

# La classe `neoschool`

Version : 1.1.2 — Auteur : Razik Ikhlef

<https://apps.edulatem.xyz>

[razik.ikhlef@csilyon.fr](mailto:razik.ikhlef@csilyon.fr)

## RÉSUMÉ

La classe `neoschool` propose aux enseignants du secondaire un ensemble d'outils pour concevoir leurs documents pédagogiques (évaluations, supports de cours, fiches d'exercices avec leur correction, etc). Pour répondre aux différents besoins, elle propose trente-six thèmes de couleurs prédéfinis, des options de classe variées pour la mise en page et la typographie, des environnements spécialisés, des commandes dédiées et différents styles d'en-têtes préformatés pour chaque type de document. Elle intègre plusieurs dizaines d'extensions LaTeX couramment utilisées (voir la liste ci-dessous), ce qui allège considérablement le préambule et évite autant que possible les incompatibilités. Multilingue, elle prend en charge le français, l'anglais et l'allemand.

## SOMMAIRE

<b>1</b>	<b>Extensions chargées</b>	<b>2</b>	3.1.3	Styles avec bulles . . . . .	7
			3.1.4	Autres styles de titre . . . . .	8
<b>2</b>	<b>Options de configuration</b>	<b>3</b>	<b>3.2</b>	<b>Configuration de l'en-tête</b>	
2.1	Langue . . . . .	3		( <code>\neoheader</code> ) . . . . .	8
2.2	Apparence globale . . . . .	3	<b>3.3</b>	<b>Options de formatage des titres</b> . .	8
2.2.1	Thèmes prédéfinis . . . . .	3	3.3.1	Styles globaux . . . . .	8
2.2.2	Modes de couleur . . . . .	4	3.3.2	Style du titre principal	
2.2.3	Personnalisation des couleurs .	4		( <code>\maketitle</code> ) . . . . .	9
2.2.4	Options d'affichage . . . . .	4	3.3.3	Style des sections ( <code>\section</code> )	9
2.2.5	Options d'arrondi des boîtes . .	4	3.3.4	Style des en-têtes et des pieds	
2.3	Personnalisation du résumé . . . .	5		de page . . . . .	9
2.4	Typographie . . . . .	5	3.3.5	Exemple complet de configuration	9
2.5	Mise en page . . . . .	5	<b>4</b>	<b>Disposition des contenus</b>	<b>10</b>
2.5.1	Marges et espacement . . . . .	5	4.1	Positionnement absolu d'objets . .	10
2.5.2	Options de sortie (mise en page		4.2	Mise en page à deux colonnes . . .	10
	multiple) . . . . .	6	4.3	Mise en page côte à côte (environne-	
2.5.3	Table des matières . . . . .	6		ment ( <code>sidebyside</code> ) . . . . .	10
2.5.4	En-têtes et pieds de page . . .	6	4.4	Association texte et image . . . . .	11
<b>3</b>	<b>Styles de document</b>	<b>6</b>	4.5	QR codes et contenus . . . . .	11
3.1	Styles de titre ( <code>\maketitle</code> ) . . .	6	4.6	Grilles et papiers . . . . .	12
3.1.1	Styles d'examen . . . . .	7	4.6.1	Grilles personnalisables . . . . .	12
3.1.2	Styles d'évaluation . . . . .	7	4.6.2	Remplissage automatique . . . .	13

4.6.3	Pages entières (styles de fond) . . . . .	13	7.1.2	Langages préconfigurés . . . . .	19
4.7	Boîtes simples ( <code>neobox</code> ) . . . . .	13	7.1.3	Commandes additionnelles . . . . .	19
<b>5</b>	<b>Exercices</b>	<b>14</b>	7.2	Autres options de classe pour le code	19
5.1	Environnements <code>exercice</code> et <code>solution</code> . . . . .	14	7.3	Option <code>minted</code> . . . . .	19
5.1.1	Configuration d'un exercice . . . . .	14	7.4	Algorithmes et pseudocode . . . . .	20
5.1.2	Options globales ( <code>\xsimsetup</code> ) . . . . .	14	<b>8</b>	<b>Notes et annotations</b>	<b>20</b>
5.1.3	Templates d'exercices (styles d'affichage) . . . . .	15	8.1	Notes marginales ( <code>todonotes</code> ) . . . . .	20
5.1.4	QCM et choix multiples . . . . .	16	8.2	Admonitions . . . . .	21
<b>6</b>	<b>Environnements mathématiques (Théorèmes, Définitions, etc.)</b>	<b>16</b>	<b>9</b>	<b>Notation et correction</b>	<b>21</b>
6.1	Styles de théorèmes (options de classe) . . . . .	16	9.1	Outils de notation . . . . .	21
6.2	Options communes des environnements . . . . .	17	9.2	Zones de réponse . . . . .	22
6.3	Options de numérotation (options de classe) . . . . .	17	9.3	Marqueurs et symboles . . . . .	22
6.4	Environnements mathématiques disponibles . . . . .	17	9.4	Évaluation par compétences . . . . .	22
<b>7</b>	<b>Code informatique</b>	<b>18</b>	<b>10</b>	<b>Commandes mathématiques et outils spéciaux</b>	<b>23</b>
7.1	Option <code>listings</code> . . . . .	18	10.1	Commandes mathématiques . . . . .	23
7.1.1	Environnement <code>code</code> . . . . .	18	10.1.1	Mise en valeur et coloration . . . . .	23
			10.1.2	Support APMEP . . . . .	23
			10.2	Outils spéciaux . . . . .	23
			10.2.1	Arbres et graphes . . . . .	23
			10.2.2	Grille mathématique ( <code>mathgrid</code> ) . . . . .	24

## ① EXTENSIONS CHARGÉES

Les extensions suivantes sont automatiquement chargées par la classe `neoschool`.

<code>adorn</code>	<code>forest</code>	<code>pdftexcmds</code>	<code>tcolorbox</code>
<code>adjustbox</code>	<code>iftex</code>	<code>pgffor</code>	<code>textcase</code>
<code>algpseudocode</code>	<code>ifthen</code>	<code>pgfplots</code>	<code>tikz</code>
<code>amsmath</code>	<code>kvoptions</code>	<code>pifont</code>	<code>tikzpagenodes</code>
<code>babel</code>	<code>lastpage</code>	<code>qrcode</code>	<code>tikzsymbols</code>
<code>bookmark</code>	<code>listings</code>	<code>scrlayer-scrpage</code>	<code>ulem</code>
<code>calc</code>	<code>marginnote</code>	<code>setspace</code>	<code>xcolor</code>
<code>changepage</code>	<code>microtype</code>	<code>silence</code>	<code>xhfill</code>
<code>cuted</code>	<code>minted</code>	<code>siunitx</code>	<code>xkeyval</code>
<code>environ</code>	<code>multicol</code>	<code>tabularray</code>	<code>xsim</code>
<code>fancyvrb</code>	<code>nccmath</code>	<code>tasks</code>	<code>xstring</code>
<code>fontawesome5</code>	<code>needspace</code>		

La compilation en `pdflatex` charge les extensions `fontenc` (avec l'option `T1`), `inputenc` (avec l'option `utf8`), `newpxtext` et `newpxmath`. La compilation en `lualatex` rend disponible les

extensions `fontspec`, `luacas`, `lua-ul`, `luacolor` et applique les polices `TeX Gyre PagellaX` et `TeX Gyre Heros` en plus de `newpxmath`. L'option **graphics** charge les extensions `graphicx` et `wrapfig`. L'option **faketest** (anciennement `draft`) charge les extensions `blindtext` et `lipsum`. L'option **mathastext** charge l'extension `mathastext`.

L'option **math** charge les extensions `annotate-equations`, `bm`, `breqn`, `cancel`, `mathrsfs`, `nccomma`, `numprint`, `tdsfrmth` (avec les options `suite` et `taupe`), `tkz-euclide`, `witharrows` et `xlop`. L'option **notes=longueur** permet d'afficher des notes encadrées (`todonotes`) dans les marges gauche et droite, de façon alternée, et définit leur largeur. Elle charge l'extension `todonotes`.

L'option **apmep**, qui permet de compiler directement les sujets d'annales de l'association du même nom, définit un ensemble de commandes mathématiques et charge les extensions `esvect`, `fourier-orns`, `numprint` (avec l'option `np`), `pstricks` (et de nombreux paquets de l'écosystème `pstricks`), `tabularx`, `textcomp`, `tkz-tab` et `enumitem`. L'option **mathics** permet de faire du calcul formel via `mathics` (version libre de `mathematica`) et charge les extensions `asymptote` et `latexalpha2`.

## ② OPTIONS DE CONFIGURATION

### 2.1 Langue

- **english, french, german** : active les traductions et conventions liées à chaque langue. Ces options affectent les intitulés des théorèmes et environnements, la typographie et les conventions mathématiques.
- **nofrenchlist** : désactive le style français des listes (remet des points à la place des tirets).
- **frenchlistaspar** : traite les listes comme des paragraphes en français (ajout de ponctuation finale et majuscule initiale).
- **frenchmath** : applique les conventions mathématiques françaises (majuscules et lettres grecques droites en maths).

### 2.2 Apparence globale

#### 2.2.1 Thèmes prédéfinis

L'option **theme** = `nom-du-theme` définit le thème de couleurs du document. Les thèmes disponibles sont : `classic` (thème par défaut), `abyss`, `aether`, `atlantic`, `autumn`, `blossom`, `botanical`, `burgundy`, `coral`, `cyprus`, `day`, `deepocean`, `duo`, `eagle`, `earth`, `foresthues`, `frost`, `glacier`, `goldensummer`, `graphite`, `heather`, `heritage`, `kassio`, `magma`, `modern`, `neon`, `nordic`, `purplebliss`, `retrocafe`, `saffronsky`, `scribe`, `sepia`, `summer`, `twilight`, `winkle` et `midnight` (thème sombre).

```
1 \documentclass[theme=retrocafe]{neoschool}
```

### 2.2.2 Modes de couleur

Ces options modifient la façon dont les couleurs du thème sont appliquées.

- **unicolor** : utilise une couleur unique (basée sur la couleur du titre) pour tous les environnements, avec variations par transparence.
- **examcolor** : Règle les couleurs des environnements pratiques (exercices, activités) et des admonitions (info, warning) sur la couleur théorique principale (`definitionColor`). Cela crée une palette plus sobre et bicolore, adaptée aux examens.
- **print** : convertit toutes les couleurs en noir et blanc pour l'impression.

### 2.2.3 Personnalisation des couleurs

Permet de surcharger les couleurs du thème choisi.

- **globalcolor** = `couleur` : couleur du texte principal.
- **titlecolor** = `couleur` ou **titlehexcolor** = `code hex` : couleur du titre principal.
- **headcolor** = `couleur` : couleur des titres de section (`\section`).
- **subcolor** = `couleur` : couleur des titres de sous-section (`\subsection`).
- **subsubcolor** = `couleur` : couleur des titres de sous-sous-section (`\subsubsection`).
- **headfootcolor** = `couleur` : couleur du texte des en-têtes et pieds de page.

### 2.2.4 Options d'affichage

- **nothmframe** : supprime les cadres autour des environnements (théorèmes, exercices...).
- **nothmback** : supprime les fonds colorés des environnements.
- **nocodeframe** : supprime les cadres autour des blocs de code.
- **nocodeback** : supprime les fonds colorés des blocs de code.
- **nocodenum** : supprime la numérotation des lignes des blocs de code.
- **noadframe** : (défaut) supprime les cadres autour des boîtes d'admonition.
- **noadback** : (défaut) supprime les fonds colorés des boîtes d'admonition.
- **adframe** : réactive l'affichage des cadres des boîtes d'admonition.
- **adback** : réactive l'affichage des fonds colorés des boîtes d'admonition.
- **scale** : harmonise la taille des polices lors de la compilation avec `lualatex`.
- **inlinecodebox** : affiche le code en ligne (`\texttt`) dans une boîte encadrée et colorée.

### 2.2.5 Options d'arrondi des boîtes

Ces options (de type `option=longueur`) permettent de personnaliser le rayon des coins arrondis pour différents types d'environnements.

- **boxarc** = `longueur` : arrondi pour `neobox` et `answerframe` (défaut : `1mm`).
- **codeboxarc** = `longueur` : arrondi pour les boîtes de code (défaut : `1mm`).
- **adboxarc** = `longueur` : arrondi pour les admonitions (défaut : `2pt`).
- **sideboxarc** = `longueur` : arrondi pour `sidebyside` (défaut : `1pt`).

- **thmboxarc** = `longueur` : arrondi pour les théorèmes (défaut : `1mm`).
- **exboxarc** = `longueur` : arrondi pour les exercices (défaut : `1mm`).
- **pseudoboxarc** = `longueur` : arrondi pour `pseudocode` (défaut : `1mm`).

## 2.3 Personnalisation du résumé

L'option `abstracttitle` permet de changer le titre de l'environnement `abstract`.

```
1 \documentclass[abstracttitle=Synthèse]{neoschool}
```

## 2.4 Typographie

- **sfbody** : utilise la police sans empattements (*sans serif*) pour le corps du texte.
- **sfall** : utilise la police sans empattements pour tout le document (titres, texte, etc.).
- **boldlistlabels** : met en gras les labels des listes (ex : **1.**, **a.**).
- **mathastext** : utilise la police de caractères du texte courant pour les mathématiques.
- **mainface** = `NomPolice` : définit la police principale (avec empattements).
- **mainfaceoptions** = `options` : options pour la police principale (ex : `Scale=MatchLowercase`).
- **sansface** = `NomPolice` : définit la police sans empattements.
- **sansfaceoptions** = `options` : options pour la police sans empattements.
- **monoface** = `NomPolice` : définit la police à chasse fixe (pour le code).
- **monofaceoptions** = `options` : options pour la police à chasse fixe.
- **mathface** = `NomPoliceMath` : définit la police mathématique.
- **mathfaceoptions** = `options` : options pour la police mathématique (`pdflatex` uniquement).
- **mathfacescale** = `facteur` : définit un facteur d'échelle pour la police mathématique (utile en compilation `lualatex` ou `xelatex`).
- **facefamily** = `NomFamille` : définit une famille complète de polices (ex : `fira`).
- **facefamilyoptions** = `options` : options pour la famille de polices.

```
1 % Exemple avec Fira Sans
2 \documentclass[
3   facefamily=Fira Sans,
4   facefamilyoptions={%
5     sfdefault,
6     lining
7   },
8   monoface=Fira Mono,
9   monofaceoptions={Scale=0.85}
10 ]{neoschool}
```

## 2.5 Mise en page

### 2.5.1 Marges et espacement

- **margin** = `longueur` : définit la largeur des marges horizontales (par défaut : `2cm`). Les marges verticales sont ajustées automatiquement.

- **notes** = `longueur` : active les notes marginales (`todonotes`) et définit leur largeur.
- **leftnotes**[=`longueur`] : active les notes marginales (`todonotes`) uniquement dans la marge de gauche et définit leur largeur.
- **rightnotes**[=`longueur`] : active les notes marginales (`todonotes`) uniquement dans la marge de droite et définit leur largeur.
- **noindent** : supprime l'indentation en début de paragraphe.
- **indent** = `longueur` : définit la taille de l'indentation (par défaut : `1em`).
- **vspace** = `facteur` : définit un facteur de compression/dilatation pour les espacements verticaux (paragraphe, listes, titres). Par défaut : `1.0`. Une valeur < 1 compresse, > 1 dilate.

### 2.5.2 Options de sortie (mise en page multiple)

Ces options permettent d'imprimer plusieurs pages logiques sur une seule page physique.

- **2a5toa4** : imprime 2 fois la même page A5 sur une feuille A4 paysage.
- **2a4toa3** : imprime 2 fois la même page A4 sur une feuille A3 paysage.
- **4a5toa3** : imprime 4 fois la même page A5 sur une feuille A3.
- **2toa3** : imprime 2 pages A4 différentes sur une feuille A3 paysage.
- **bookleta5** : crée un livret A5 (pages A5 sur feuilles A4 pliées).
- **bookleta4** : crée un livret A4 (pages A4 sur feuilles A3 pliées).

### 2.5.3 Table des matières

- **compacttoc** : réduit les espacements dans la table des matières.
- **monotoc** : la table des matières utilise la couleur du texte (`globalcolor`).
- **twocolumntoc** : affiche la table des matières sur deux colonnes.

### 2.5.4 En-têtes et pieds de page

- **fullheader** : active un en-tête et pied de page complets (type de document, titre, niveau / date, établissement, pagination). s'utilise conjointement avec `\neoheader`.
- **headrule** : ajoute un filet sous l'en-tête.
- **footrule** : ajoute un filet au-dessus du pied de page.
- **headfootrule** : ajoute les deux filets.
- **nofooter** : supprime complètement l'en-tête et le pied de page (style `empty`).
- **pageonlyfooter** : affiche uniquement le numéro de page centré en bas (style `pagenum`).

## ③ STYLES DE DOCUMENT

### 3.1 Styles de titre (`\maketitle`)

Ces options modifient radicalement l'apparence de la première page et du titre.

### 3.1.1 Styles d'examen

Idéal pour les devoirs surveillés, examens blancs.

- **exam** : style complet avec tableau pour nom, classe, date, etc., et bandeau de notation. Nécessite `\neoheader`.
- **shortexam** : style plus compact pour examens, avec informations essentielles. Nécessite `\neoheader`.
- **mockexam** : style spécifique pour épreuves type Bac/Brevet blanc, avec page de garde normalisée (France). Nécessite `\neoheader`.

```
1 \documentclass[exam]{neoschool}
2 \neoheader{
3   type = Devoir Surveillé n°1,
4   school = Lycée Imaginaire,
5   level = Terminale Spé Maths,
6   duration = 2h,
7   calculator = exam % ou true/false
8 }
9 \title{Suites numériques}
10 \date{21 octobre 2025}
11 \subject{Mathématiques} % Optionnel
12 \begin{document}
13 \maketitle
14 ...
15 \end{document}
```

### 3.1.2 Styles d'évaluation

Adaptés aux contrôles, interrogations rapides.

- **eval** : style standard, informations réparties dans les coins. Nécessite `\neoheader`.
- **evalicons** : style `eval` avec icônes personnalisables via `\neoheader`.
- **evalgrade** : style `eval` avec bandeau de notation ajouté automatiquement.
- **evaliconsgrade** : combine `evalicons` et `evalgrade`.
- **shorteval** : style compact sur une seule ligne en haut de page. Nécessite `\neoheader`.

### 3.1.3 Styles avec bulles

Effets visuels pour des documents moins formels.

- **bubbles** : titre centré sur fond de bulles colorées.
- **topbubbles** : bulles confinées en haut de la page, titre en dessous.
- **bottombubbles** : bulles confinées en bas de la page, titre au-dessus.

### 3.1.4 Autres styles de titre

Options plus classiques ou minimalistes.

- **titleornament** : ajoute des ornements sous le titre.
- **titlerule** : ajoute un petit filet sous le titre.
- **titlemidrule** : ajoute un filet centré de largeur moyenne sous le titre.
- **titlefullrule** : ajoute un filet sur toute la largeur sous le titre.
- **fancybox** : titre dans un bandeau coloré rotaté en haut à gauche.
- **onlytitleleft** / **onlytitle** / **onlytitleright** : affiche uniquement le titre, aligné à gauche / centré / à droite. Ignore auteur, date, etc.
- **shorttitle** : style compact, titre centré sur une ligne en haut de page.
- **shortlesson** : style compact pour fiches de cours, avec type, titre et niveau sur une ligne. Nécessite `\neoheader`.

### 3.2 Configuration de l'en-tête (`\neoheader`)

Cette commande configure les informations utilisées par les styles `exam`, `eval`, `mockexam`, `shortlesson` et l'option `fullheader`.

```
1 \neoheader{
2   type = {Interrogation de cours}, % Type de document
3   school = {Collège A. Turing}, % Nom de l'établissement
4   academy = {Lyon}, % Nom de l'académie (pour mockexam)
5   level = {Quatrième}, % Niveau de classe
6   duration = {20 minutes}, % Durée (pour exam, mockexam)
7   calculator = {false}, % Calculatrice: true / false / exam
8   leftcontent = {\faFlask}, % Icône gauche (pour evalicons)
9   rightcontent = {\faCalculator} % Icône droite (pour evalicons)
10  leftcontentfill = {true}, % (pour evalicons, booléen)
11  rightcontentfill = {true} % (pour evalicons, booléen)
12 }
```

### 3.3 Options de formatage des titres

Ces options contrôlent l'apparence (police, graisse, forme, alignement) des différents titres du document.

#### 3.3.1 Styles globaux

S'appliquent par défaut à tous les titres (sections, théorèmes, exercices...).

- **headstyle** = `style` (`sffamily` par défaut) : famille de police (ex : `rmfamily`, `sffamily`).
- **headweight** = `graisse` (`bfseries` par défaut) : graisse (ex : `mdseries`, `bfseries`, `sbseries` pour semi-gras si disponible).
- **headshape** = `forme` (`scshape` par défaut) : forme (ex : `upshape`, `itshape`, `scshape` pour petites capitales).



### 3.3.2 Style du titre principal ( `\maketitle` )

Surcharge les styles globaux pour le titre principal.

- **titlestyle** = `style` (hérite de `headstyle`).
- **titleweight** = `graisse` (hérite de `headweight`).
- **titleshape** = `forme` (`upshape` par défaut).
- **titlealign** = `alignement` (`center` par défaut) : `left`, `center`, `right`.

### 3.3.3 Style des sections ( `\section` )

Options spécifiques pour les titres de section.

- **sectionnumstyle** = `style` (`circle` par défaut) : apparence du numéro (`circle`, `box`, `dash`, `plain`).
- **sectiontextstyle** = `style` (`sc` par défaut) : casse du texte (`sc`, `upper`, `lower`).
- **sectionstyle** = `style` (`normal` par défaut) : style visuel global (`ornaments`, `underline`, `normal`, `highlighted`, `shadedline`).
- **sectionalign** = `alignement` (`center` par défaut) : `left`, `center`, `right`.

### 3.3.4 Style des en-têtes et des pieds de page

- **headfootstyle** = `style` (hérite de `titlestyle`) : style de police pour le texte des en-têtes et pieds de page.

### 3.3.5 Exemple complet de configuration

```
1 \documentclass[
2     % Style global
3     headstyle=sffamily,
4     headweight=bfseries,
5     headshape=scshape,
6     % Titre principal
7     titlestyle=rmfamily,
8     titleweight=bfseries,
9     titleshape=upshape,
10    titlealign=left,
11    % Sections
12    sectionnumstyle=box,
13    sectiontextstyle=upper,
14    sectionstyle=underline,
15    sectionalign=left,
16    % En-têtes/pieds de page
17    headfootstyle=sffamily
18 ]{neoschool}
```

## ④ DISPOSITION DES CONTENUS

### 4.1 Positionnement absolu d'objets

La commande `\positionobject{x}{y}{scale}{contenu}` place *contenu* aux coordonnées (x, y) depuis le coin supérieur gauche de la page, avec un facteur d'échelle.

```
1 % Logo en haut à droite
2 \positionobject{15cm}{1cm}{0.5}{%
3   \includegraphics[width=3cm]{logo.png}%
4 }
5
6 % Texte en bas à gauche
7 \positionobject{2cm}{25cm}{1.2}{%
8   \textit{Note importante}%
9 }
```

### 4.2 Mise en page à deux colonnes

La commande `\splitcontent[w1][gap]{col1}{col2}` divise l'espace horizontal.

- *w1* : largeur de la première colonne (défaut : 0.5 pour 50 %).
- *gap* : espace entre les colonnes (défaut : 0.02 pour 2 %).
- *col1, col2* : contenu des colonnes.

```
\splitcontent[0.45][0.03]{%
  Contenu colonne 1 (40 %)
  \lipsum[1][1-2]
}{%
  Contenu colonne 2 (52 %)
  \lipsum[1][1-2]
}
```

Contenu colonne 1 (45 %)	Contenu colonne 2 (52 %)
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.	Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo.

### 4.3 Mise en page côte à côte (environnement `sidebyside`)

Crée deux boîtes `tcolorbox` côte à côte. Utiliser `\tcblower` pour passer de la boîte gauche à la droite. Accepte les options de `tcolorbox`.

```

\begin{sidebyside}[
  title=\centering
  Comparaison,
]
  Avantages
  \begin{itemize}
    \item Point 1
  \end{itemize}
  \tblower % Sépare les
    deux colonnes
  Inconvénients
  \begin{itemize}
    \item Point A
  \end{itemize}
\end{sidebyside}

```

Comparaison	
Avantages — Point 1	Inconvénients — Point A

## 4.4 Association texte et image

La commande `\textwithimage[*]{w_img}{s_img}{texte}{chemin_img}` combine du texte et une image.

- `*` : si présent, met l'image à gauche (droite par défaut).
- `w_img` : largeur relative de l'image (ex : 0.3 pour 30 %).
- `s_img` : échelle de l'image dans sa boîte (ex : 0.95).
- `texte` : le texte.
- `chemin_img` : chemin vers le fichier image.

```

% Image à droite (30 % de la largeur
)
\textwithimage{0.3}{1}{%
  Description de l'image qui
  sera affichée à droite.
}{example-image-a} % image fictive

```

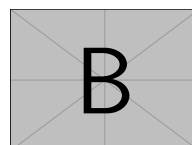
Description de l'image qui  
sera affichée à droite.



```

% Image à gauche (40 % de la largeur
)
\textwithimage*{0.4}{0.9}{%
  Description à droite de l'image.
}{example-image-b} % image fictive

```



Description à droite de  
l'image.

## 4.5 QR codes et contenus

La commande `\withqrcode[*]{taille}{url}{contenu}` intègre un QR code à côté d'un contenu.

- `*` : place le QR code à droite (gauche par défaut).
- `taille` : taille du QR code (défaut : 2cm).
- `url` : url ou texte à encoder dans le QR code.
- `contenu` : texte ou autre contenu à afficher à côté.

```
% QR code à gauche (défaut 2cm)
\withqrcode{https://www.google.com}{
  Scannez pour visiter Google.
}
```



Scannez pour visiter Google.

```
% QR code à droite (3cm)
\withqrcode*[3cm]{https://fr.
  wikipedia.org}{
  Plus d'informations sur Wikipedia.
}
```

Plus d'informations sur Wikipedia.



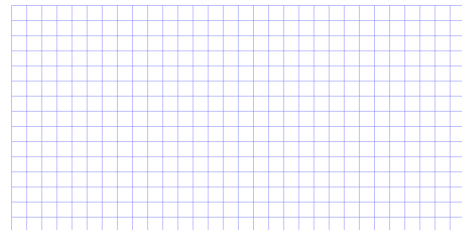
## 4.6 Grilles et papiers

### 4.6.1 Grilles personnalisables

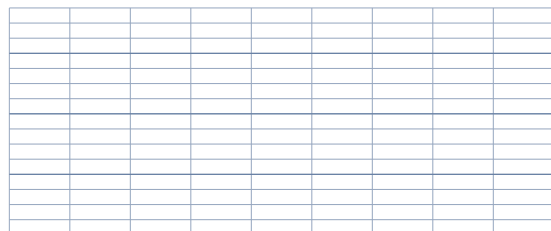
Commandes pour dessiner des zones quadrillées.

- `\grid[couleur]{largeur}{hauteur}` : grille à petits carreaux (5mm x 5mm).
- `\customgrid[couleur][dx][dy]{largeur}{hauteur}` : grille avec espacement personnalisé (dx, dy).
- `\frenchgrid[coul_p][coul_g]{largeur}{hauteur}` : grille à grands carreaux de type Seyès. La version étoilée `\frenchgrid*` centre la grille horizontalement.

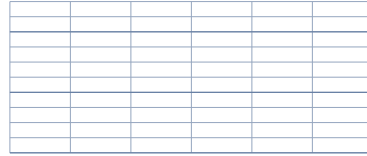
```
\customgrid[blue!50][2mm][2mm]{6cm}
{3cm}
```



```
\frenchgrid{7.5cm}{3cm}
```



```
\frenchgrid*{5cm}{2cm} % centrée
```



#### 4.6.2 Remplissage automatique

La commande `\gridfill` remplit l'espace vertical restant sur la page avec une grille `customgrid` (5mm par défaut). La version étoilée `\gridfill*` utilise `frenchgrid`.

```
1 % Syntaxe : \gridfill*[couleur][dx][dy]
2
3 % Remplit avec une grille customgrid bleue 2mm x 2mm
4 \gridfill[blue][2mm][2mm]
5
6 % Remplit avec une grille frenchgrid (par défaut)
7 \gridfill*
```

#### 4.6.3 Pages entières (styles de fond)

Applique un style de fond à la page courante.

- `\notebook` : style cahier ligné avec marge rouge.
- `\nbminorgrid` : fond quadrillé petits carreaux (5mm).
- `\nbmajorgrid` : fond quadrillé Seyès.

```
1 \nbminorgrid % Applique le quadrillage fin à toute la page
```

#### 4.7 Boîtes simples ( `neobox` )

Environnement pour créer des boîtes `tcolorbox` simples, avec ou sans cadre. Accepte les options de `tcolorbox`.

- `neobox` : boîte avec cadre.
- `neobox*` : boîte sans cadre visible (mais fond possible).

```
\begin{neobox}[
  title=Boîte standard
]
Contenu...
\end{neobox}
```

Boîte standard

Contenu...

```
\begin{neobox*}
Contenu sans cadre visible...
\end{neobox*}
```

Contenu sans cadre visible...

## ⑤ EXERCICES

Basé sur l'extension `xsim`.

### 5.1 Environnements `exercice` et `solution`

#### 5.1.1 Configuration d'un exercice

Options disponibles dans `\begin{exercice}[options]`.

- **points** = `nombre` : nombre de points.
- **bonus-points** = `nombre` : points bonus.
- **level** = `nombre` : niveau de difficulté (1 à 5, affiché en étoiles ★).
- **subtitle** = `texte` : sous-titre ou description brève.
- **icon** = `nom-icone-fa` : icône FontAwesome (ex : `pencil-alt`). Nécessite l'option de classe `exerciceicons`.
- **topic** = `thème` : thème/chapitre (pour tri/sélection future).
- **grade** = `classe/niveau` : niveau de classe (pour tri/sélection).
- **subject** = `matière` : matière (pour tri/sélection).
- **ID** = `identifiant` : identifiant unique pour référencer l'exercice (`\exercisenum{id}`).
- **template** = `nom-template` : style d'affichage spécifique pour cet exercice (voir liste ci-dessous). Surcharge le style global.

```
\begin{exercice}[
  points=3,
  bonus-points=1,
  level=2,
  subtitle={Application},
  ID=calc01,
  template=sober-box
]
  Calculer $3 \times (5+2)$.
\end{exercice}
```

```
\begin{solution}
  $3 \times (5+2) = 3 \times 7 = 21$.
\end{solution}
```

```
\end{solution}
```

```
Voir exercice \exercisenum{calc
01}.
```

**EXERCICE 1 · [★★] · Application · (3 points)** Calculer  $3 \times (5 + 2)$ .

**CORRIGÉ DE L'EXERCICE 1**  $3 \times (5 + 2) = 3 \times 7 = 21$ .

Voir exercice 1.

#### 5.1.2 Options globales (`\xsimsetup`)

À placer dans le préambule pour affecter tout le document.

- **exercice/template** = `nom-template` : définit le style par défaut pour tous les exercices.

- **solution/template** = `nom-template-sol` : définit le style par défaut pour toutes les solutions (ex : `sol-inline`, `sol-block`, `sol-dotted`).
- **solution/print** = `true/false` : affiche ou masque globalement les solutions. Peut être surchargé par l'option de classe **answers**.

Autres options de classe affectant les exercices :

- **exercisestyle** = `nom-template` : définit le style d'affichage (template) par défaut pour tous les exercices (ex : `inline`, `box-hrule`, `elegant-box`). Si non spécifié, il utilise le style défini par `theoremstyle`.
- **exerciseicons** : active l'affichage des icônes pour tous les exercices.
- **answers** / **answersonly** : affiche les solutions après chaque exercice / Affiche uniquement les solutions.
- **shuffle** : mélange aléatoirement les réponses dans les QCM (`choices`, `checkboxes`).
- **sectionthmcounter** : numérote les exercices (et théorèmes) par section (ex : Ex 1.1, Ex 1.2, Ex 2.1...).
- **sharedexcounter** : partage le même compteur entre exercices et théorèmes (combiné avec `sharedthmcounter`).
- **blocksol** : utilise le template `sol-block` pour les solutions.

### 5.1.3 Templates d'exercices (styles d'affichage)

Valeurs possibles pour `template` (option d'exercice ou `xsimsetup`).

- **box**, **elegant-box**, **shaded-box**, **slanted-box**, **sober-box**, **classic-box**, **classy-box** : différents styles de boîtes encadrées avec titre.
- **rect-box**, **rect-box-out** : boîtes rectangulaires simples, avec ou sans contour.
- **num-box**, **num-box-out** : boîtes compactes affichant juste le numéro.
- **ex-num-box**, **ex-num-box-out** : comme `num-box` mais avec "Ex." avant le numéro.
- **box-hrule**, **box-hrule-out**, **box-hrule-in** : titre dans une boîte suivi d'une ligne horizontale.
- **boxed**, **boxed-out** : titre dans une petite boîte (remplie ou contour).
- **inline** (défaut si `amslikethm`) : style simple en ligne, comme un paragraphe.
- **section** : style comme un titre de section.
- **terminal** : style imitant une console, avec icône `>_`.
- **block** : style minimaliste sans boîte ni décoration.
- **hrule** : style avec une simple ligne horizontale sous le titre.
- **smart-box** : boîte simple avec titre coloré.
- **minimal-ams** : style minimaliste, titre en gras sans boîte.
- **rule-ams** : style minimaliste avec titre et ligne horizontale.

```
\xsimsetup{exercise/template=box-
  hrule-out}
\begin{exercise}[points=2]
  Style box-hrule-out.
\end{exercise}
\xsimsetup{exercise/template=inline}
% Retour au défaut
```

**EXERCICE 2**

Style box-hrule-out.

←←← 2 POINTS

### 5.1.4 QCM et choix multiples

Environnements à utiliser à l'intérieur d'un exercice.

- **choices** (*n*) : pour QCM à réponse unique. `\choice` pour une réponse, `\choice[\correct]` pour la bonne réponse. *n* est le nombre de colonnes.
- **checkboxes** (*n*) : pour QCM à réponses multiples. `\checkbox` pour une réponse, `\checkbox[\correct*]` pour une bonne réponse. *n* est le nombre de colonnes.

```
\begin{exercise}[ID=qcm-deriv,
  points=1]
  Quelle est la dérivée de  $f(x)=x^2$ 
  ?
  \begin{choices}(2) % Sur 2 colonnes
    \choice[\correct]  $x \mapsto 2x$ 
    \choice  $x \mapsto x$ 
    \choice  $x \mapsto x^2$ 
    \choice  $x \mapsto 2$ 
  \end{choices}
\end{exercise}
```

**EXERCICE 3 · (1 POINT)** Quelle est la dérivée de  $f(x) = x^2$  ?

- ☐  $x \mapsto 2x$ 
☐  $x \mapsto x$   
☐  $x \mapsto x^2$ 
☐  $x \mapsto 2$

```
\begin{exercise}[ID=qcm-props, points
=2]
  Cochez les propriétés vraies.
  \begin{checkboxes}(1) % Sur 1 colonne
    \checkbox[\correct*]  $1+1=2$ 
    \checkbox  $\sqrt{2}$  est rationnel.
    \checkbox[\correct*]  $0 \times 5 = 0$ 
  \end{checkboxes}
\end{exercise}
```

**EXERCICE 4 · (2 POINTS)** Cochez les propriétés vraies.

- ☐  $1 + 1 = 2$   
☐  $\sqrt{2}$  est rationnel.  
☐  $0 \times 5 = 0$

## ⑥ ENVIRONNEMENTS MATHÉMATIQUES (THÉORÈMES, DÉFINITIONS, ETC.)

Basés sur `tcolorbox`.

### 6.1 Styles de théorèmes (options de classe)

Ces options de classe définissent l'apparence par défaut de tous les environnements de type théorème.

- **classythm**, **soberthm**, **elegantthm**, **classicthm**, **slantedthm**, **shadedthm**, **boxedthm** : Styles variés avec boîtes, couleurs, etc.
- **amslikethm** (défaut) : Style minimaliste similaire aux environnements standard de l'AMS.
- **theoremstyle** = `nom-style` : permet de définir le style de théorème en utilisant son nom (ex : `elegant`, `amslike`, `shaded`). C'est une alternative à l'utilisation des options booléennes ci-dessus.



```

1 % Applique le style \texttt{elegantthm} à tous les environnements mathématiques.
2 \documentclass[elegantthm]{neoschool}

```

## 6.2 Options communes des environnements

Passées entre crochets : `\begin{theorem}[options]`.

- **title** = `texte` : titre personnalisé (ex : `title=Théorème de Pythagore`).
- **label** = `nom` : étiquette pour référencement (`\ref`, `\nameref`).
- **colback** = `couleur` : couleur de fond spécifique.
- **colframe** = `couleur` : couleur du cadre spécifique.
- **coltitle** = `couleur` : couleur du titre spécifique.
- **fonttitle** = `commandes` : commandes de formatage pour le titre (ex : `fonttitle=\sffamily\bfseries`).

## 6.3 Options de numérotation (options de classe)

- **sectionthmcounter** : les compteurs sont relatifs à chaque section (ex : Thm 1.1, Def 1.2, Thm 2.1...).
- **sharedthmcounter** : un seul compteur est partagé entre tous les environnements (Thm 1, Def 2, Lem 3...).
- **thmgrouppcounter** : regroupe les environnements (théorème, lemme, corollaire, proposition, propriété) sous un même compteur partagé.

## 6.4 Environnements mathématiques disponibles

- **theorem** (référence : `thm`)
- **lemma** (référence : `lem`)
- **corollary** (référence : `cor`)
- **conjecture** (référence : `conj`)
- **proposition** (référence : `prop`)
- **property** (référence : `prop`)
- **definition** (référence : `def`)
- **method** (référence : `meth`)
- **activity** (référence : `act`)
- **application** (référence : `appl`)
- **remark** (non numéroté)
- **remarks** (non numéroté, pour plusieurs remarques)
- **example** (référence : `ex`)
- **examples** (non numéroté, pour plusieurs exemples)
- **proof** : environnement pour les démonstrations.

```
\begin{definition}[
  title=Nombre premier,
  label=premier
]
Un entier naturel est dit premier
s'il admet exactement deux diviseurs
distincts : 1 et lui-même.
\end{definition}

Voir la définition~\ref{def:premier}.
```

**DÉFINITION 1 (NOMBRE PREMIER)** Un entier naturel est dit premier s'il admet exactement deux diviseurs distincts : 1 et lui-même.

Voir la définition 1.

```
\begin{proof}
  La preuve est triviale et laissée
  en exercice au lecteur.
\end{proof}
```

*Démonstration.* La preuve est triviale et laissée en exercice au lecteur.  $\square$

## ⑦ CODE INFORMATIQUE

Deux systèmes sont disponibles via les options de classe **listings** (défaut) ou **minted**.

### 7.1 Option `listings`

Utilise l'extension `listings`. La coloration syntaxique est définie par le thème de couleurs choisi.

#### 7.1.1 Environnement `code`

Affiche un bloc de code.

```
\begin[options]{code}{langage}[titre][style-boite]
... code ...
\end{code}
```

- `options` : options de `listings` (ex : `numbers=none`, `frame=single`).
- `langage` : langage du code (ex : `python`, `latex`, `c++` ...).
- `titre` : titre optionnel pour la boîte.
- `style-boite` : style de la boîte `tcolorbox` (voir ci-dessous).
- `code*` : variante sans numérotation automatique du listing.

L'option de classe `codestyle=[style]` (par défaut `box-elegant`) permet de définir le style de boîte pour l'ensemble du document. Le paramètre `[style-boite]` de l'environnement permet de surcharger ce choix localement. Styles de boîtes disponibles : `box-minimal`, `box-subtle`, `box-fancy`, `box-elegant`, `box-sober`, `box-academic`, `box-diagonal`, `box-bevel`, `box-corner`, `box-rounded`, `box-downhill`, `box-toptitle`, `box-bottomtitle`.

```
\begin{code}{python}[Fonction Carre
][box-minimal]
def square(x):
    """Calcule le carré de x."""
    return x * x
\end{code}
```

#### PROGRAMME 28 — FONCTION CARRE

```
def square(x):
    """Calcule le carré de x."""
    return x * x
```

### 7.1.2 Langages préconfigurés

Python, Java, C++, JavaScript, SQL, LaTeX, Bash, Assembly, Lisp, JSON, YAML, TOML, CSV, Markdown.

### 7.1.3 Commandes additionnelles

- `\codeinline[lang]{code}` : affiche du code en ligne avec coloration (*lang* optionnel).
- `\codeinput[opt]{lang}{fichier}[titre][style]` : importe et affiche le code depuis un fichier.

Le code `\codeinline[python]{x = 5}` initialise x.

Le code `x = 5` initialise x.

## 7.2 Autres options de classe pour le code

- **codewidth** = `facteur` : (défaut : `1.0`) définit la largeur des blocs de code en tant que facteur de `\linewidth`.
- **centeredcode** : option booléenne pour centrer les blocs de code sur la page.

### 7.3 Option `minted`

Utilise l'extension `minted`, nécessitant Python et Pygments installés, ainsi qu'une compilation avec l'option `-shell-escape`. Offre une coloration syntaxique plus riche.

L'environnement `code`, `code*`, `\codeinline` et `\codeinput` fonctionnent de manière similaire, mais les options passées sont celles de `minted` (ex : `linenos`, `highlightlines={2,3}`, `style=tango`). Les styles de boîtes restent les mêmes.

```

\documentclass[minted]{neoschool}
...
\usemintedstyle{friendly}
\begin{code}[linenos]{python}[
    Exemple Minted]
import math

def circle_area(radius):
    return math.pi * radius**2
\end{code}

```

#### PROGRAMME 30 — EXEMPLE MINTED

```

1 import math
2
3 def circle_area(radius):
4     return math.pi * radius**2

```

## 7.4 Algorithmes et pseudocode

La classe fournit un environnement `pseudocode` basé sur `algpseudocode` et `tcolorbox` pour afficher des algorithmes. Il est automatiquement numéroté et titré.

```

\begin{pseudocode}{Algorithme
    Exemple}
\Require $n \ge 0$
\Ensure $y = x^n$
\State $y \leftarrow 1$
\If{$n < 0$}
    \State $X \leftarrow 1/x$
    \State $N \leftarrow -n$
\Else
    \State $X \leftarrow x$
    \State $N \leftarrow n$
\EndIf
\While{$N \neq 0$}
    \If{$N$ is even}
        \State $X \leftarrow X \times X$
        \State $N \leftarrow N/2$
    \Else[$N$ is odd]
        \State $y \leftarrow y \times X$
        \State $N \leftarrow N-1$
    \EndIf
\EndWhile
\end{pseudocode}

```

#### Algorithme 1 — Algorithme Exemple

```

entrée  $n \geq 0$ 
sortie  $y = x^n$ 
 $y \leftarrow 1$ 
si  $n < 0$  alors
     $X \leftarrow 1/x$ 
     $N \leftarrow -n$ 
sinon
     $X \leftarrow x$ 
     $N \leftarrow n$ 
fin si
tant que  $N \neq 0$  faire
    si  $N$  is even alors
         $X \leftarrow X \times X$ 
         $N \leftarrow N/2$ 
    sinon [ $N$  is odd]
         $y \leftarrow y \times X$ 
         $N \leftarrow N - 1$ 
    fin si
fin tant que

```

## ⑧ NOTES ET ANNOTATIONS

### 8.1 Notes marginales ( `todonotes` )

Activées par l'option de classe `notes=longueur`.

- `\tdnote[options]{texte}` : Crée une note dans la marge. Les notes alternent côté gauche/droit. Les options (`backgroundcolor`, `linecolor`, etc.) sont celles de `todonotes`.

- `\boxnote[label]{texte}` + `\tdmark[label]` : Permet de placer une `tdnote` (`boxnote`) ancrée à un endroit précis (`tdmark`) même à l'intérieur d'environnements où `todonotes` échoue. Le *label* doit être identique.

```

1 Ceci est un texte\tdnote{Point important!}.
2 \boxnote[thm-imp]{Ne pas oublier cette hypothèse.}
3 \begin{theorem}
4   \tdmark[thm-imp] Si  $x > 0$ , alors...
5 \end{theorem}

```

## 8.2 Admonitions

Boîtes colorées avec icônes pour attirer l'attention sur certains points.

- `note` : remarques générales (📝).
- `info` : informations complémentaires (ℹ️).
- `warning` : avertissements (⚠️).
- **important** : points essentiels (❗).
- **tip** : conseils, astuces (💡).
- **reminder** : à retenir, mémoriser (📌).
- **summary** : résumé, en bref (📄).
- **toolbox** : liste de matériel, prérequis (🔧).

Chaque environnement accepte `[Titre optionnel][Icône optionnelle]`. L'option de classe **inlinenadmonition** (activée par défaut) affiche le titre et le contenu sur la même ligne. Pour désactiver ce comportement et forcer le contenu à s'afficher sous le titre, utilisez l'option **blockadmonition**.

```

\begin{warning}[Attention][\faSkull]
  Ne pas diviser par zéro !
\end{warning}

```

💀 **ATTENTION** Ne pas diviser par zéro!

```

\begin{tip}
  Penser à factoriser d'abord.
\end{tip}

```

💡 **CONSEIL** Penser à factoriser d'abord.

## ⑨ NOTATION ET CORRECTION

### 9.1 Outils de notation

- `\gradingstrip[total]` : affiche un bandeau pour la note et l'appréciation. *total* optionnel surcharge le total défini par l'option de classe `totalpoints` (défaut : 20).
- `\mrk[*][comm]{pts}` : affiche (pts pt) ou (pts pts) dans la marge. `*` place à gauche, *comm* ajoute un commentaire.

<code>\gradingstrip[10] % Barème sur 10</code>  Question 1 <code>\mrk*</code> [Détail du barème ici... lorem ipsum dolor sit amet]{3}	<table> <tr> <th>Note</th><th>Appréciation</th></tr> <tr> <td>10</td><td></td></tr> </table> Question 1	Note	Appréciation	10	
Note	Appréciation				
10					

## 9.2 Zones de réponse

- `\answerfield[largeur]{lignes}` : crée une zone de réponse avec fond coloré, d'une hauteur de *lignes* lignes de texte et d'une *largeur* (défaut : `0.975\linewidth`).
- `\answerframe[largeur]{lignes}[options]` : crée une zone de réponse encadrée, d'une hauteur de *lignes* et d'une *largeur* (défaut : `\linewidth`).
- `\vardots[longueur]` : dessine une ligne pointillée de *longueur* (défaut : `\linewidth`).

Réponse : <code>\answerfield[6cm]{1}</code>	Réponse :
---	-----------

Réponse : <code>\answerframe{1}</code>	Réponse :
--	-----------

Signature : <code>\vardots[4cm]</code>	Signature : .....
--	-------------------

## 9.3 Marqueurs et symboles





- `\cmark` : ✓ (vert).
- `\xmark` : ✗ (rouge).
- `\unchecked` : □ (pour listes).
- `\done` : ✓ (pour listes).
- `\wontfix` : ✗ (pour listes).

<code>\begin{itemize}</code> <code>\unchecked</code> Tâche 1 <code>\done</code> Tâche 2 <code>\wontfix</code> Tâche 3 <code>\end{itemize}</code>	<input type="checkbox"/> Tâche 1 <input checked="" type="checkbox"/> Tâche 2 <input checked="" type="checkbox"/> Tâche 3
--	--

## 9.4 Évaluation par compétences

`\competencies{Comp1\Comp2...}` : crée un tableau pour évaluer des compétences selon 4 niveaux de maîtrise (avec emojis).

```
\competencies{
  Maîtriser les fractions
  \\
  Résoudre une équation du
  1\up{er} degré \\
  Calculer une dérivée
}
```

Compétences				
Maîtriser les fractions				
Résoudre une équation du 1 <sup>er</sup> degré				
Calculer une dérivée				

## 10 COMMANDES MATHÉMATIQUES ET OUTILS SPÉCIAUX

### 10.1 Commandes mathématiques

#### 10.1.1 Mise en valeur et coloration

- `\mhl[couleur]{expr}` : surligne l'expression mathématique *expr* avec la *couleur* (défaut : couleur spécifique du thème).
- `\mc[couleur]{expr}` : colore l'expression mathématique *expr* avec la *couleur* (défaut : couleur spécifique du thème).
- `\mathbox<fond>[bordure]{contenu}` : encadre rapidement du *contenu* mathématique dans une boîte `tcolorbox`. La couleur de *fond* (défaut : blanc) et de *bordure* (défaut : couleur spécifique du thème) sont personnalisables.

$f(x) = \text{\mhl[cyan!20]{x^2} + \mc[blue]{3x} - 1}$

$$f(x) = x^2 + 3x - 1$$

`$\mathbox{E=mc^2}$` % Fond blanc, bordure theme

$$E = mc^2$$

`$\mathbox<yellow!20>[red]{a^2+b^2=c^2}$` % Perso

$$a^2 + b^2 = c^2$$

#### 10.1.2 Support APMEP

Commandes disponibles avec l'option de classe `apmep`.

- Vecteurs : `\vectt{AB}`.
- Repères : `\Oij`, `\Oijk`, `\Ouv`.
- Symboles/Commandes : `\euro` (€), `\cg` (⌋), `\cd` (⌈), `\pg` (≥), `\pp` (≤), `\barre{x}` ( $\bar{x}$ ).

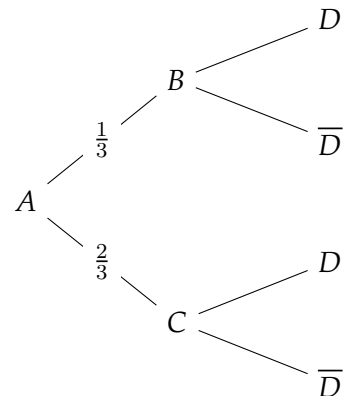
### 10.2 Outils spéciaux

#### 10.2.1 Arbres et graphes

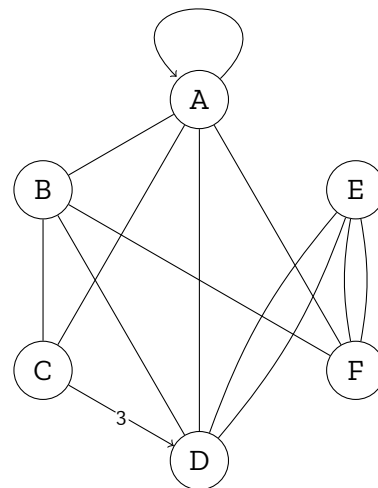
- **Arbres** (environnement `neotree`) : utilise `forest`. La syntaxe est celle de `forest`, avec des options passées à l'environnement. L'option `w=val` ajoute un poids sur une branche.

- **Graphes** (commande `\neograph`, `lualatex` requis) : utilise TikZ `graphs`. La syntaxe est celle de TikZ pour les graphes.

```
\begin{neotree}[l=2cm, s sep=1cm]
  A [B, w=\frac{1}{3} [D] [\overline{D}]]
    [C, w=\frac{2}{3} [D] [\overline{D}]]
\end{neotree}
```



```
% Nécessite lualatex
\neograph{
  A -- {B, C, D, F},
  B -- {C, D, F},
  C -> ["3"] D,
  D -- [bend left=10] {E},
  E -- [bend left=10] {D},
  E -- [bend left=10] {F},
  F -- [bend left=10] {E},
  A -- [loop] A
}
```



### 10.2.2 Grille mathématique (`mathgrid`)

Environnement pour aligner des blocs d'équations (`align*`) en grille.

- `\begin{mathgrid}{n}` : démarre une grille à  $n$  colonnes.
- `\neoline` : commence une nouvelle ligne dans la grille.
- `\neocol[span]{contenu}` : ajoute une colonne contenant *contenu* (un bloc `align*`). *span* (optionnel, défaut 1) indique sur combien de colonnes s'étend le contenu.



```

\begin{mathgrid}{2} % 2
  colonnes
  \neoline % Ligne 1
  \neocol{} % Col 1
  A &= 1+1 \\ A &= 2
}
  \neocol{} % Col 2
  B &= 3 \times 4 \\ B &= 12
}
  \neoline % Ligne 2
  \neocol[2]{} % Colonne unique (
    span=2)
  C &= A+B \\ C &= 2+12 \\ C &=
    14
}
\end{mathgrid}

```

$$\begin{array}{ll}
 A = 1 + 1 & B = 3 \times 4 \\
 A = 2 & B = 12 \\
 C = A + B & \\
 C = 2 + 12 & \\
 C = 14 &
 \end{array}$$