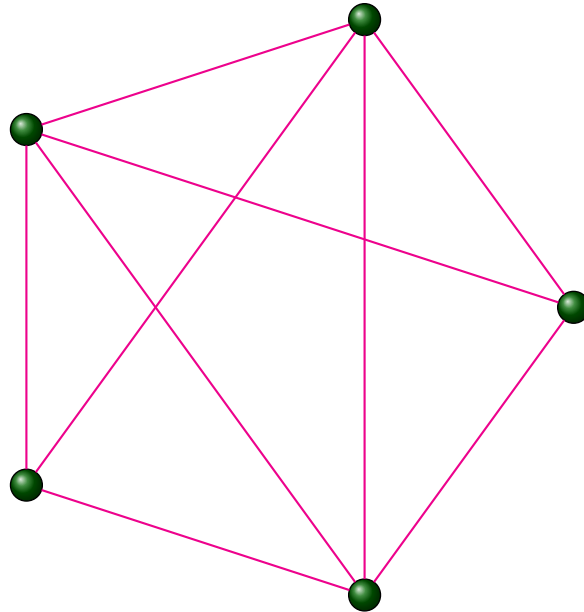
```
\begin{tikzpicture}
  \GraphInit[vstyle=Shade]
  \SetGraphUnit{4}
  \SetVertexNoLabel
  \SetGraphShadeColor{red!50}{black}{red}
  \Vertices{circle}{A,B,C,D,E}
  \Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}
```

8.2 `\SetGraphArtColor`

```
\SetGraphArtColor{<ball color>}{<color>}
```

`\SetGraphArtColor` permet de modifier les couleurs pour le style `Art`.

8.2.1 Exemple

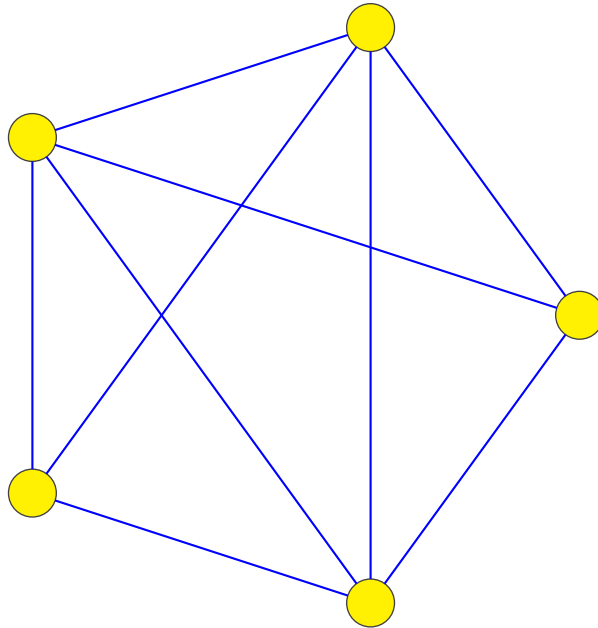


```
\begin{tikzpicture}
  \SetVertexArt
  \SetGraphArtColor{green!40!black}{magenta}
  \SetGraphUnit{4}
  \SetVertexNoLabel
  \Vertices{circle}{A,B,C,D,E}
  \Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}
```

8.3 `\SetGraphColor`

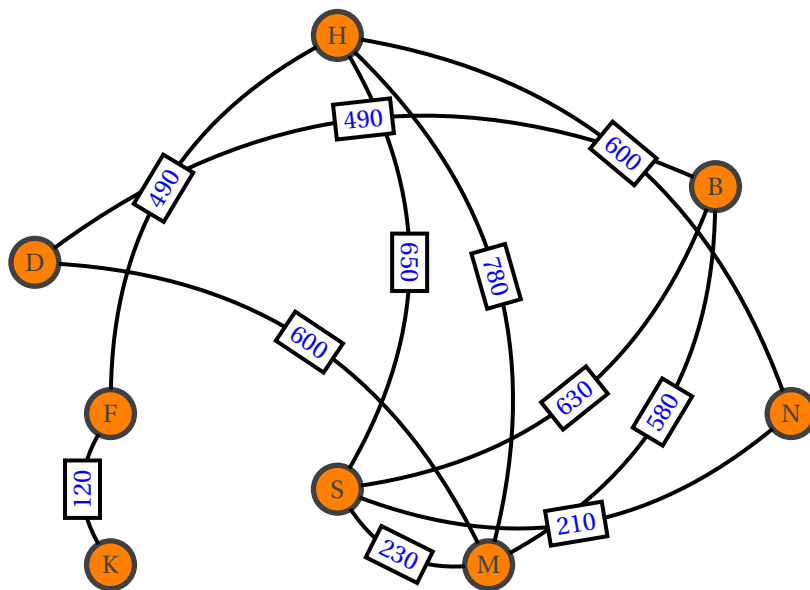
```
\SetGraphColor{<fill color>}{<color>}
```

`\SetGraphColor` permet de modifier les couleurs pour le style `Normal`.

8.3.1 Exemple avec `\SetGraphColor`

```
\begin{tikzpicture}
  \SetGraphColor{yellow}{blue}
  \SetGraphUnit{4}
  \SetVertexNoLabel
  \Vertices{circle}{A,B,C,D,E}
  \Edges(A,B,C,D,E,A,C,E,B,D)
\end{tikzpicture}
```


8.4 Variation I autour des styles

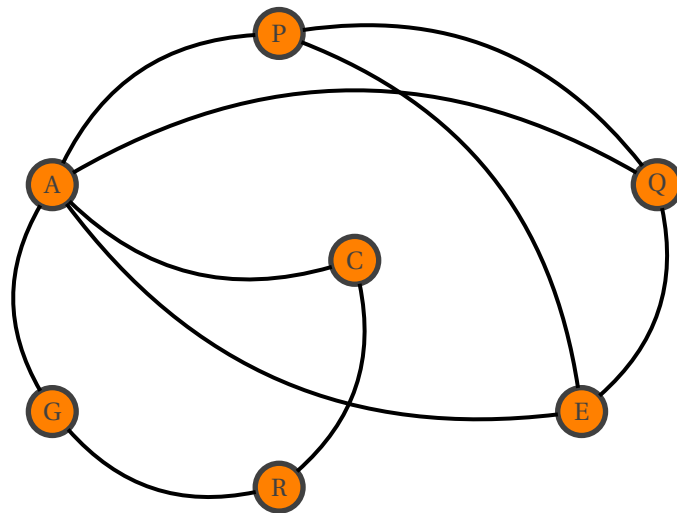


```

\begin{tikzpicture}
  \SetVertexNormal[Shape      = circle,
                  FillColor  = orange,
                  LineWidth  = 2pt]
  \SetUpEdge[lw      = 1.5pt,
            color    = black,
            labelcolor = white,
            labeltext = red,
            labelstyle = {sloped,draw,text=blue}]
  \Vertex[x=0 ,y=0]{K}
  \Vertex[x=0 ,y=2]{F}
  \Vertex[x=-1,y=4]{D}
  \Vertex[x=3 ,y=7]{H}
  \Vertex[x=8 ,y=5]{B}
  \Vertex[x=9 ,y=2]{N}
  \Vertex[x=5 ,y=0]{M}
  \Vertex[x=3 ,y=1]{S}
  \tikzset{EdgeStyle/.append style = {bend left}}
  \Edge[label = $120$](K)(F)
  \Edge[label = $650$](H)(S)
  \Edge[label = $780$](H)(M)
  \Edge[label = $490$](D)(B)
  \Edge[label = $600$](D)(M)
  \Edge[label = $580$](B)(M)
  \Edge[label = $600$](H)(N)
  \Edge[label = $490$](F)(H)
  \tikzset{EdgeStyle/.append style = {bend right}}
  \Edge[label = $630$](S)(B)
  \Edge[label = $210$](S)(N)
  \Edge[label = $230$](S)(M)
\end{tikzpicture}

```

8.5 Variation II autour des styles

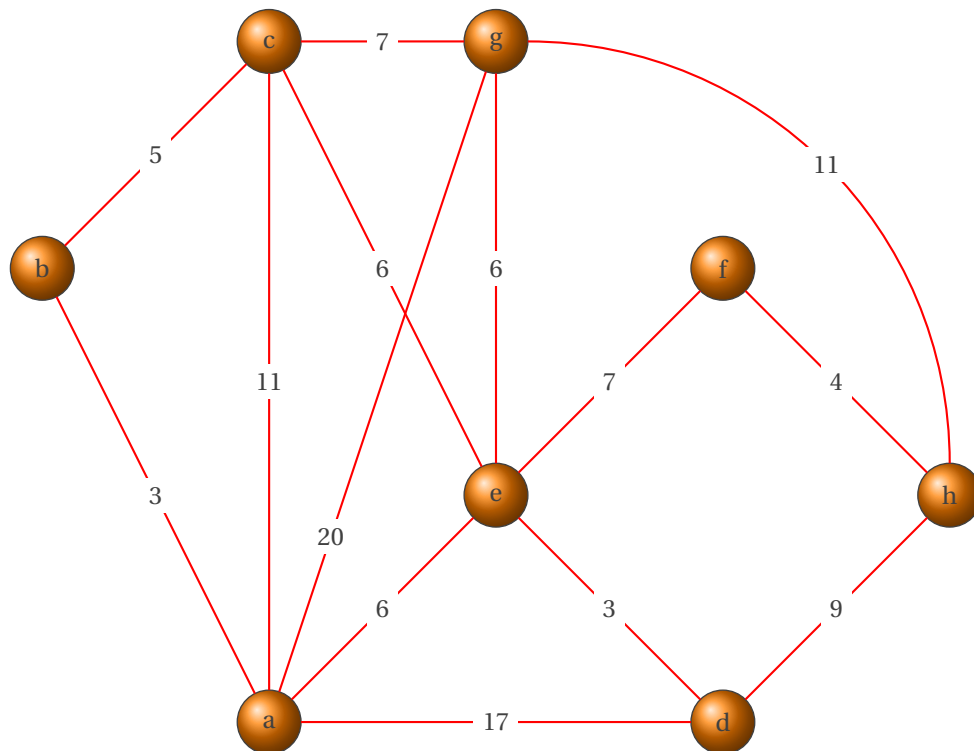


```

\begin{tikzpicture}
  \SetVertexNormal[Shape      = circle,
                  FillColor  = orange,
                  LineWidth  = 2pt]
  \SetUpEdge[lw      = 1.5pt,
             color    = black,
             labelcolor = white,
             labeltext = red,
             labelstyle = {sloped,draw,text=blue}]
  \tikzstyle{EdgeStyle}=[bend left]
  \Vertex[x=0, y=0]{G}
  \Vertex[x=0, y=3]{A}
  \Vertex[x=3, y=5]{P}
  \Vertex[x=4, y=2]{C}
  \Vertex[x=8, y=3]{Q}
  \Vertex[x=7, y=0]{E}
  \Vertex[x=3, y=-1]{R}
  \Edges(G,A,P,Q,E) \Edges(C,A,Q) \Edges(C,R,G) \Edges(P,E,A)
\end{tikzpicture}

```

8.6 Variation III autour des styles

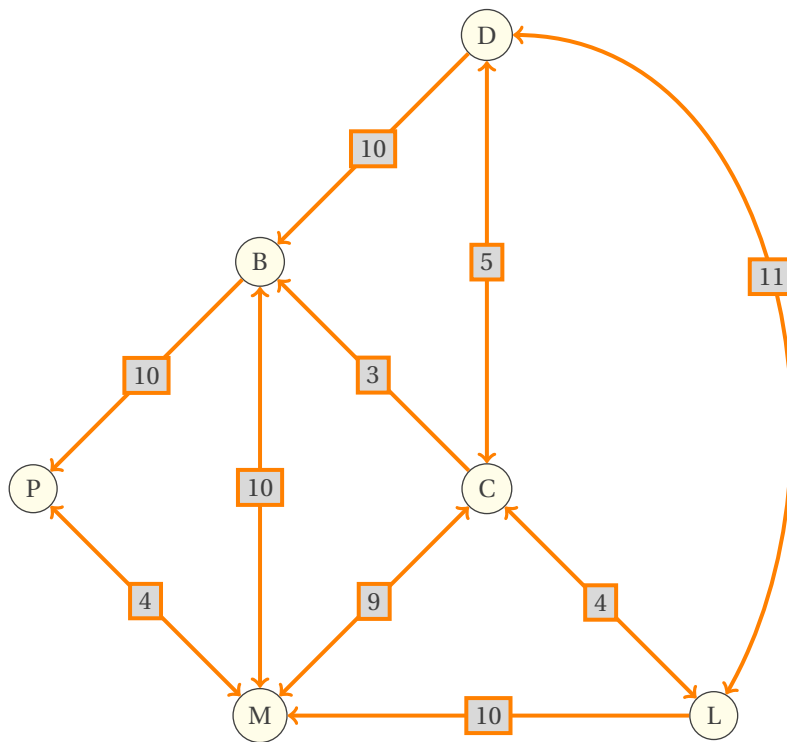


```

\begin{tikzpicture}
  \GraphInit[vstyle=Shade]
  \SetGraphUnit{3}
  \Vertex{e}
  \NOEA(e){f}\SOEA(e){d}
  \SOEA(f){h}\NOWE(f){g}
  \WE(g){c} \SOWE(e){a} \SOWE(c){b}
  \tikzstyle{LabelStyle}=[fill=white]
  \tikzstyle{EdgeStyle}=[color=red]
  \Edge[label=$3$](a)(b)
  \Edge[label=$11$](a)(c)
  \Edge[label=$6$](a)(e)
  \Edge[label=$17$](a)(d)
  \Edge[style={pos=.25},label=$20$](a)(g)
  \Edge[label=$5$](c)(b)
  \Edge[label=$6$](c)(e)
  \Edge[label=$7$](c)(g)
  \Edge[label=$7$](f)(e)
  \Edge[label=$3$](d)(e)
  \Edge[label=$9$](d)(h)
  \Edge[label=$6$](g)(e)
  \Edge[style={bend left,out=45,in=135},label=$11$](g)(h)
  \Edge[label=$4$](f)(h)
\end{tikzpicture}

```

8.7 Variation IV autour des styles

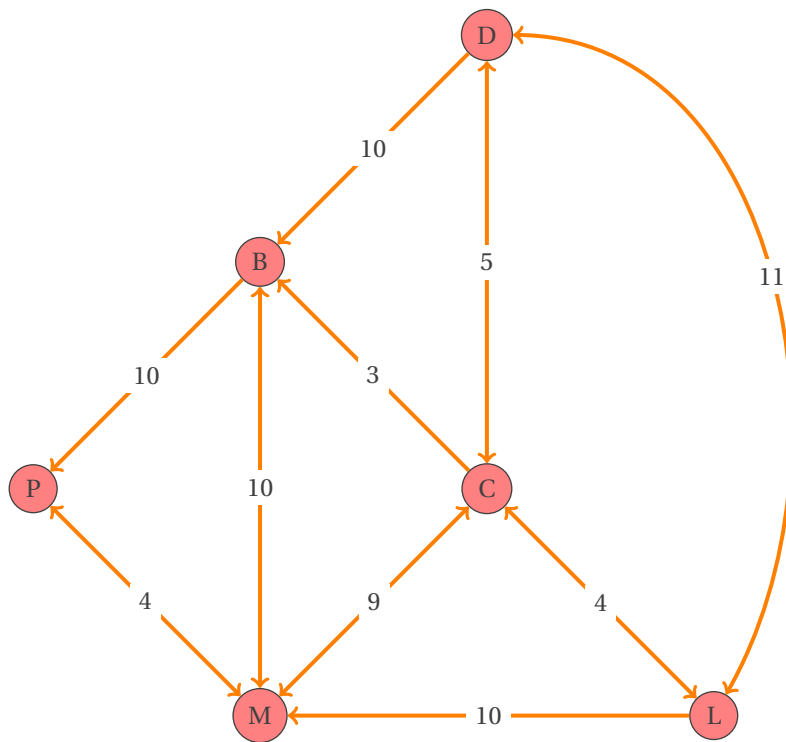


```

\begin{tikzpicture}
  \SetUpEdge[lw      = 1.5pt,
             color    = orange,
             labelcolor = gray!30,
             labelstyle = {draw}]
  \SetGraphUnit{3}
  \GraphInit[vstyle=Normal]
  \Vertex{P}
  \NOEA(P){B}
  \SOEA(P){M}
  \NOEA(B){D}
  \SOEA(B){C}
  \SOEA(C){L}
  \tikzset{EdgeStyle/.style={->}}
  \Edge[label=$3$](C)(B)
  \Edge[label=$10$](D)(B)
  \Edge[label=$10$](L)(M)
  \Edge[label=$10$](B)(P)
  \tikzset{EdgeStyle/.style={<->}}
  \Edge[label=$4$](P)(M)
  \Edge[label=$9$](C)(M)
  \Edge[label=$4$](C)(L)
  \Edge[label=$5$](C)(D)
  \Edge[label=$10$](B)(M)
  \tikzset{EdgeStyle/.style={<->,relative=false,in=0,out=60}}
  \Edge[label=$11$](L)(D)
\end{tikzpicture}

```

8.8 Variation V autour des styles



```

\begin{tikzpicture}

\SetUpEdge[lw      = 1.5pt,
           color    = orange,
           labelcolor = white]
\GraphInit[vstyle=Normal] \SetGraphUnit{3}
\tikzset{VertexStyle/.append style={fill = red!50}}
\Vertex{P}
\NOEA(P){B} \SOEA(P){M} \NOEA(B){D}
\SOEA(B){C} \SOEA(C){L}
\tikzset{EdgeStyle/.style={->}}
\Edge[label=$3$](C)(B)
\Edge[label=$10$](D)(B)
\Edge[label=$10$](L)(M)
\Edge[label=$10$](B)(P)
\tikzset{EdgeStyle/.style={<->}}
\Edge[label=$4$](P)(M)
\Edge[label=$9$](C)(M)
\Edge[label=$4$](C)(L)
\Edge[label=$5$](C)(D)
\Edge[label=$10$](B)(M)
\tikzset{EdgeStyle/.style={<->,relative=false,in=0,out=60}}
\Edge[label=$11$](L)(D)
\end{tikzpicture}

```

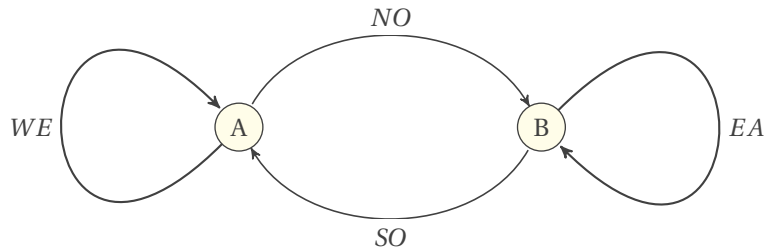
9 Graphes probabilistes

9.1 La macro `\grProb`

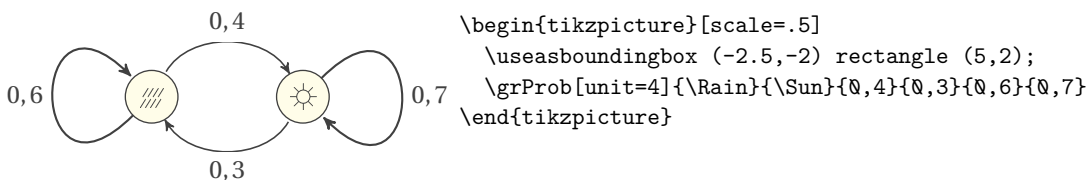
```
\grProb[⟨local options⟩] {⟨left⟩} {⟨right⟩} {⟨N⟩}{⟨S⟩}{⟨W⟩}{⟨E⟩}
```

Arguments	Définition	
Vertex-left	Nom du sommet à gauche	
Vertex-right	Nom du sommet à droite	
label N	Étiquette située en haut	
label S	Étiquette située en bas	
label W	Étiquette située à gauche	
label E	Étiquette située à droite	
options	défaut	définition
unit	4cm	distance entre les sommets
LposA	180	angle si label extérieur en A
LposB	0	angle si label extérieur en B
Ldist	0cm	écart entre le node et le label
LoopDist	4cm	longueur des boucles

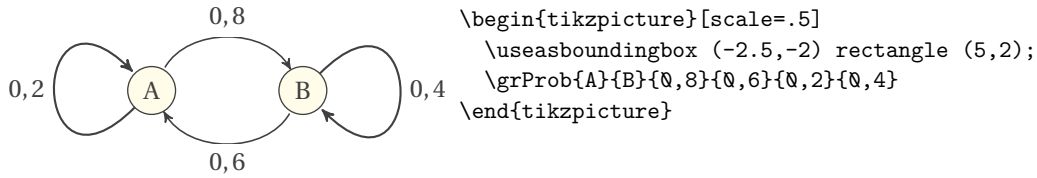
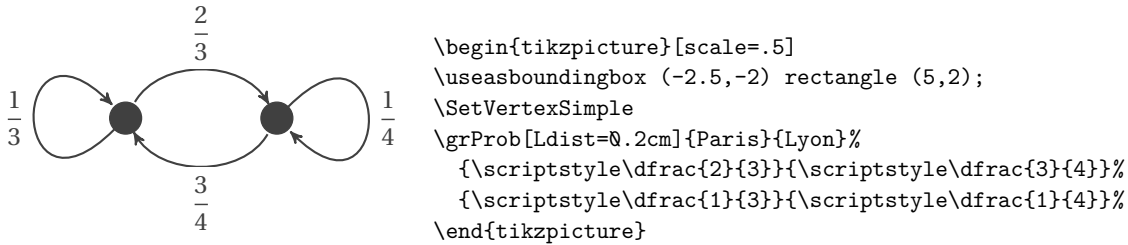
Cette macro permet de créer un graphe probabiliste d'ordre 2.

9.2 Utilisation de `\grProb`

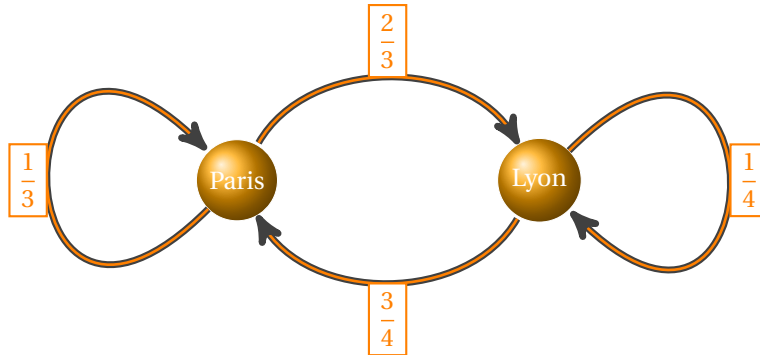
```
\begin{tikzpicture}
  \useasboundingbox (-2.5,-2) rectangle (7.5,2);
  \grProb{A}{B}{NO}{SO}{WE}{EA}
\end{tikzpicture}
```



```
\begin{tikzpicture}[scale=.5]
  \useasboundingbox (-2.5,-2) rectangle (5,2);
  \grProb[unit=4]{\Rain}{\Sun}{0,4}{0,3}{0,6}{0,7}
\end{tikzpicture}
```

9.3 `\grProb` et le style par défaut9.4 `\grProb` et le style « Simple »

9.5 Utilisation d'un style personnalisé



```

\begin{tikzpicture}
  \useasboundingbox (-2.5,-2.5) rectangle (7.5,2.5);
  \tikzset{VertexStyle/.style = {shape = circle,
    shading = ball,
    ball color = Orange,
    minimum size = 20pt,
    draw,color=white}}
  \tikzset{LabelStyle/.style = {draw,color=orange,fill=white}}
  \tikzset{EdgeStyle/.style = {->, thick,
    double = orange,
    double distance = 1pt}}

  \grProb[Ldist=0.1cm,LposA=0,LposB=180]%
    {Paris}{Lyon}%
    {\scriptstyle\frac{2}{3}}{\scriptstyle\frac{3}{4}}%
    {\scriptstyle\frac{1}{3}}{\scriptstyle\frac{1}{4}}%
\end{tikzpicture}

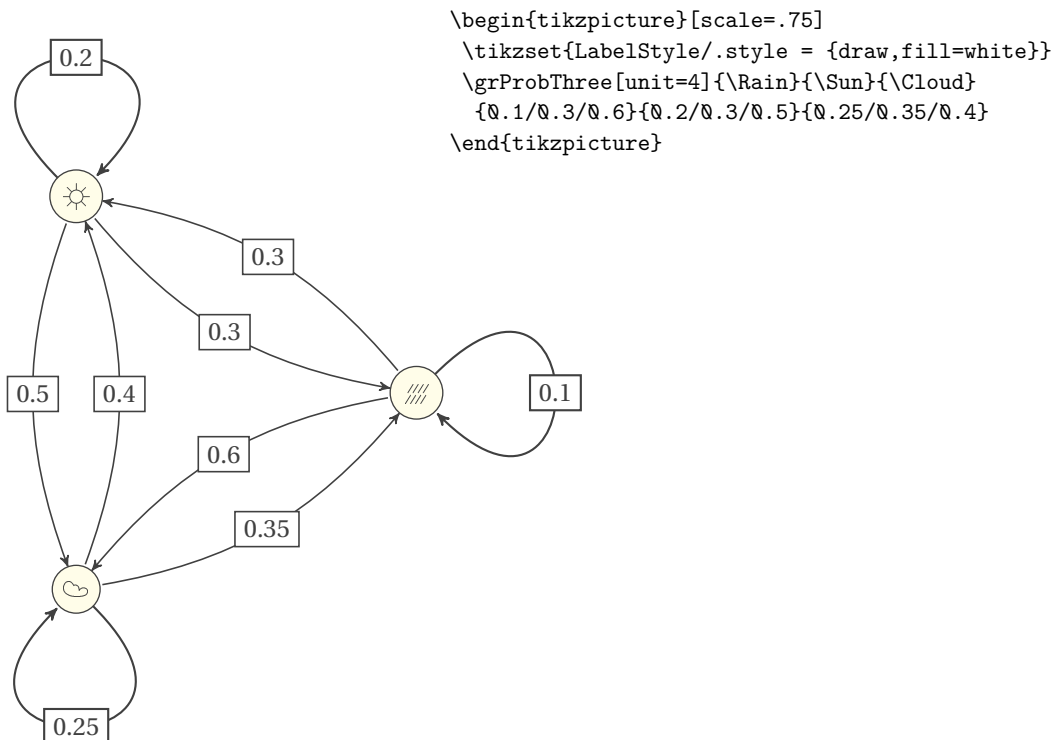
```

9.6 La macro `\grProbThree`

<code>\grProbThree[⟨local options⟩] {⟨right⟩} {⟨up⟩}{⟨down⟩} {⟨rr/ru/rd⟩}{⟨uu/ud/ur⟩}{⟨dd/dr/du⟩}</code>		
Arguments	Définition	
Vertex-right	Nom du sommet à droite	
Vertex-up	Nom du sommet en haut	
Vertex-down	Nom du sommet en bas	
rr/ru/rd	arête partant de r vers r etc...	
uu/ud/ur	arête partant de u vers u etc...	
dd/dr/du	arête partant de d vers d etc...	
Options	Défaut	Définition
unit	4cm	distance entre les sommets
LposA	180	angle si label extérieur en A
LposB	0	angle si label extérieur en B
Ldist	0cm	écart entre le node et le label
LoopDist	4cm	longueur des boucles

Cette macro permet de créer un graphe probabiliste d'ordre 3.

9.6.1 Graphe probabiliste d'ordre 3



10 Colorisation Welsh

Ce chapitre montre comment colorer des sommets. Le plus simple est d'utiliser le style `Welsh` et la macro `\AddVertexColor` afin de colorer les sommets.

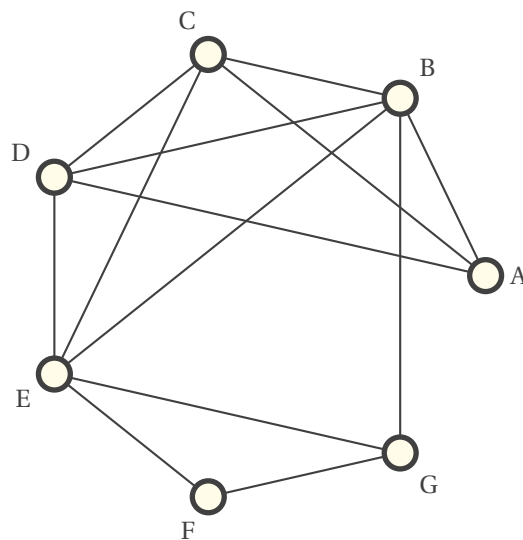
10.1 La macro `\AddVertexColor`

```
\AddVertexColor{<color>}{<List of vertices>}
```

Cette macro permet de colorer des sommets. Le premier argument est la couleur, le second une liste de sommets.

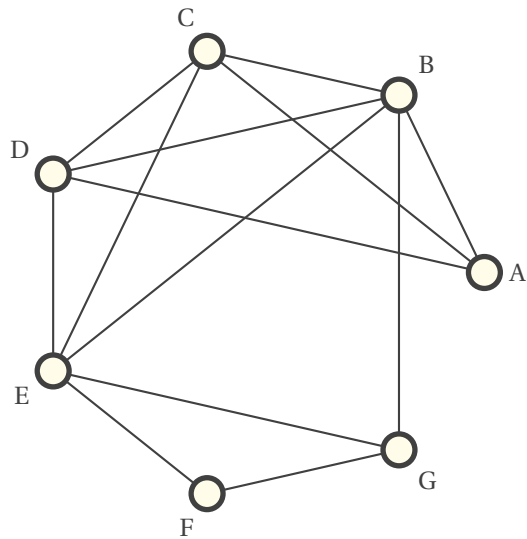
10.2 Exemple d'utilisation

Une compagnie aérienne propose des vols directs entre certaines villes, notées A, B, C, D, E, F et G. Cela conduit au graphe \mathcal{G} suivant, dont les sommets sont les villes et les arêtes représentent les liaisons aériennes :



1. Sur les cartes d'embarquement, la compagnie attribue à chaque aéroport une couleur, de sorte que deux aéroports liés par un vol direct aient des couleurs différentes. Proposer un coloriage adapté, cette condition.
2. Que peut-on en déduire sur le nombre chromatique de \mathcal{G} ?

```
\begin{tikzpicture}
\renewcommand*{\VertexLineWidth}{2pt}
\GraphInit[vstyle=Welsh]
\Vertices[unit=3]{circle}{A,B,C,D,E,F,G}
\Edges(G,E,F,G,B,D,E,C,D,A,C,B,A) \Edges(B,E)
\end{tikzpicture}
```

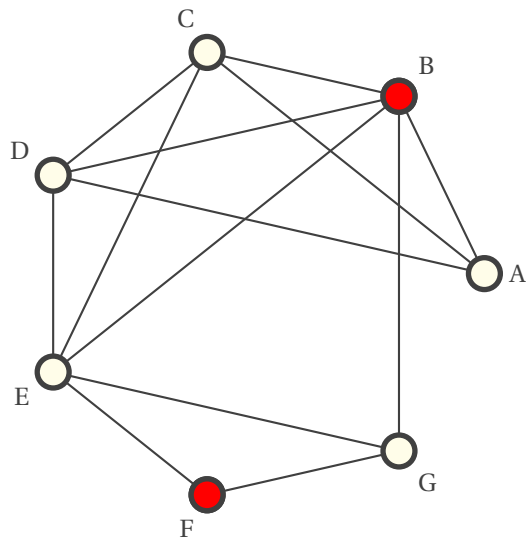


<i>Sommet</i>	<i>Degré</i>
B	5
E	5
C	4
D	4
A	3
G	3
F	2

```

\begin{tikzpicture}
\renewcommand*{\VertexLineWidth}{2pt}
\GraphInit[vstyle=Welsh]
\Vertices[unit=3]{circle}{A,B,C,D,E,F,G}
\Edges(G,E,F,G,B,D,E,C,D,A,C,B,A) \Edges(B,E)
\end{tikzpicture}

```

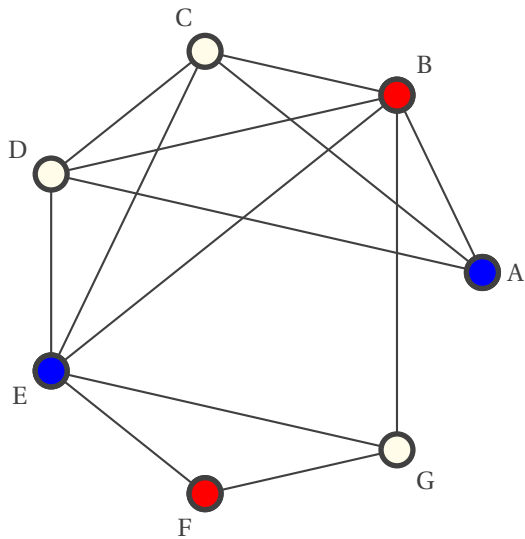


<i>Sommet</i>	<i>Degré</i>	<i>Couleur</i>
B	5	rouge
E	5	
C	4	
D	4	
A	3	
G	3	
F	2	rouge

```

\begin{tikzpicture}
\renewcommand*{\VertexLineWidth}{2pt}
\GraphInit[vstyle=Welsh]
\Vertices[unit=3]{circle}{A,B,C,D,E,F,G}
\SetVertexNoLabel
\AddVertexColor{red}{B,F}
\Edges(G,E,F,G,B,D,E,C,D,A,C,B,A) \Edges(B,E)
\end{tikzpicture}

```

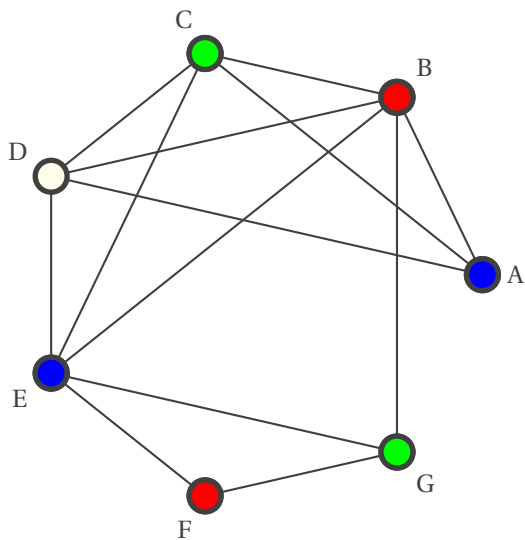


<i>Sommet</i>	<i>Degré</i>	<i>Couleur</i>
B	5	rouge
E	5	bleu
C	4	
D	4	
A	3	bleu
G	3	
F	2	rouge

```

\begin{tikzpicture}
  \renewcommand*{\VertexLineWidth}{2pt}
  \GraphInit[vstyle=Welsh]
  \Vertices[unit=3]{circle}{A,B,C,D,E,F,G}
  \SetVertexNoLabel
  \AddVertexColor{red}{B,F}
  \AddVertexColor{blue}{E,A}
  \Edges(G,E,F,G,B,D,E,C,D,A,C,B,A)
  \Edges(B,E)
\end{tikzpicture}

```



<i>Sommet</i>	<i>Degré</i>	<i>Couleur</i>
B	5	rouge
E	5	bleu
C	4	vert
D	4	
A	3	bleu
G	3	vert
F	2	rouge

```

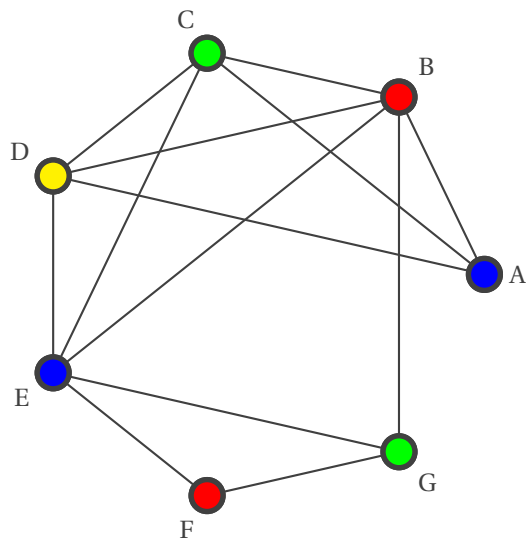
\begin{tikzpicture}
  \renewcommand*{\VertexLineWidth}{2pt}
  \GraphInit[vstyle=Welsh]
  \Vertices[unit=3]{circle}{A,B,C,D,E,F,G}
  \SetVertexNoLabel
  \AddVertexColor{red}{B,F} \AddVertexColor{blue}{E,A}

```

```

\AddVertexColor{green}{C,G}
\Vertex[Node]{D}
\Edges(G,E,F,G,B,D,E,C,D,A,C,B,A) \Edges(B,E)
\end{tikzpicture}

```



<i>Sommet</i>	<i>Degré</i>	<i>Couleur</i>
B	5	rouge
E	5	bleu
C	4	vert
D	4	jaune
A	3	bleu
G	3	vert
F	2	rouge

```

\begin{tikzpicture}
\renewcommand*{\VertexLineWidth}{2pt}
\GraphInit[vstyle=Welsh]
\Vertices[unit=3]{circle}{A,B,C,D,E,F,G}
\SetVertexNoLabel
\AddVertexColor{red}{B,F} \AddVertexColor{blue}{E,A}
\AddVertexColor{green}{C,G}\AddVertexColor{yellow}{D}
\Vertex[Node]{D}
\Edges(G,E,F,G,B,D,E,C,D,A,C,B,A)\Edges(B,E)
\end{tikzpicture}

```

11 Annales.

11.1 Amérique du nord juin 2003

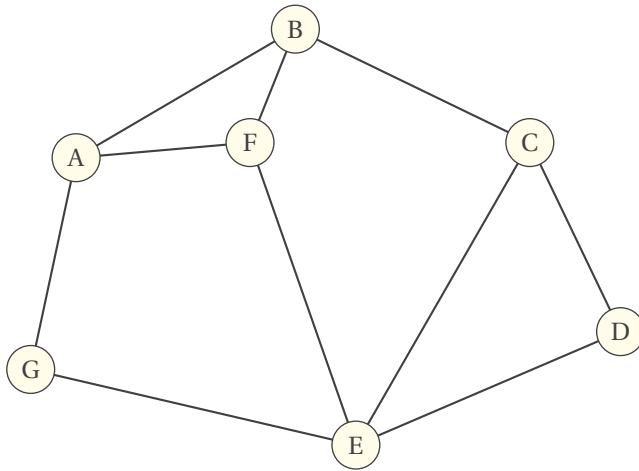
Soit le graphe G joint en annexe constitué des sommets A, B, C, D, E, F et G .

1. Quel est son ordre et le degré de chacun de ses sommets ?
2. Reproduire sur la copie et compléter le tableau des distances entre deux sommets de G :

Distance	A	B	C	D	E	F	G
A	X						
B	X	X					
C	X	X	X				
D	X	X	X	X			
E	X	X	X	X	X		
F	X	X	X	X	X	X	
G	X	X	X	X	X	X	X

En déduire le diamètre de ce graphe.

3. a) Donner un sous-graphe complet d'ordre 3 de G .
Qu'en déduire pour le nombre chromatique de G ?
b) Proposer une coloration du graphe G et en déduire son nombre chromatique.
4. Donner la matrice M associée à G (vous numéroterez les lignes et les colonnes dans l'ordre alphabétique).
5. En utilisant la matrice M_2 donnée en annexe 1, déduire le nombre de chaînes de longueur 2 partant de A sans y revenir.

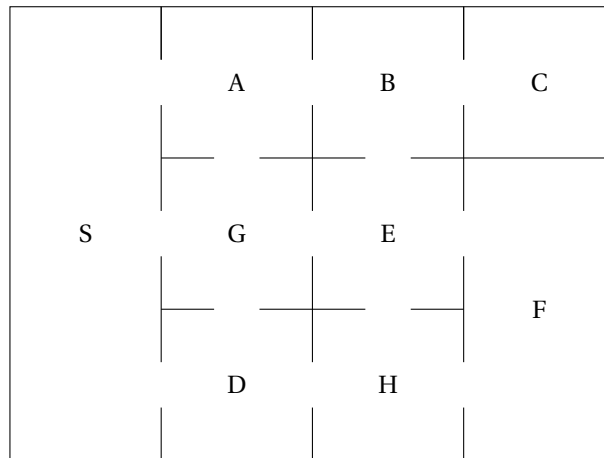


$$M^2 = \begin{pmatrix} 3 & 1 & 1 & 0 & 2 & 1 & 0 \\ 1 & 3 & 0 & 1 & 2 & 1 & 1 \\ 1 & 0 & 3 & 1 & 1 & 2 & 1 \\ 0 & 1 & 1 & 2 & 1 & 1 & 1 \\ 2 & 2 & 1 & 1 & 4 & 0 & 0 \\ 1 & 1 & 2 & 1 & 0 & 3 & 2 \\ 0 & 1 & 1 & 1 & 0 & 2 & 2 \end{pmatrix}$$

```
\begin{tikzpicture}
  \Vertex[x=1.3,y=3.8]{A}      \Vertex[x=4.2,y=5.5]{B}
  \Vertex[x=7.3,y=4]{C}      \Vertex[x=8.5,y=1.5]{D}
  \Vertex[x=5,y=0]{E}        \Vertex[x=3.6,y=4]{F}
  \Vertex[x=0.7,y=1]{G}
  \Edges(A,B,C,D,E,G,A,F,E,C) \Edge(B)(F)
\end{tikzpicture}
```

11.2 Antilles-Guyane juin 2003

1. Un musée est constitué de 9 salles notées A, B, C, D, E, F, G, H et S.
Le plan du musée est représenté ci-dessous :



Ainsi, un visiteur qui se trouve dans la salle S peut atteindre directement les salles A, B ou G. S'il se trouve dans la salle C, il peut se rendre directement dans la salle B, mais pas dans la salle F.

On s'intéresse au parcours d'un visiteur dans ce musée. On ne se préoccupe pas de la manière dont le visiteur accède au musée ni comment il en sort. Cette situation peut être modélisée par un graphe, les sommets étant les noms des salles, les arêtes représentant les portes de communication.

- Dessiner un graphe modélisant la situation décrite.
 - Est-il possible de visiter le musée, en empruntant chaque porte une fois et une seule? Justifier en utilisant un théorème du cours sur les graphes.
 - Pour rompre une éventuelle monotonie, le conservateur du musée souhaite différencier chaque salle de sa ou des salles voisines (c'est-à-dire accessibles par une porte) par la moquette posée au sol. Quel est le nombre minimum de types de moquettes nécessaires pour répondre à ce souhait? Justifier.
2. On note M la matrice à 9 lignes et 9 colonnes associée au graphe précédent, en convenant de l'ordre suivant des salles S, A, B, C, D, E, F, G, H. Le graphe n'étant pas orienté, comment cela se traduit-il sur la matrice?
3. On donne la matrice :

$$M^4 = \begin{pmatrix} 18 & 12 & 11 & 02 & 20 & 12 & 06 & 12 & 12 \\ 12 & 20 & 03 & 06 & 11 & 20 & 05 & 18 & 05 \\ 11 & 03 & 16 & 00 & 19 & 03 & 08 & 04 & 12 \\ 02 & 06 & 00 & 03 & 01 & 07 & 01 & 04 & 01 \\ 20 & 11 & 19 & 01 & 31 & 09 & 11 & 12 & 19 \\ 12 & 20 & 03 & 07 & 09 & 28 & 09 & 20 & 09 \\ 06 & 05 & 08 & 01 & 11 & 09 & 09 & 08 & 09 \\ 12 & 18 & 04 & 04 & 12 & 20 & 08 & 20 & 06 \\ 12 & 05 & 12 & 01 & 19 & 09 & 09 & 06 & 17 \end{pmatrix}$$

- Combien y-a-t-il de chemins qui en 4 étapes, partent de D et reviennent à D?
- Combien y-a-t-il de chemins qui en 4 étapes, partent de S et reviennent à C? Les citer.
- Est-il toujours possible de joindre en 4 étapes deux salles quelconques? Justifier.

Code du graphe précédent, uniquement fait avec tikz sans tkz-berge

```
\begin{tikzpicture}
  \draw (0,0) rectangle (8,6);
  \draw(2,0)--(2,0.7);
  \draw(2,1.3)--(2,2.7);
  \draw(2,3.3)--(2,4.7);
  \draw(2,5.3)--(2,6);
  \draw(4,0)--(4,0.7);
  \draw(4,1.3)--(4,2.7);
  \draw(4,3.3)--(4,4.7);
  \draw(4,5.3)--(4,6);
  \draw(6,0)--(6,0.7);
  \draw(6,1.3)--(6,2.7);
  \draw(6,3.3)--(6,4.7);
  \draw(6,5.3)--(6,6);
  \draw(2,5.3)--(2,6);
  \draw(4,5.3)--(4,6);
  \draw(6,5.3)--(6,6);
  \draw(2,2)--(2.7,2);
  \draw(3.3,2)--(4.7,2);
  \draw(5.3,2)--(6,2);
  \draw(2,4)--(2.7,4);
  \draw(3.3,4)--(4.7,4);
  \draw(5.3,4)--(8,4);
  \node at (1,3){S};
  \node at (3,3){G};
  \node at (3,1){D};
  \node at (3,5){H};
  \node at (5,1){H};
  \node at (5,3){E};
  \node at (5,5){B};
  \node at (7,2){F};
  \node at (7,5){C};
\end{tikzpicture}
```

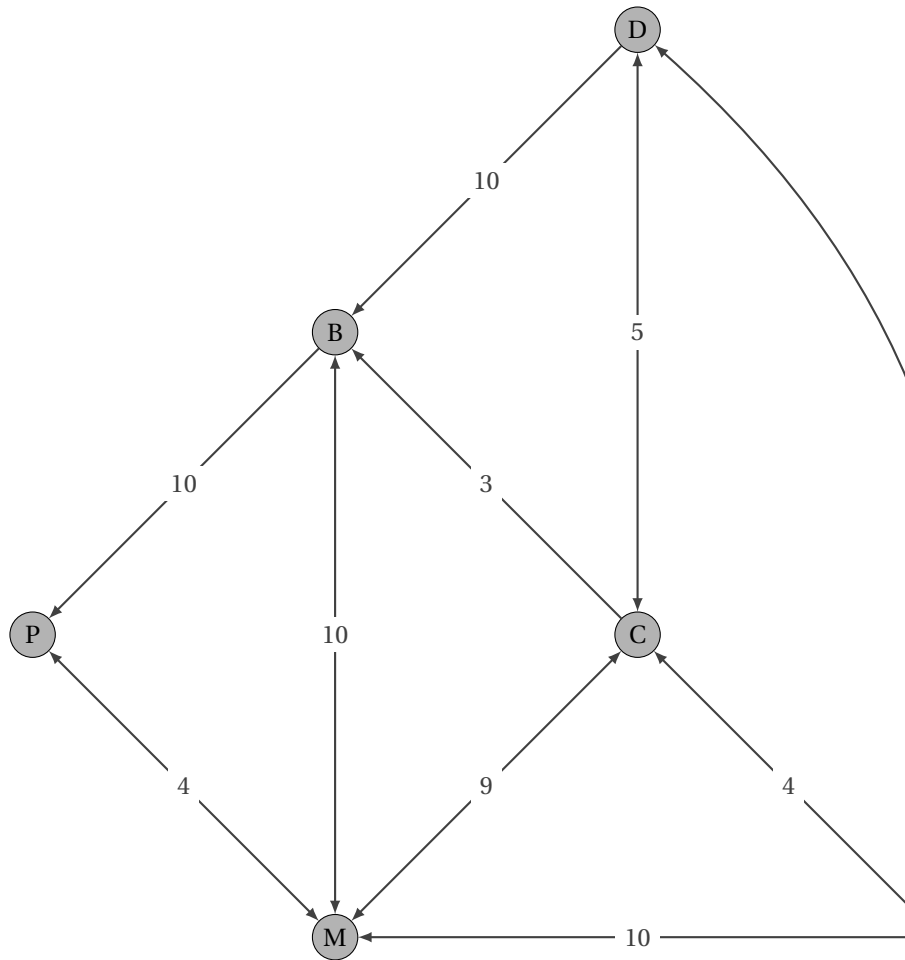
11.3 Asie juin 2003

Dans la ville de GRAPHE, on s'intéresse aux principales rues permettant de relier différents lieux ouverts au public, à savoir la mairie (M), le centre commercial (C), la bibliothèque (B), la piscine (P) et le lycée (L). Chacun de ces lieux est désigné par son initiale. Le tableau ci-contre donne les rues existant entre ces lieux.

	B	C	L	M	P
B		X		X	X
C	X		X	X	
L		X		X	
M	X	X	X		X
P	X			X	

- Dessiner un graphe représentant cette situation.
- Montrer qu'il est possible de trouver un trajet empruntant une fois et une seule toutes les rues de ce plan. Justifier. Proposer un tel trajet.
Est-il possible d'avoir un trajet partant et arrivant du même lieu et passant une fois et une seule par toutes les rues?

- Dimitri habite dans cette ville; le graphe ci-contre donne le **nouveau** plan du quartier avec les sens de circulation dans les différentes rues et le temps de parcours entre les différents lieux.



Code du graphe précédent

```

\begin{minipage}[c]{\Q,68\textwidth}
\begin{tikzpicture}[>=latex]
  \SetGraphUnit{4}
  \tikzset{VertexStyle/.style = {shape          = circle,
                                draw            = black,
                                inner sep      = 2pt,%
                                minimum size   = 6mm,
                                outer sep     = 0pt,
                                fill          = gray!60!}}

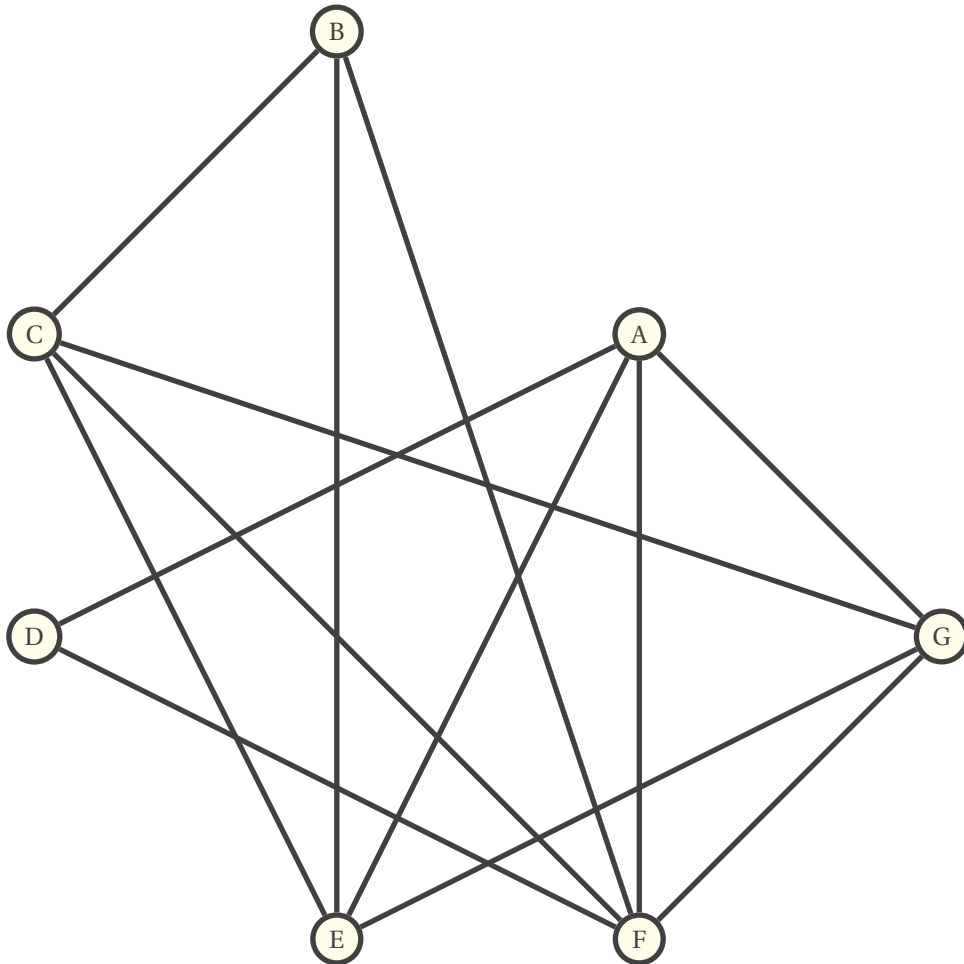
  \Vertex {P}
  \NOEA(P){B}
  \SOEA(P){M}
  \NOEA(B){D}
  \SOEA(B){C}
  \SOEA(C){L}
  \tikzset{LabelStyle/.style = {fill=white}}
  \tikzset{EdgeStyle/.style = {<->}}
  \Edge[label=$4$] (P)(M)
  \Edge[label=$9$] (C)(M)
  \Edge[label=$4$] (C)(L)
  \Edge[label=$5$] (C)(D)
  \Edge[label=$10$] (B)(M)
  \tikzset{EdgeStyle/.style = {<->,bend right}}
  \Edge[label=$11$] (L)(D)
  \tikzset{EdgeStyle/.style = {->}}
  \Edge[label=$3$] (C)(B)
  \Edge[label=$10$] (D)(B)
  \Edge[label=$10$] (L)(M)
  \Edge[label=$10$] (B)(P)
\end{tikzpicture}
\end{minipage}

```

11.4 France juin 2003

Un concert de solidarité est organisé dans une grande salle de spectacle. À ce concert sont conviés sept artistes de renommée internationale Luther Allunison (A), John Biaisé (B), Phil Colline (C), Bob Ditrâne (D), Jimi Endisque (E), Robert Fripe (F) et Rory Garaguerre (G).

Les différents musiciens invités refusant de jouer avec certains autres, l'organisateur du concert doit prévoir plusieurs parties de spectacle. Les arêtes du graphe Γ ci-dessous indiquent quels sont les musiciens qui refusent de jouer entre eux.

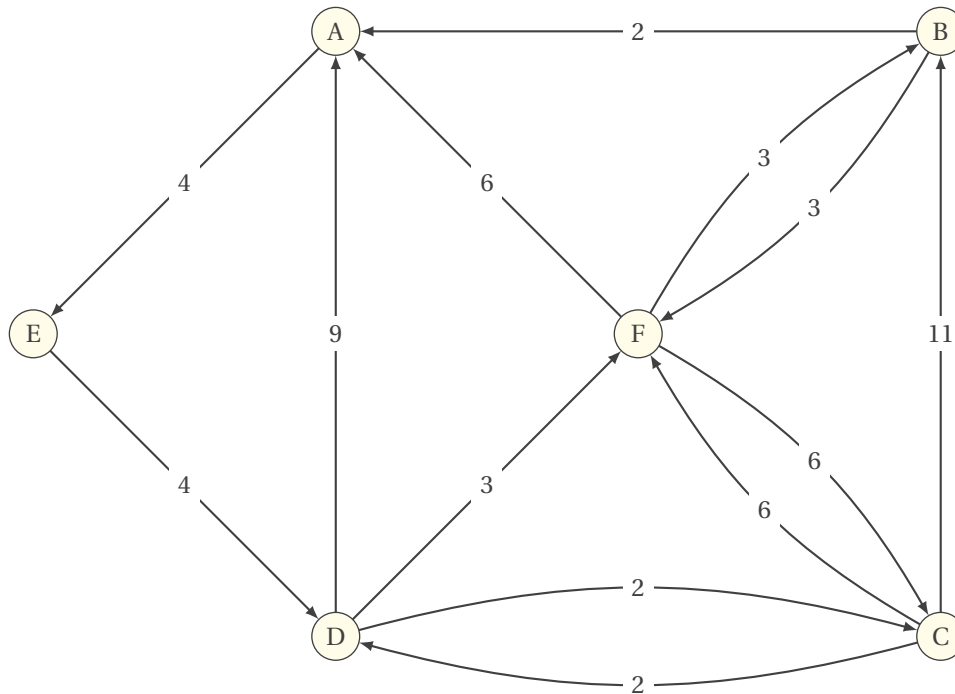


1. Déterminer la matrice associée au graphe Γ (les sommets de Γ étant classés dans l'ordre alphabétique).
2. Quelle est la nature du sous-graphe de Γ constitué des sommets A, E, F et G?
Que peut-on en déduire pour le nombre chromatique $\chi(\Gamma)$ du graphe Γ ?
3. Quel est le sommet de plus haut degré de Γ ?
En déduire un encadrement de $\chi(\Gamma)$.
4. Après avoir classé l'ensemble des sommets de Γ par ordre de degré décroissant, colorier le graphe Γ figurant en annexe.
5. Combien de parties l'organisateur du concert doit-il prévoir?
Proposer une répartition des musiciens pour chacune de ces parties.

```
\begin{tikzpicture}
  \SetGraphUnit{4}
  \GraphInit[vstyle=Normal]
  \tikzset{EdgeStyle/.style = {line width = 2pt}}
  \tikzset{VertexStyle/.append style = {line width = 2pt}}
  \Vertex{D}
  \SOEA(D){E}\EA(E){F}
  \NOEA(F){G}\NOWE(G){A}
  \NOWE(A){B}\SOWE(B){C}
  \Edges(F,G,A,D,F,B,E,G,C,F,A,E,C,B)
\end{tikzpicture}
```

11.5 Centres Étrangers juin 2003

Un livreur d'une société de vente à domicile doit, dans son après-midi, charger son camion à l'entrepôt noté A, livrer cinq clients que nous noterons B, C, D, E et F, puis retourner à l'entrepôt. Le réseau routier, tenant compte des sens de circulation, et les temps de parcours (en minutes) sont indiqués sur le graphe G suivant :



1. Donner la matrice M associée au graphe G.

On utilisera le modèle suivant :

	A	B	C	D	E	F
A						
B						
C						
D						
E						
F						

2. On donne la matrice M^6 :

$$M^6 = \begin{pmatrix} 8 & 6 & 6 & 3 & 4 & 6 \\ 19 & 11 & 12 & 9 & 6 & 16 \\ 36 & 28 & 23 & 22 & 18 & 34 \\ 37 & 24 & 25 & 17 & 15 & 31 \\ 15 & 12 & 9 & 10 & 8 & 15 \\ 28 & 22 & 19 & 15 & 15 & 26 \end{pmatrix}$$

On s'intéresse aux chemins partant de l'entrepôt A et se terminant en A.

- Combien existe-t-il de chemins de longueur 6 reliant A à A?
- Citer ces chemins.
- Parmi ceux qui passent par tous les sommets du graphe, lequel minimise le temps de parcours?
- Quelle conséquence peut tirer le livreur du dernier résultat?

3. Au départ de sa tournée, le livreur a choisi de suivre l'itinéraire le plus rapide. Malheureusement, le client C n'est pas présent au passage du livreur et celui-ci décide de terminer sa livraison par ce client. Indiquer quel est le chemin le plus rapide pour revenir à l'entrepôt A à partir de C. La réponse devra être justifiée.

Code du graphe précédent

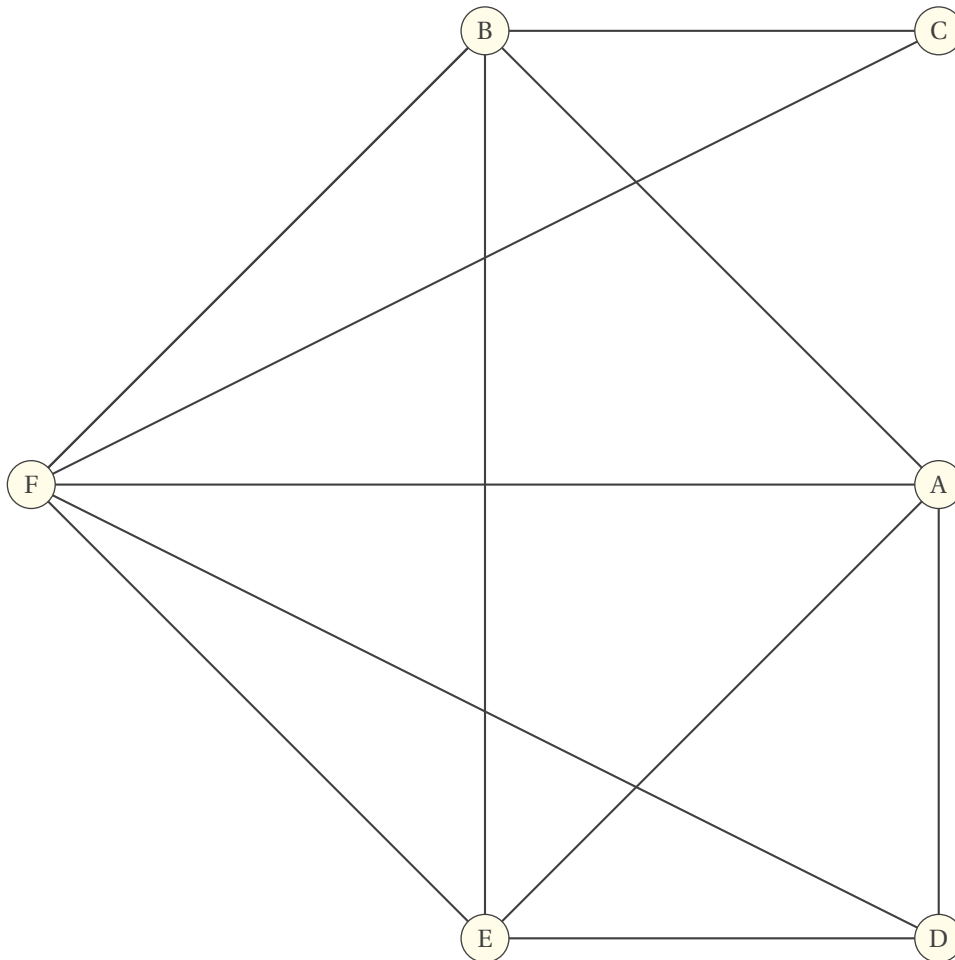
```
\begin{tikzpicture}[>=latex]
  \SetGraphUnit{3}
  \Vertex {F}
  \NOWE(F){A}
  \NOEA(F){B}
  \SOEA(F){C}
  \SOWE(F){D}
  \SOWE(A){E}
  \tikzstyle{EdgeStyle}=[->]
  \tikzstyle{LabelStyle}=[fill=white]
  \Edge[label=$4$](A)(E)
  \Edge[label=$4$](E)(D)
  \Edge[label=$9$](D)(A)
  \Edge[label=$2$](B)(A)
  \Edge[label=$11$](C)(B)
  \Edge[label=$3$](D)(F)
  \Edge[label=$6$](F)(A)
  \tikzstyle{EdgeStyle}=[->,bend left=15]
  \Edge[label=$2$](D)(C)
  \Edge[label=$2$](C)(D)
  \Edge[label=$3$](F)(B)
  \Edge[label=$3$](B)(F)
  \Edge[label=$6$](F)(C)
  \Edge[label=$6$](C)(F)
\end{tikzpicture}
```

11.6 Amérique du Nord juin 2004

Les parties A et B sont indépendantes.

Partie A

On considère le graphe G_1 ci-dessous :



1. Justifier les affirmations suivantes :

A_1 : « le graphe G_1 admet au moins une chaîne eulérienne ».

A_2 : « La chaîne DABCFBEFAE n'est pas une chaîne eulérienne de G_1 ».

- Déterminer un sous-graphe complet de G_1 , ayant le plus grand ordre possible. En déduire un minorant du nombre chromatique γ de ce graphe.
- Déterminer un majorant de ce nombre chromatique. (On justifiera la réponse).
- En proposant une coloration du graphe G_1 , déterminer son nombre chromatique.

Partie B

Soit la matrice M d'un graphe orienté G_2 dont les sommets A, B, C, D et E sont pris dans l'ordre alphabétique.

On donne

$$M = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

et

$$M^3 = \begin{pmatrix} 6 & 6 & 4 & 5 & 3 \\ 5 & 6 & 5 & 3 & 6 \\ 5 & 7 & 4 & 3 & 6 \\ 3 & 5 & 3 & 3 & 3 \\ 6 & 6 & 3 & 3 & 5 \end{pmatrix}.$$

1. Construire le graphe G_2 .
2. Déterminer le nombre de chaînes de longueur 3 reliant B à D. Les citer toutes.

Code du graphe précédent

```
\begin{tikzpicture}[>=latex]
  \SetGraphUnit{6}
  \Vertex{F}
  \NOEA(F){B}
  \SOEA(F){E}
  \EA(B){C}
  \EA(E){D}
  \NO(D){A}
  \Edges(B,F,E,D,A,E,B,A,F,B,C,F,D)
\end{tikzpicture}
```

11.7 Centres étrangers mai 2004

Un jardinier possède un terrain bien ensoleillé avec une partie plus ombragée. Il décide d'y organiser des parcelles où il plantera 8 variétés de légumes :

- de l'ail (A),
- des courges (Co),
- des choux (Ch),
- des poireaux (Px),
- des pois (Po),
- des pommes de terre (Pt),
- des radis (R),
- et des tomates (T).

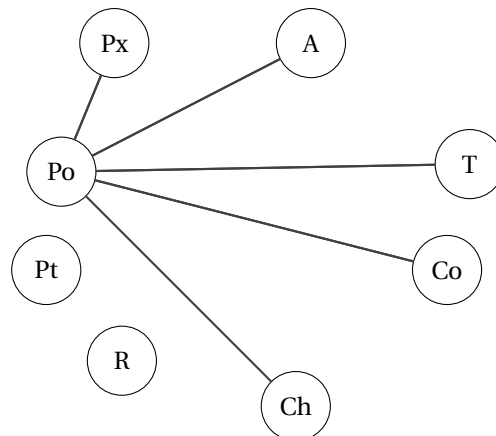
Il consulte un almanach où figurent des incompatibilités de plantes, données par les deux tableaux :

Expositions incompatibles de plantes	
Plantes d'ombre partielle	Plantes de plein soleil
pois radis	choux tomates courges
Par exemple : les pois sont incompatibles avec les choux, les tomates et les courges	

Associations incompatibles de plantes dans une même parcelle	
pois	ail, poireaux
potatoes de terre	courges, radis et tomates
choux	tomates, ail poireaux et courges
courges	tomates
Par exemple : les pois sont incompatibles avec l'ail et les poireaux	

Pour tenir compte de ces incompatibilités le jardinier décide de modéliser la situation sous la forme d'un graphe de huit sommets, chaque sommet représentant un légume.

1. Sur la feuille annexe : compléter le graphe mettant en évidence les incompatibilités d'exposition ou les associations incompatibles indiquées dans les deux tableaux ci-dessus.
2. Calculer la somme des degrés des sommets du graphe, en déduire le nombre de ses arêtes.
3. Rechercher un sous-graphe complet d'ordre 4, qu'en déduit-on pour le nombre chromatique du graphe ?
4. Donner le nombre chromatique du graphe et l'interpréter en nombre minimum de parcelles que le jardinier devra créer.
5. Donner une répartition des plantes par parcelle de façon à ce que chaque parcelle contienne exactement deux types de plantes et que le nombre de parcelles soit minimum.
6. Donner une répartition des plantes de façon à ce qu'une parcelle contienne trois plantes et que le nombre de parcelles soit minimum.



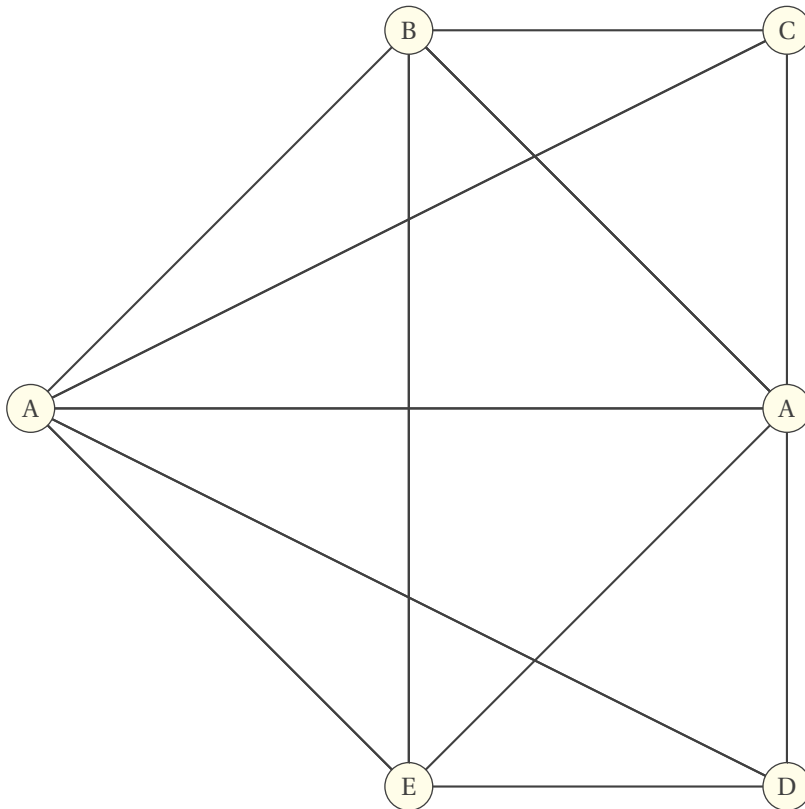
Code du graphe précédent

```
\begin{tikzpicture}
\tikzstyle{VertexStyle}= [shape      = circle,
                           fill        = white,%
                           minimum size = 26pt,%
                           draw]

\Vertex[x=1,y=0.8]{R}
\Vertex[x=0.2,y=3.3]{Po}
\Vertex[x=0,y=2]{Pt}
\Vertex[x=0.9,y=5]{Px}
\Vertex[x=3.5,y=5]{A}
\Vertex[x=5.6,y=3.4]{T}
\Vertex[x=5.3,y=2]{Co}
\Vertex[x=3.3,y=0.2]{Ch}
\Edges(Po,Px,Po,A,Po,T,Po,Co,Po,Ch)
\end{tikzpicture}
```

11.8 France juin 2004

Le graphe ci-dessous indique, sans respecter d'échelle, les parcours possibles entre les sept bâtiments d'une entreprise importante.



Un agent de sécurité effectue régulièrement des rondes de surveillance. Ses temps de parcours en minutes entre deux bâtiments sont les suivants :

- AB : 16 minutes ;
- AG : 12 minutes ;
- BC : 8 minutes ;
- BE : 12 minutes ;
- BG : 8 minutes ;
- CD : 7 minutes ;
- CE : 4 minutes ;
- CG : 10 minutes ;
- DE : 2 minutes ;
- EF : 8 minutes ;
- EG : 15 minutes ;
- FG : 8 minutes.

Sur chaque arête, les temps de parcours sont indépendants du sens de parcours.

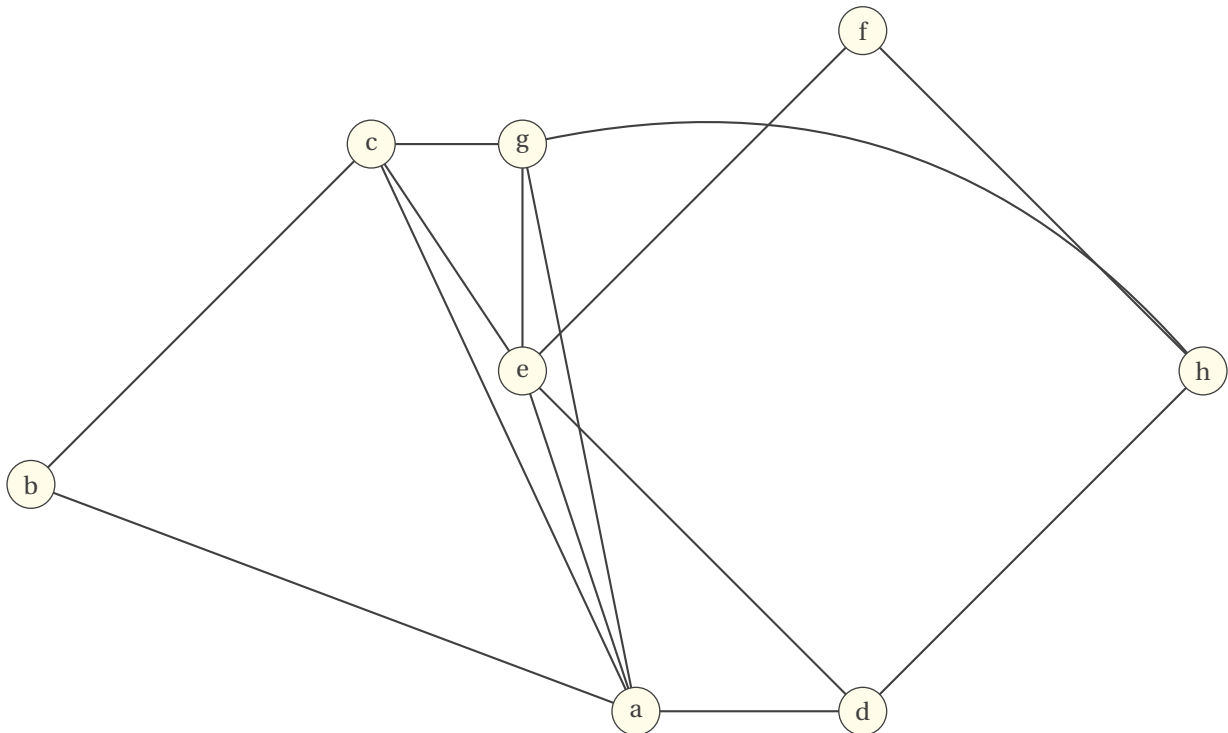
1. En justifiant la réponse, montrer qu'il est possible que l'agent de sécurité passe une fois et une seule par tous les chemins de cette usine. Donner un exemple de trajet.
2. L'agent de sécurité peut-il revenir à son point de départ après avoir parcouru une fois et une seule tous les chemins? Justifier la réponse.
3. Tous les matins, l'agent de sécurité part du bâtiment A et se rend au bâtiment D.
En utilisant un algorithme que l'on explicitera, déterminer le chemin qu'il doit suivre pour que son temps de parcours soit le plus court possible, et donner ce temps de parcours.

```
\begin{tikzpicture}
  \SetGraphUnit{5}
  \Vertex{A}    \NOEA(F){B}  \SOEA(F){E}
  \EA(B){C}    \EA(E){D}    \NO(D){A}
  \Edges(F,E,F,D,F,C,F,A,F,B,A,E,E,D,D,A,B,A,C,B,E,B)
\end{tikzpicture}
```

11.9 La Réunion juin 2004

Partie A

On note G le graphe représenté ci-dessous et M sa matrice obtenue en prenant les sommets dans l'ordre alphabétique. La matrice M^3 est également donnée.



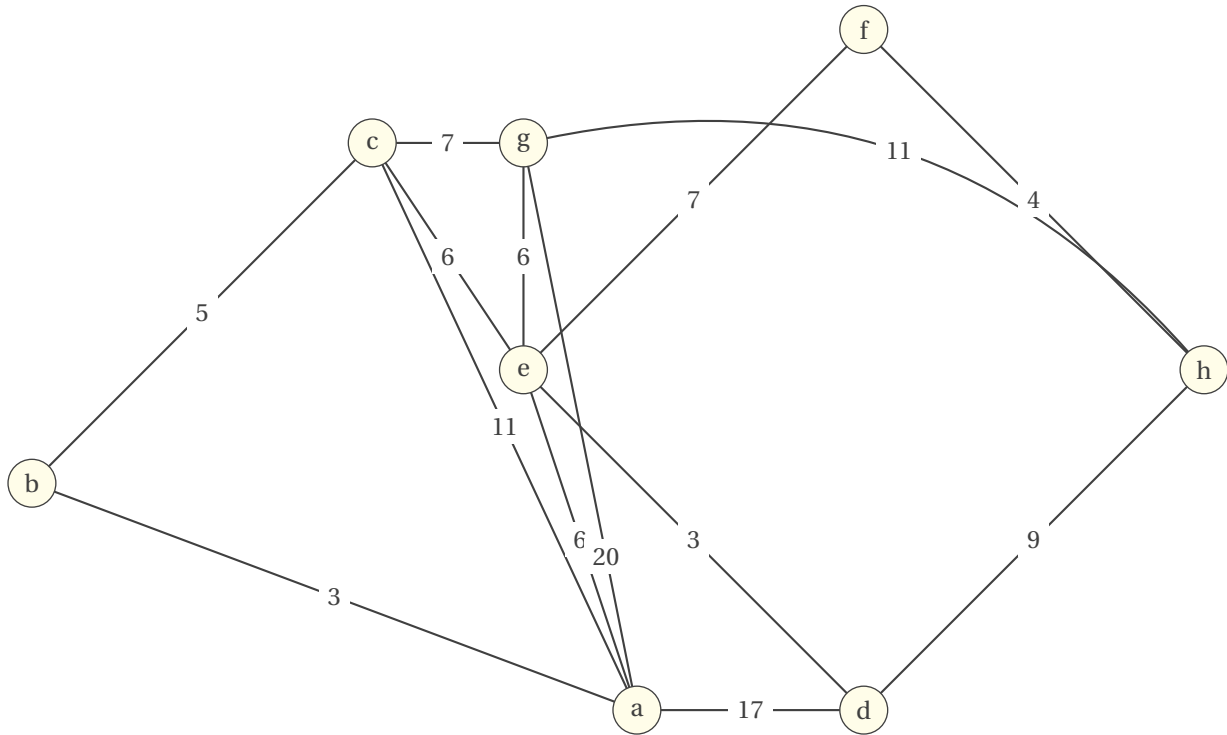
$$M^3 = \begin{pmatrix} 10 & 8 & 11 & 10 & 12 & 5 & 13 & 4 \\ 8 & 2 & 7 & 3 & 5 & 2 & 4 & 3 \\ 11 & 7 & 8 & 6 & 12 & 3 & 10 & 5 \\ 10 & 3 & 6 & 2 & 11 & 1 & 4 & 8 \\ 12 & 5 & 12 & 11 & 8 & 8 & 13 & 3 \\ 5 & 2 & 3 & 1 & 8 & 0 & 2 & 6 \\ 13 & 4 & 10 & 4 & 13 & 2 & 6 & 9 \\ 4 & 3 & 5 & 8 & 3 & 6 & 9 & 0 \end{pmatrix}$$

Dire, en justifiant votre réponse, si les affirmations suivantes sont vraies ou fausses :

1. L'ordre du graphe est égal au plus grand des degrés des sommets.
2. Le graphe G contient un sous-graphe complet d'ordre 3.
3. Les sommets de G peuvent être coloriés avec trois couleurs sans que deux sommets adjacents soient de même couleur.
4. Il est possible de parcourir ce graphe en passant une fois et une seule par chaque arête.
5. Il existe au moins un chemin de longueur 3 qui relie chaque sommet à chacun des sept autres sommets du graphe.
6. il y a 72 chemins de longueur 3 qui relient le sommet e à chacun des huit sommets du graphe.

Partie B

Le graphe suivant représente un réseau de lignes d'autobus. Les sommets du graphe désignent les arrêts. Les poids des arêtes sont les durées de parcours, en minutes, entre deux arrêts (correspondances comprises).



Déterminer, à l'aide d'un algorithme, la durée minimum pour aller de l'arrêt *a* à l'arrêt *h* et donner ce trajet.

Code du graphe précédent

```

\begin{tikzpicture}[>=latex]
  \SetGraphUnit{4.5}
  \Vertex {e}
  \NOEA(e){f}
  \SOEA(e){d}
  \SOEA(f){h}
  \Vertex[position={above of=e,yshift=2cm}]{g}
  \Vertex[position={left of=g,xshift=-1cm}]{c}
  \Vertex[position={left of=d,xshift=-2cm}]{a}
  \SOWE(c){b}
  \Edges(a,c,g) \Edges(d,h,f,e,d,a,e,g,a,b,c,e)
  \Edge[style={bend left}](g)(h)
\end{tikzpicture}

```

et

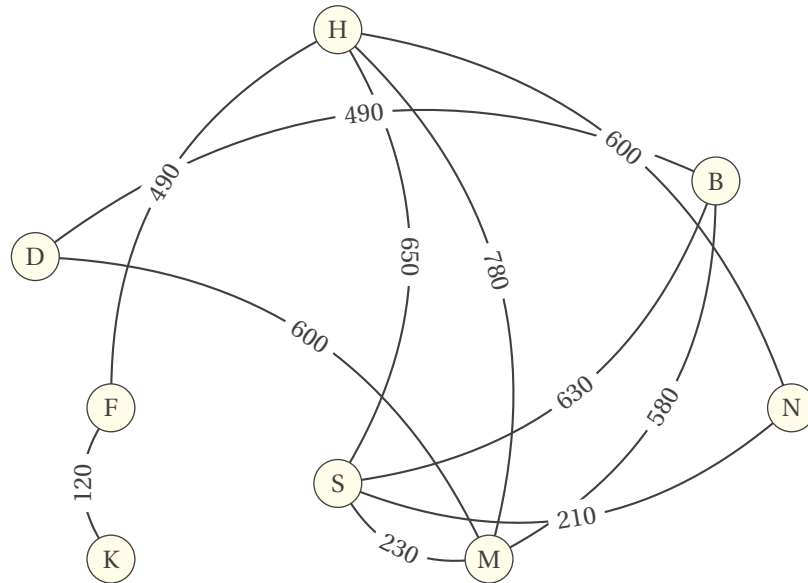
```

\begin{tikzpicture}[>=latex]
  \SetGraphUnit{4.5}
  \Vertex {e}
  \NOEA(e){f}
  \SOEA(e){d}
  \SOEA(f){h}
  \Vertex[position={above of=e,yshift=2cm}]{g}
  \Vertex[position={left of=g,xshift=-1cm}]{c}
  \Vertex[position={left of=d,xshift=-2cm}]{a}
  \SOWE(c){b}
  \tikzstyle{LabelStyle}=[fill=white]
  \Edge[label=$3$](a)(b)
  \Edge[label=$11$](a)(c)
  \Edge[label=$6$](a)(e)
  \Edge[label=$17$](a)(d)
  \Edge[style={pos=.25},label=$20$](a)(g)
  \Edge[label=$5$](c)(b)
  \Edge[label=$6$](c)(e)
  \Edge[label=$7$](c)(g)
  \Edge[label=$7$](f)(e)
  \Edge[label=$3$](d)(e)
  \Edge[label=$9$](d)(h)
  \Edge[label=$6$](g)(e)
  \Edge[style={bend left},label=$11$](g)(h)
  \Edge[label=$4$](f)(h)
\end{tikzpicture}

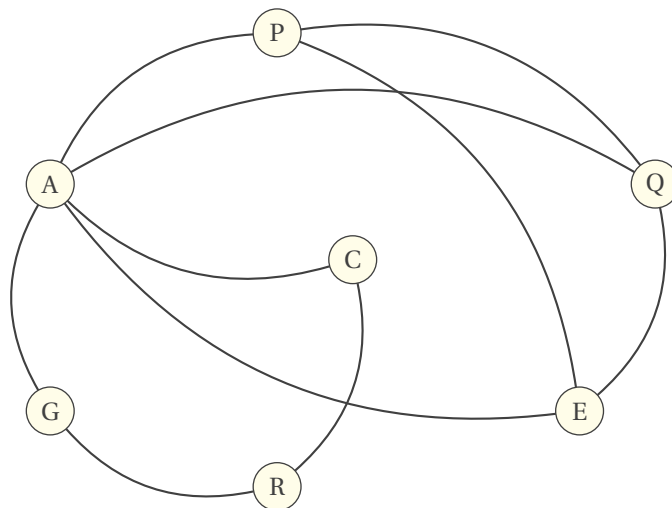
```


11.10 Amérique du Sud Nov 2006

1. À l'occasion de la coupe du monde de football 2006 en Allemagne, une agence touristique organise des voyages en car à travers les différentes villes où se joueront les matchs d'une équipe nationale. Les routes empruntées par les cars sont représentées par le graphe ci-dessous. Le long de chaque arête figure la distance en kilomètres séparant les villes. Les lettres B, D, F, H, K, M, N et S représentent les villes Berlin, Dortmund, Francfort, Hambourg, Kaiserslautern, Munich, Nuremberg et Stuttgart.



- En précisant la méthode utilisée, déterminer le plus court chemin possible pour aller de Kaiserslautern à Berlin en utilisant les cars de cette agence.
2. Pour des raisons de sécurité, les supporters de certaines équipes nationales participant à la coupe du monde de football en 2006 ne peuvent être logés dans le même hôtel. On donne ci-dessous le graphe d'incompatibilité entre les supporters de différentes équipes : par exemple, un supporter de l'équipe A ne peut être logé avec un supporter de l'équipe P.

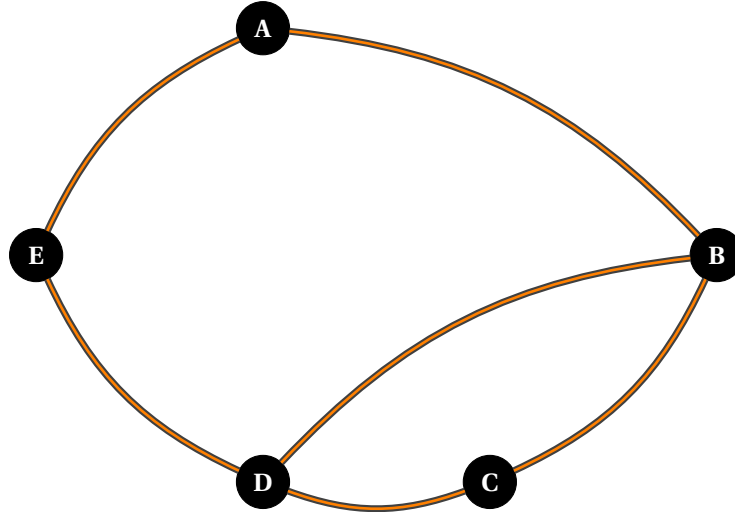


- a) Déterminer le nombre chromatique de ce graphe en justifiant la valeur trouvée.
- b) Proposer une répartition des supporters par hôtel en utilisant un nombre minimum d'hôtels.

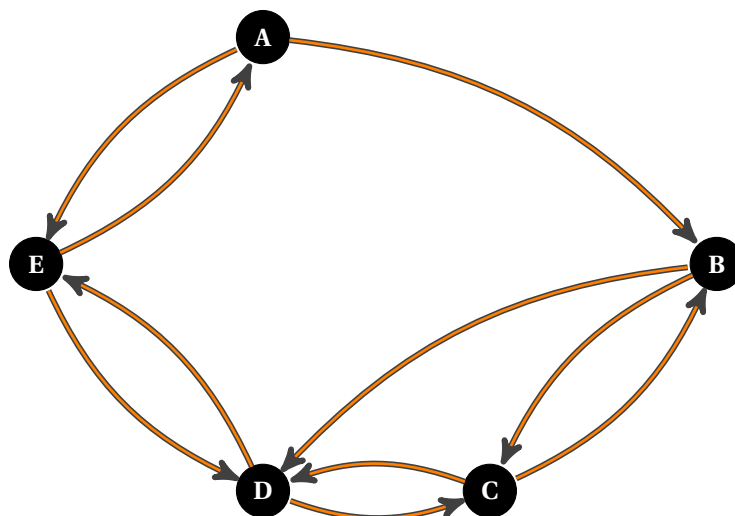
11.11 Liban juin 2006

1. Dans un parc, il y a cinq bancs reliés entre eux par des allées.

On modélise les bancs par les sommets A, B, C, D, E et les allées par les arêtes du graphe G ci-dessous :



- a) On désire peindre les bancs de façon que deux bancs reliés par une allée soient toujours de couleurs différentes.
Donner un encadrement du nombre minimal de couleurs nécessaires et justifier.
Déterminer ce nombre.
- b) Est-il possible de parcourir toutes les allées de ce parc sans passer deux fois par la même allée?
2. Une exposition est organisée dans le parc. La fréquentation devenant trop importante, on décide d'instaurer un plan de circulation : certaines allées deviennent à sens unique, d'autres restent à double sens. Par exemple la circulation dans l'allée située entre les bancs B et C pourra se faire de B vers C et de C vers B, alors que la circulation dans l'allée située entre les bancs A et B ne pourra se faire que de A vers B. Le graphe G' ci-dessous modélise cette nouvelle situation :



- a) Donner la matrice M associée au graphe G' . (On ordonnera les sommets par ordre alphabétique).

b) On donne $M^5 = \begin{pmatrix} 1 & 6 & 9 & 6 & 10 \\ 4 & 5 & 7 & 11 & 5 \\ 4 & 6 & 6 & 11 & 5 \\ 1 & 5 & 10 & 6 & 10 \\ 6 & 5 & 5 & 14 & 2 \end{pmatrix}$

Combien y a-t-il de chemins de longueur 5 permettant de se rendre du sommet D au sommet B?

Les donner tous.

c) Montrer qu'il existe un seul cycle de longueur 5 passant par le sommet A.

Quel est ce cycle?

En est-il de même pour le sommet B?

Code des graphes précédents

```

\begin{tikzpicture}
  \SetGraphUnit{3}
  \tikzstyle{VertexStyle}=[shape          = circle,
                           fill           = black,
                           minimum size  = 20pt,
                           text          = white,
                           draw]
  \Vertex[L= {\textbf{E}}]{E}
  \NOEA[L = {\textbf{A}}](E){A}
  \SOEA[L = {\textbf{D}}](E){D}
  \EA[L   = {\textbf{C}}](D){C}
  \NOEA[L = {\textbf{B}}](C){B}
  \tikzstyle{EdgeStyle}=[double          = orange,
                          double distance = 1pt,
                          thick,
                          bend right     = 20]
  \Edges(B,A,E,D,C,B,D)
\end{tikzpicture}

```

```

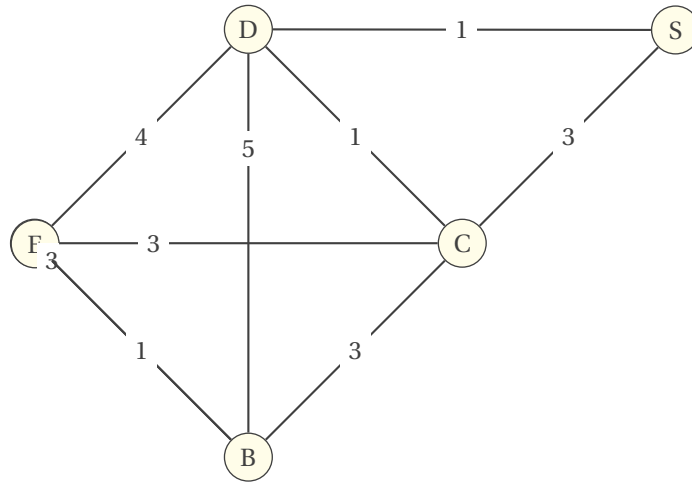
\begin{tikzpicture}
  \SetGraphUnit{3}
  \tikzstyle{VertexStyle}=[shape          = circle,
                           fill           = black,
                           minimum size  = 20pt,
                           text          = white,
                           draw]
  \tikzstyle{TempStyle}=[double          = orange,
                          double distance = 1pt]
  \Vertex[L= {\textbf{E}}]{E}
  \NOEA[L = {\textbf{A}}](E){A}
  \SOEA[L = {\textbf{D}}](E){D}
  \EA[L   = {\textbf{C}}](D){C}
  \NOEA[L = {\textbf{B}}](C){B}
  \tikzstyle{EdgeStyle}=[TempStyle,
                          post,
                          bend right     = 20]
  \Edges(A,E,D,C,B,D)
  \tikzstyle{EdgeStyle}=[TempStyle,%
                          pre,%
                          bend right     = 20]
  \Edges(B,A)
  \tikzstyle{EdgeStyle}=[TempStyle,%
                          pre,%
                          bend left      = 20]
  \Edges(A,E,D,C,B)
\end{tikzpicture}

```

12 Dijkstra

Algorithme de Dijkstra : Plus courte chaîne du sommet E au sommet S .

12.1 Dijkstra exemple 1



```

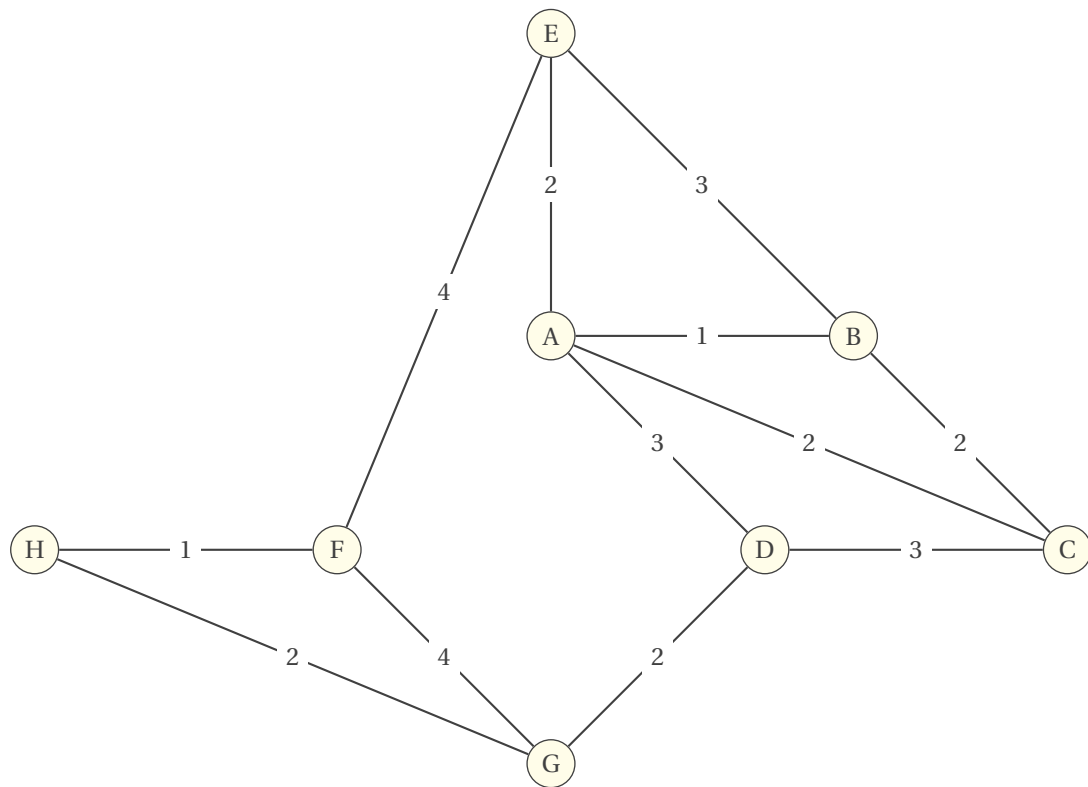
\begin{tikzpicture}
  \GraphInit[vstyle=Dijkstra]
  \SetGraphUnit{4}
  \Vertices[square]{B,C,D,A}
  \SetGraphUnit{2.82}
  \NOWE(B){E}
  \NOEA(C){S}
  \Edge[label=$3$](E)(A)
  \Edge[label=$1$](E)(B)
  \Edge[label=$1$](A)(B)
  \Edge[label=$3$](B)(C)
  \Edge[label=$3$,style={pos=.25}](A)(C)
  \Edge[label=$5$,style={pos=.75}](B)(D)
  \Edge[label=$4$](A)(D)
  \Edge[label=$1$](S)(D)
  \Edge[label=$3$](C)(S)
  \Edge[label=$1$](C)(D)
\end{tikzpicture}

```

E	A	B	C	D	S	Choix
0	∞	∞	∞	∞	∞	E
█	3(E)	1(E)	∞	∞	∞	B
█	█ 2(B)	█	4(B)	6(B)	∞	A
█	█	█	█ 4(B)	6(B)	∞	C
█	█	█	█	█ 5(C)	7(C)	D
█	█	█	█	█	6(D)	S

Le plus court chemin est donc $EBCDS$

12.2 Dijkstra exemple 2



```

\begin{tikzpicture}
  \GraphInit[vstyle=Dijkstra]
  \SetGraphUnit{4}
  \Vertices{square}{G,D,A,F}
  \WE(F){H}
  \EA(A){B}
  \EA(D){C}
  \NO(A){E}
  \Edge[label=$1$](H)(F)
  \Edge[label=$4$](G)(F)
  \Edge[label=$2$](H)(G)
  \Edge[label=$2$](G)(D)
  \Edge[label=$3$](D)(C)
  \Edge[label=$4$](F)(E)
  \Edge[label=$3$](A)(D)
  \Edge[label=$2$](A)(E)
  \Edge[label=$1$](A)(B)
  \Edge[label=$2$](A)(C)
  \Edge[label=$2$](C)(B)
  \Edge[label=$3$](E)(B)
\end{tikzpicture}

```

<i>H</i>	<i>F</i>	<i>G</i>	<i>E</i>	<i>D</i>	<i>A</i>	<i>C</i>	<i>B</i>	Choix
0	∞	∞	∞	∞	∞	∞	∞	<i>H</i>
	1(<i>H</i>)	2(<i>H</i>)	∞	∞	∞	∞	∞	<i>F</i>
		2(<i>H</i>)	5(<i>F</i>)	∞	∞	∞	∞	<i>G</i>
			5(<i>F</i>)	4(<i>G</i>)	∞	∞	∞	<i>D</i>
			5(<i>F</i>)		7(<i>D</i>)	7(<i>D</i>)	∞	<i>E</i>
					7(<i>D</i>)	7(<i>D</i>)	8(<i>E</i>)	<i>A</i>
						7(<i>D</i>)	8(<i>E</i>)	<i>C</i>
							8(<i>E</i>)	<i>B</i>

Le plus court chemin est donc *HFE*B

```

\def\ry{\vrule width 5pt$}
\def\iy{\infty$}
\ vbox{\tabskip=0pt \offinterlineskip
\def\ablerule{\noalign{\hskip\tabskip\hrule}}
\halign to \hsize{\strut#\vrule # \tabskip=0.6em plus8em&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#&
\hfil#\hfil& \vrule#\tabskip=0pt\cr\ablerule
&& $H$ && $F$ && $G$ && $E$ && $D$ && $A$ && $C$ && $B$%
&& Choix && \cr\ablerule
&& $0$ && \iy && \iy && \iy && \iy && \iy && \iy && \iy && \iy%
&& $H$ && \cr\ablerule
&& \ry && $1(H)$ && $2(H)$ && \iy && \iy && \iy && \iy && \iy && \iy%
&& $F$ && \cr\ablerule
&& \ry && \ry && $2(H)$ && $5(F)$ && \iy && \iy && \iy && \iy && \iy%
&& $G$ && \cr\ablerule
&& \ry && \ry && \ry && $5(F)$ && $4(G)$ && \iy && \iy && \iy && \iy%
&& $D$ && \cr\ablerule
&& \ry && \ry && \ry && $5(F)$ && \ry && $7(D)$ && $7(D)$ && \iy && \iy%
&& $E$ && \cr\ablerule
&& \ry && \ry && \ry && \ry && \ry && $7(D)$ && $7(D)$ && $8(E)$%
&& $A$ && \cr\ablerule
&& \ry && \ry && \ry && \ry && \ry && \ry && $7(D)$ && $8(E)$%
&& $C$ && \cr\ablerule
&& \ry && \ry && \ry && \ry && \ry && \ry && \ry && $8(E)$%
&& $B$ && \cr\ablerule}}

```

Index

`\AddVertexColor{<color>}{<List of vertices>}`, 47
`\AddVertexColor`, 4, 47

`\EA`, 4, 11
`\Edge`, 4, 21, 32
`\Edge`: options
 color, 21
 labelcolor, 21
 labelstyle, 21
 labeltext, 21
 label, 21
 local, 21
 lw, 21
 style, 21
`\Edges`, 4, 6, 23
`\Edges`: options
 color, 23
 labelstyle, 23
 label, 23
 lw, 23
 style, 23

`\GraphInit`, 6, 24
`\GraphInit`: options
 vstyle, 24
`\GraphInit[<local options>]`, 24
`\grProb`, 4, 44, 45
`\grProb`: arguments
 Vertex-left, 44
 Vertex-right, 44
 label E, 44
 label N, 44
 label S, 44
 label W, 44
`\grProb`: options
 Ldist, 44
 LoopDist, 44
 LposA, 44
 LposB, 44
 unit, 44
`\grProbThree`, 46
`\grProbThree`: arguments
 Vertex-down, 46
 Vertex-right, 46
 Vertex-up, 46
 dd/dr/du, 46
 rr/ru/rd, 46
 uu/ud/ur, 46
`\grProbThree`: options
 Ldist, 46
 LoopDist, 46
 LposA, 46
 LposB, 46
 unit, 46
`\grProbThree[<local options>] {<right>} {<up>}{<down>} {<rr/ru/rd>}{<uu/ud/ur>}{<dd/dr/du>}`, 46
`\grProb[<local options>] {<left>} {<right>} {<N>}{<S>}{<W>}{<E>}`, 44

`\Loop`, 4, 22

`\Loop`: options

- `color`, 22
- `labelstyle`, 22
- `label`, 22
- `lw`, 22
- `style`, 22

`\NO`, 4, 11

`\NOEA`, 4, 6, 11

`\NOWE`, 4, 11

Package

- `tkz-berge`, 36

`\SetGraphArtColor{<ball color>}{<color>}`, 37

`\SetGraphArtColor`, 4, 37

`\SetGraphColor{<fill color>}{<color>}`, 38

`\SetGraphColor`, 4, 38

`\SetGraphShadeColor{<ball color>}{<color>}{<double>}`, 36

`\SetGraphShadeColor`, 4, 36

`\SetGraphUnit` , 12

`\SetGraphUnit{2}`, 11

`\SetGraphUnit{<nombre>}`, 11

`\SetGraphUnit`, 11, 14, 15

`\SetUpEdge`, 4, 24, 33, 34

`\SetUpEdge`: options

- `color`, 33
- `labelstyle`, 33
- `labeltext`, 33
- `label`, 33
- `lw`, 33
- `style`, 33

`\SetUpEdge[<local options>]`, 33

`\SetUpVertex`, 4, 24, 31

`\SetUpVertex`: options

- `LabelOut`, 31
- `Ldist`, 31
- `Lpos`, 31
- `NoLabel`, 31
- `style`, 31

`\SetUpVertex[<local options>]`, 31

`\SetVertexLabel`, 19

`\SetVertexLabelIn`, 4, 20

`\SetVertexLabelOut`, 4, 20

`\SetVertexMath`, 4, 19

`\SetVertexNoLabel`, 19

`\SetVertexNoMath`, 4

`\SetVertexNormal`, 24, 30

`\SetVertexNormal`: options

- `color`, 30
- `labelcolor`, 30
- `labelstyle`, 30
- `labeltext`, 30
- `label`, 30
- `lw`, 30
- `style`, 30

`\SetVertexNormal[<local options>]`, 30

`\SetVertexSimple`, 24, 28, 29
`\SetVertexSimple: options`
 `FillColor`, 29
 `LineColor`, 29
 `LineWidth`, 29
 `MinSize`, 29
 `Shape`, 29
`\SetVertexSimple[⟨local options⟩]`, 29
`\ShortCut`, 11
`\SO`, 4, 11
`\SOEA`, 4, 11
`\SOWE`, 4, 11

`\tikzset{VertexStyle/.append style = { ... } }`, 24
`\tikzset{VertexStyle/.style = { ... } }`, 24
`\tikzset`, 31

`\Vertex[a=⟨number⟩,d=⟨number⟩]{⟨vertex⟩}`, 10
`\Vertex[x=⟨number⟩,y=⟨number⟩]{⟨name⟩}`, 9
`\Vertex`, 4, 9, 11
`\Vertex: arguments`
 `Name`, 9
`\Vertex: options`
 `LabelOut`, 9
 `Ldist`, 9
 `Lpos`, 9
 `L`, 9
 `Math`, 9
 `NoLabel`, 9
 `Node`, 9
 `a`, 9
 `dir`, 9
 `d`, 9
 `empty`, 9
 `position`, 9
 `x`, 9
 `y`, 9
`\Vertex[⟨local options⟩]{⟨Name⟩}`, 9
`\Vertices`, 4, 6, 14
`\Vertices: arguments`
 `circle`, 14
 `line`, 14
 `square`, 14
 `tr1`, 14
 `tr2`, 14
 `tr3`, 14
 `tr4`, 14
`\Vertices: options`
 `dir`, 14
`\Vertices[⟨local options⟩]{⟨type⟩}{⟨List of vertices⟩}`, 14

`\WE`, 4, 11