

Alain Matthes

tkz-base.sty v2.1

AlterMundus

tkz-base.sty est un module (package) pour créer à l'aide de Tikz des graphiques le plus simplement possible. Il dépend de Tikz et est la base sur laquelle sera construite 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 possible de compiler avec pdflatex ainsi qu'avec latex, mais dans ce dernier cas, il est nécessaire de passer de PS à PDF en utilisant ps2pdf14 pour des problèmes de transparence.

doc-tkz-base v2.1a 14 07 2007

☞ Les zones de texte en orange sont des liens directs vers mon site <http://www.altermundus.fr>. Les liens en rouge concernent les débuts de chapitre, ceux en vert les exemples.

☞ Je remercie **Till Tantau** pour nous permettre d'utiliser tikz/pgf.

☞ Je remercie **Michel Bovani** pour nous permettre d'utiliser fourier et utopia avec L^AT_EX.

☞ je remercie également **Jean-Côme Charpentier, Josselin Noirel** pour les différentes idées qui m'ont permis de faire ce package.



Cette version apporte des améliorations par rapport à la version précédente mais aussi un changement de syntaxe. Il est conseillé de bien regarder la documentation.



Attention à ne pas mettre d'espace entre les arguments. Cette version se stabilise mais les remarques pour améliorer le package sont acceptées. Le package n'est pas garanti sans bug, je vous remercie pour toute découverte d'anomalie(s).



Vous remarquerez que Tikz dans sa version 1.18 est de plus en plus allergique aux caractères actifs, aussi j'ai créé deux macros `\twoptoff` et `\twopton` pour désactiver et activer « : ».



Sommaire

I. Installation	page 7
II. Compilation des exemples	page 8
III. Quelques exemples	page 9
ex. n° 1 Recette en fonction du mois	page 9
ex. n° 2 Recette moyenne en fonction de l'année	page 10
ex. n° 3 Point sur le cercle trigonométrique avec FP.sty	page 11
ex. n° 4 Point sur le cercle trigonométrique avec tikz 1.18	page 12
ex. n° 5 Définition du Radian	page 13
ex. n° 6 Sin et Cos avec Gnuplot	page 14
ex. n° 7 Sin et Cos sans Gnuplot	page 15
ex. n° 8 Fonction Logarithme	page 16
IV. Initialisation avec la macro \tkzInit	page 17
macro n° 1 Setup environnement \tkzInit	page 17
ex. n° 9 Tout par défaut	page 17
ex. n° 10 Modifier les options, <code>xstep</code> et <code>ystep</code>	page 18
ex. n° 11 Origine personnalisée <code>orig</code>	page 18
ex. n° 12 Utilisation des décimaux	page 19
ex. n° 13 Personnalisation	page 19
V. Les axes du repère	page 20
macro n° 2 Axes des abscisses \tkzX	page 20
ex. n° 15 Par défaut	page 20
ex. n° 16 Options <code>lw</code> et <code>label</code>	page 20
ex. n° 17 options <code>noticks</code> , <code>lw</code> et <code>label</code>	page 20
ex. n° 18 Options <code>pos</code> et <code>label</code>	page 21
ex. n° 19 Pas de label	page 21
ex. n° 20 Personnalisation <code>poslabel</code>	page 21
ex. n° 21 Personnalisation d'un label <code>poslabel</code>	page 21
ex. n° 22 Options <code>unit</code> et <code>nograd</code>	page 21
ex. n° 23 Option <code>unit</code>	page 22
ex. n° 24 Axe trigonométrique Option <code>trig=number</code>	page 22
macro n° 3 Intervalle horizontal hachuré \tkzH	page 23
ex. n° 25 Intervalle hachuré	page 23
macro n° 4 Axe des ordonnées \tkzY	page 24
ex. n° 26 Exemple d'utilisation	page 24
macro n° 5 Modifier les valeurs par des défauts des axes	page 25
ex. n° 27 Modification des axes par défaut \SetUpAxis	page 25
macro n° 6 Un repère \tkzRep	page 26
ex. n° 28 Utilisation du repère	page 26
VI. Les grilles	page 28
macro n° 7 Ajouter une grille \tkzGrid	page 28
ex. n° 30 Presque par défaut option <code>color</code>	page 28
ex. n° 31 origine en plus	page 28
ex. n° 32 Sous grille Option <code>sub</code>	page 29
ex. n° 33 Option <code>subxstep</code> et <code>subystep</code>	page 29
ex. n° 34 Sous grille avec un pas principal modifié	page 29

ex. n° 35	Avec des intervalles importants	page 29
ex. n° 36	Débordement	page 30
VII. Les points		page 31
macro n° 8	Placer des points <code>\tkzPoint</code>	page 31
ex. n° 37	Point sur un axe	page 31
ex. n° 38	Nom différent entre le point et le node	page 31
ex. n° 39	Avec un intervalle hachuré	page 32
ex. n° 40	Option <code>coord</code> , <code>xlabel</code> et <code>ylabel</code>	page 32
ex. n° 41	Mark option <code>mark</code> et <code>size</code>	page 32
ex. n° 42	Point et FP.sty	page 33
macro n° 9	Définir un point sans le tracer <code>\tkzPoint*</code>	page 34
ex. n° 43	Points non marqués	page 34
macro n° 10	Tracer un point déjà défini <code>\tkzDrawPoint</code>	page 34
ex. n° 44	Marqué un point défini	page 34
macro n° 11	Modifier les options par défaut pour les points <code>\SetUpPoint</code>	page 35
ex. n° 45	Modification des options par défaut : place du nom	page 35
ex. n° 46	Modification des autres options par défaut	page 35
macro n° 12	Placer des points (nuage) <code>\tkzSetOfPoints</code>	page 36
ex. n° 47	Nuage de points	page 36
ex. n° 48	Nuage de points sans nom	page 37
VIII. les segments		page 38
macro n° 13	Tracer un segment ou des segments <code>\tkzSegment</code>	page 38
ex. n° 49	Un simple segment	page 38
ex. n° 50	Segment et options <code>symbol</code> , <code>label</code> et <code>time</code>	page 39
macro n° 14	Tracer un segment avec des coordonnées <code>\tkzSegment*</code>	page 40
ex. n° 51	Utilisation de la versin étoilée	page 40
macro n° 15	Distance <code>\tkzDistance</code>	page 41
ex. n° 52	Segment et mesure de distance	page 41
macro n° 16	Marquer un segment <code>\tkzSegmentMark</code>	page 41
ex. n° 53	Segment et mesure de distance	page 41
IX. Autres Macros		page 43
macro n° 17	Placer une marque <code>\tkzMark</code>	page 43
ex. n° 54	Un simple segment	page 43
macro n° 18	Tracer un diagramme <code>\tkzDiagram</code>	page 44
ex. n° 55	Un diagramme	page 44
macro n° 19	Placer du texte <code>\tkzText</code>	page 45
ex. n° 56	Placer un texte	page 45
macro n° 20	Placer une légende <code>\tkzLegend</code>	page 46
ex. n° 57	Légende pour un diagramme	page 46
macro n° 21	Tracer une ligne verticale <code>\tkzVLine</code>	page 47
ex. n° 58	Ligne verticale	page 47
macro n° 22	Tracer une ligne horizontale <code>\tkzHLine</code>	page 47
ex. n° 59	Ligne horizontale	page 47
ex. n° 60	Diagramme avec lignes verticales et horizontales	page 48

Liste de toutes les macros par ordre d'apparition :

• \tkzInit	17
• \tkzX	20
• \tkzXH	23
• \tkzY	24
• \SetUpAxis	25
• \tkzRep	26
• \tkzGrid	28
• \tkzPoint	31
• \tkzPoint*	34
• \tkzDrawPoint	34
• \SetUpPoint	35
• \tkzSetOfPoints	36
• \tkzSegment	38
• \tkzSegment*	40
• \tkzDistance	41
• \tkzSegmentMark	41
• \tkzMark	43
• \tkzDiagram	44
• \tkzText	45
• \tkzLegend	46
• \tkzVLine	47
• \tkzHLine	47



I. Installation.

Le plus simple est de créer un dossier **prof** avec comme chemin : `texmf/tex/latex/prof`. **texmf** est en général le dossier personnel, voici les chemins de ce dossier sur mes deux ordinateurs :

- sous OS X `/Users/ego/Library/texmf` ;

- sous Ubuntu `/home/ego/texmf` .

Je suppose que si vous mettez vos **fichiers .sty** ailleurs, vous savez pourquoi!. L'installation que je propose, n'est valable que pour un utilisateur.

1/ Placez **tkz-base.sty** dans le dossier **prof** .

2/ Ouvrir un terminal, puis faire `sudo texhash`

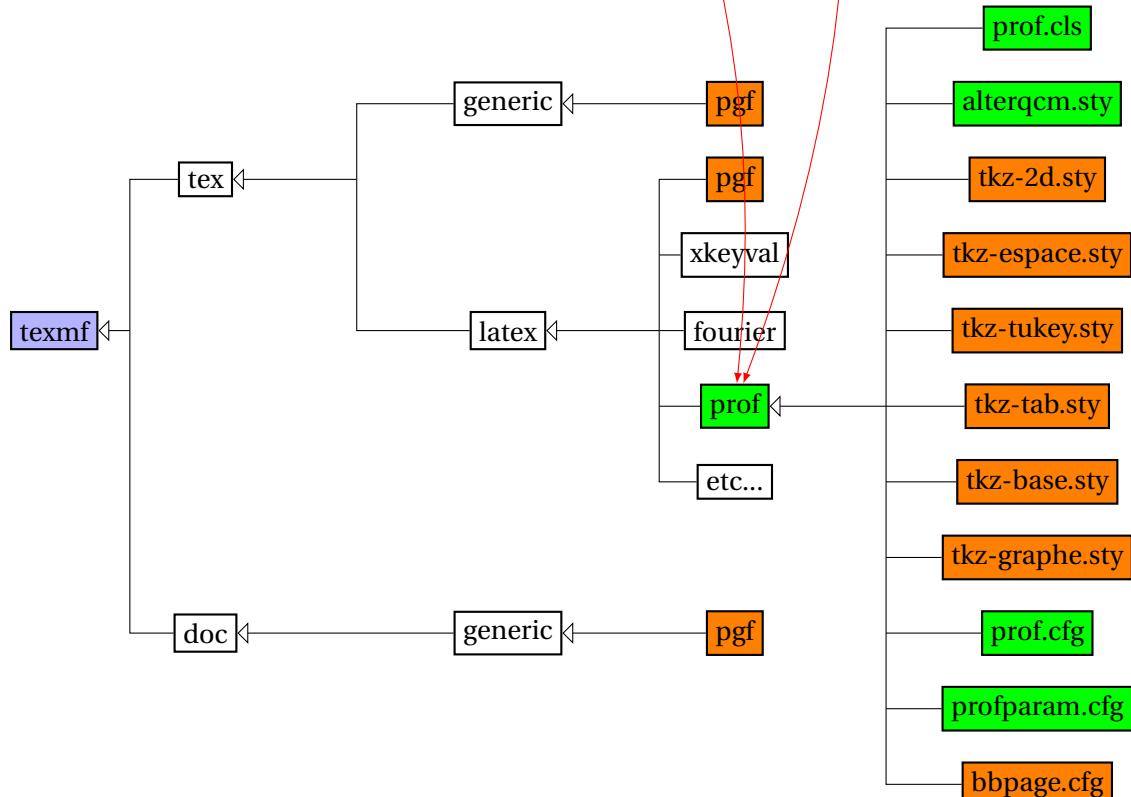
```
ego@altermundus:~$ Last login: Mon Dec 11 23:32:11 on console
Welcome to Darwin!
altermundus:~ ego$ sudo texhash
Password:
texhash: Updating /usr/local/texTeX/share/texmf/ls-R...
texhash: Updating /usr/local/texTeX/share/texmf.gwTeX/ls-R...
texhash: Updating /usr/local/texTeX/share/texmf.local/ls-R...
texhash: Updating /usr/local/texTeX/share/texmf.tetex/ls-R...
texhash: Updating /var/tmp/texfonts/ls-R...
texhash: Done.
altermundus:~ ego$
```

3/ Vérifier que **xkeyval**, **ifthen**, **fp** et **tikz 1.10** sont installés car ils sont obligatoires, pour le bon fonctionnement de tkz-base.

⚠ Pour le bon fonctionnement de ce package, il faut que vous ayez la version 1.3 de pgfutil-common.tex que l'on trouve ici : `texmf/tex/generic/pgf/utilities/pgfutil-common.tex`. Si une mise à jour est nécessaire, la bonne version se trouve là :

<http://pgf cvs.sourceforge.net/pgf/pgf/generic/pgf/utilities/>

Mon dossier **texmf** est structuré ainsi :



II . Compilation des exemples

Le code ci-dessous permet de tester votre installation de **tkz-base**. Je vous rappelle que **fp.sty** et **ifthen.sty** doivent être présents ainsi qu'une version récente de **xkeyval(2.5)**.

```
\documentclass{article}
\usepackage{tikz,tkz-base}
\usetikzlibrary{arrows,plotmarks}%
% pour les français
\usepackage[frenchb]{babel}
% need by numprint
\usepackage[np,autolanguage]{numprint}
\begin{document}
\begin{tikzpicture}
\tkzInit
\tkzGrid
\tkzX
\tkzY
\end{tikzpicture}
\end{document}
```

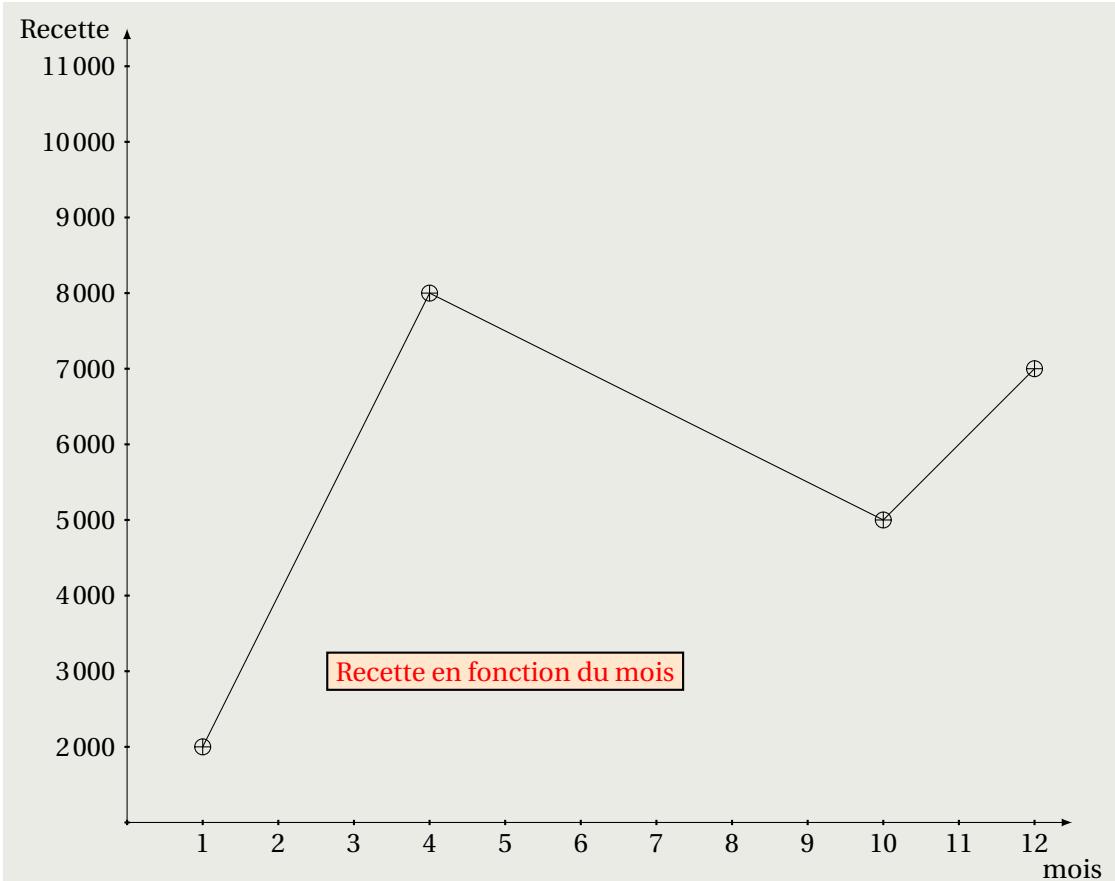
Ce module ne nécessite que les modules cités ci-dessus, pour le moment **Gnuplot** n'est pas nécessaire. Pour ceux qui utilisent **frenchb** avec **babel**, en cas de problème vous pouvez utiliser les commandes suivantes **twoptoff** et **twopton**. Tikz est en effet allergique aux caractères actifs, si le besoin se fait sentir, vous pouvez encadrer l'environnement **tikzpicture** ainsi

```
\twoptoff
\begin{tikzpicture}
...
\end{tikzpicture}
\twopton
```

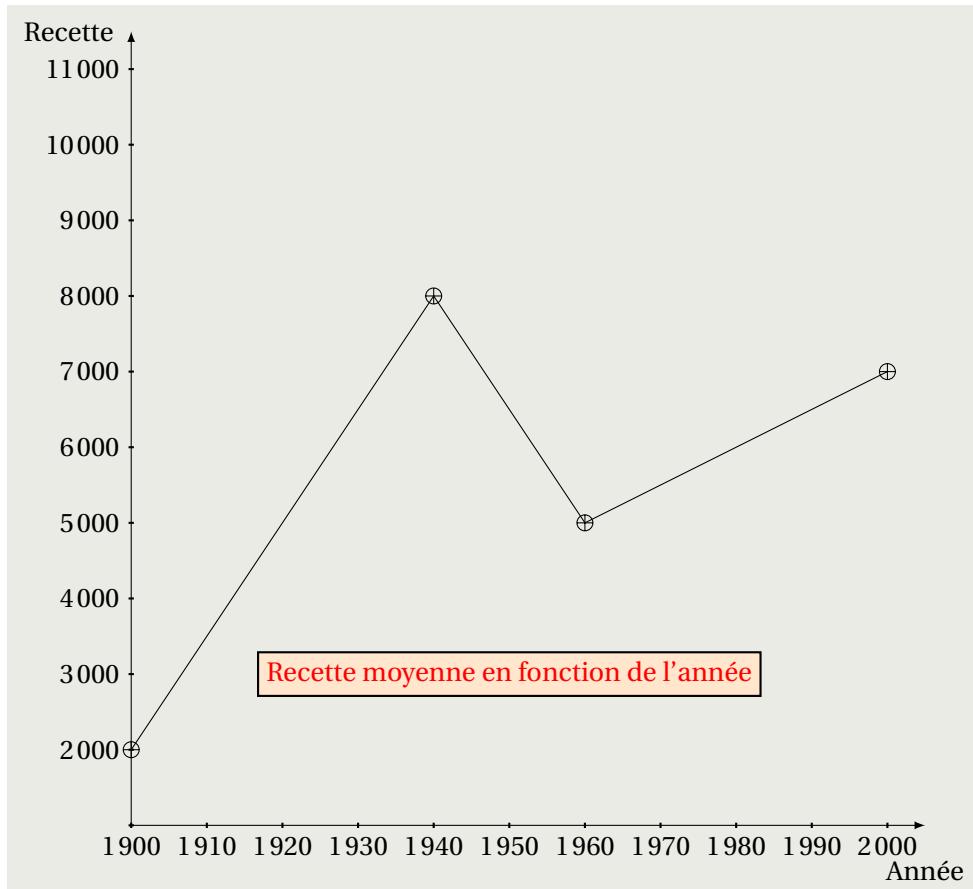
III . Quelques exemples

Voici un exemple très simple qui vous montre qu'une fois le repère défini, il est facile de travailler avec les outils de mon module. Il est cependant possible d'utiliser les outils de Tikz mais dans ce cas il faut revenir au système de coordonnées d'origine.

Exemple n° 1 Recette en fonction du mois



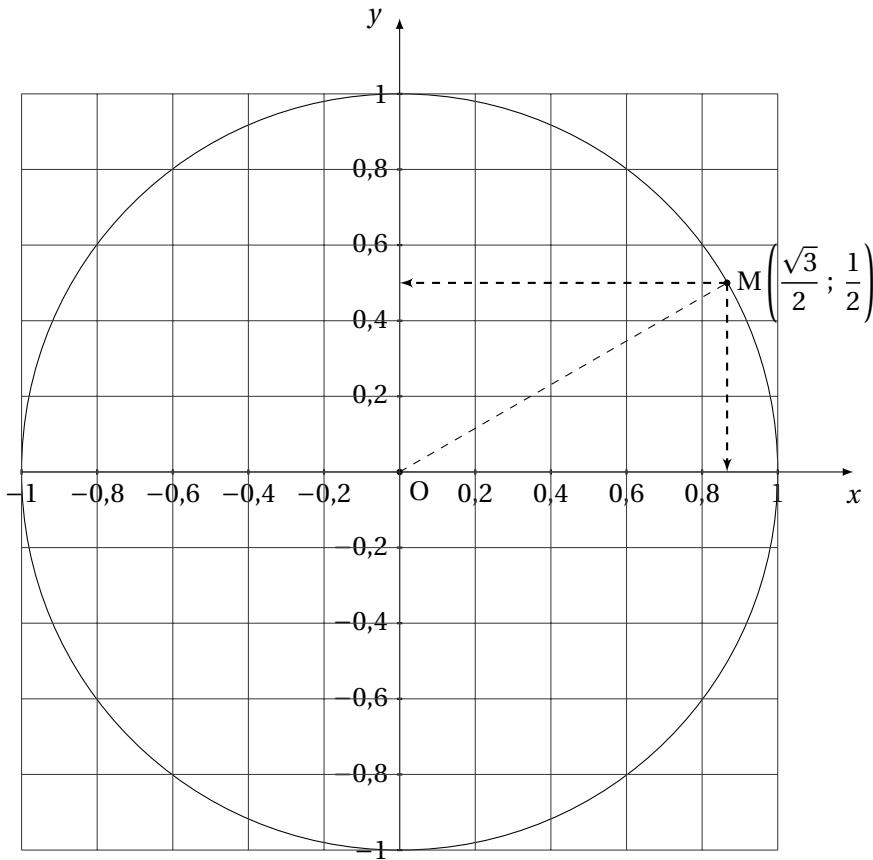
```
\begin{tikzpicture}
\SetUpPoint[mark=oplus,size=3pt,noname]
\tkzInit[xmax=12, ymin=1000, ymax=11000, ystep=1000]
\tkzX[label=mois, poslabel=10pt]
\tkzY[label=Recette]
\tkzSetOfPoints[noname,mark=oplus,size=3pt]{%
    1/2000/A,%
    4/8000/B,%
    10/5000/C,%
    12/7000/D}
\tkzSegment(A/B,B/C,C/D)
\tkzText[style = {draw},%
        color = red,%
        bkgcolor = orange!20](5,3000)%
        {Recette en fonction du mois}
\end{tikzpicture}
```

Exemple n° 2 Recette moyenne en fonction de l'année


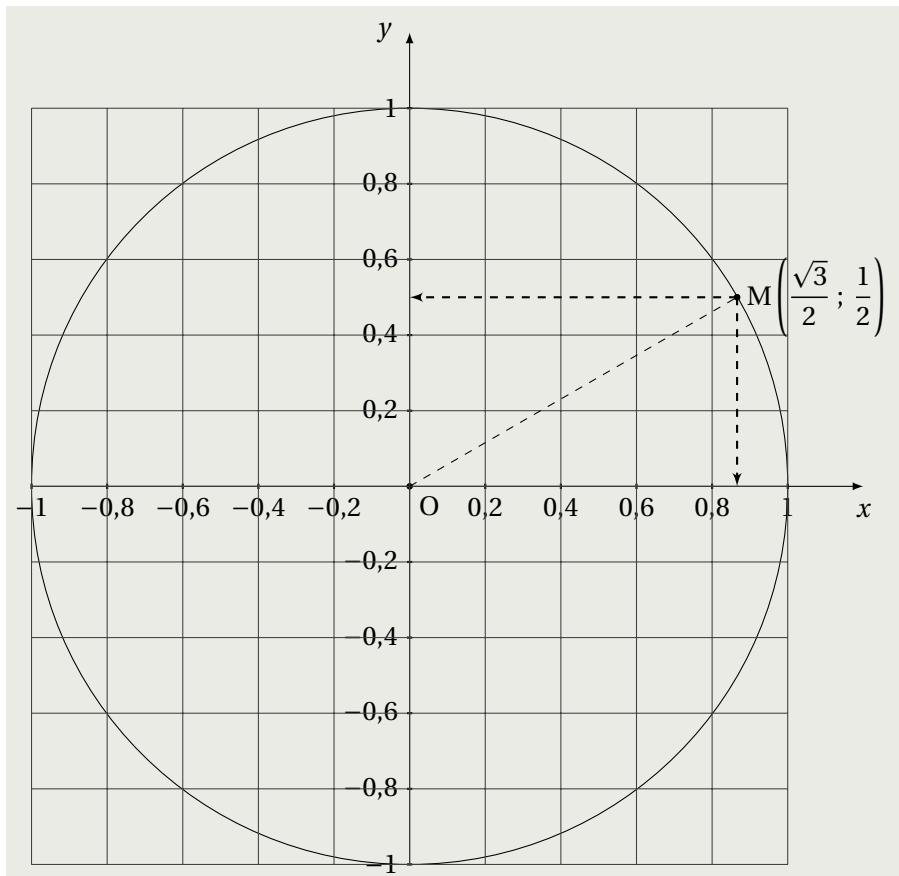
```
\begin{tikzpicture}
  \tkzInit[xmin=1900,xmax=2000,xstep=10, ymin=1000,ymax=11000,ystep=1000]
  \SetUpPoint[mark=oplus,size=3pt,noname]
  \tkzX[label=Année, poslabel=10pt, orig]
  \tkzY[label=Recette]
  \tkzSetOfPoints[noname,mark=oplus,size=3pt]{%
    1900/2000/A,%
    1940/8000/B,%
    1960/5000/C,%
    2000/7000/D}
  \tkzSegment(A/B,B/C,C/D)
  \tkzText[style = {draw},%
          color = red,%
          bkgcolor = orange!20](1950,3000)%
  {Recette moyenne en fonction de l'année}
\end{tikzpicture}
```

Exemple n° 3 Point sur le cercle trigonométrique avec FPsty

La version de **Tikz** possédant une librairie de mathématiques **pgfmath** (voir l'exemple suivant), l'utilisation du module **fp.sty** n'est plus totalement nécessaire mais pour certains calculs mais elle reste utile. Voir la documentation de **fp.sty** pour la syntaxe de celui-ci.



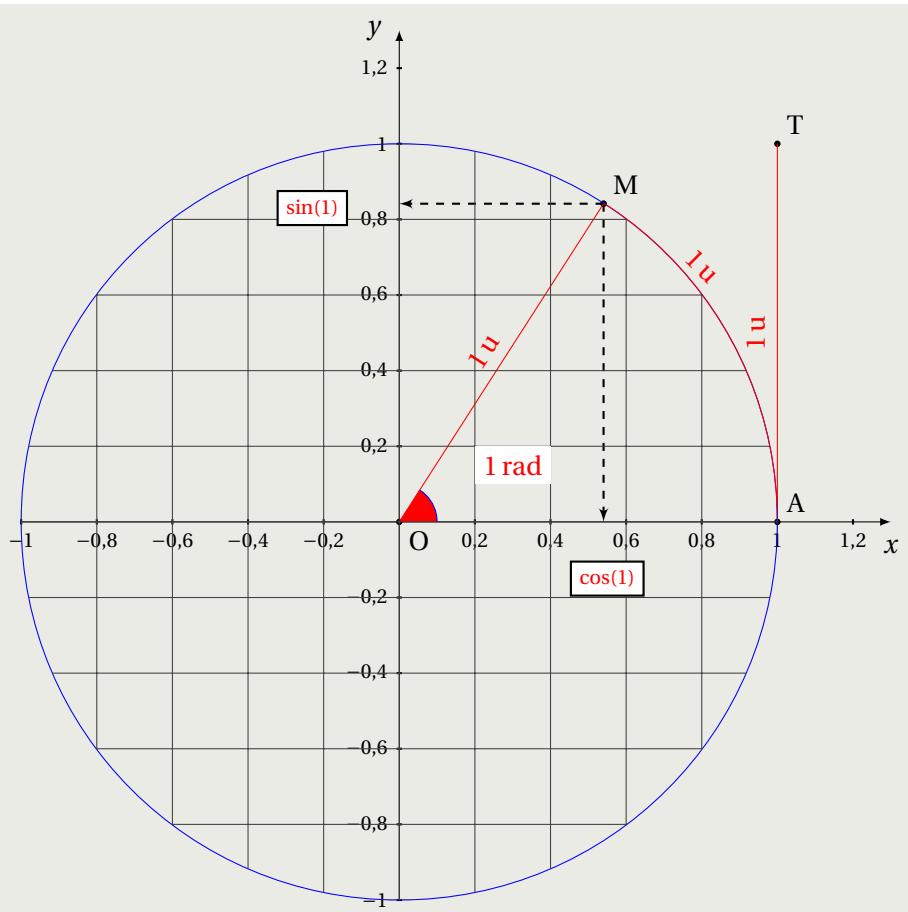
```
\begin{tikzpicture}%
\tkzInit[xmin=-1,xmax=1.1,xstep=.2,ymin=-1,ymax=1.1,ystep=.2]%
\tkzX%
\tkzY%
\tkzPoint(0,0){O}
\draw (0) circle (5cm);
\tkzGrid(-1,-1)(1,1)
\FPeval\My{(\sqrt{2}/2)}%
\FPeval\My{(\sqrt{3}/2)}%
\tkzPoint[pos=right,coord,%
name=$M$\left(\frac{\sqrt{3}}{2};\frac{1}{2}\right)$]%
(\My,\My){M}%
\tkzSegment[style=dashed](O/M)%
\end{tikzpicture}
```

Exemple n° 4 Point sur le cercle trigonométrique avec tikz 1.18

```
\begin{tikzpicture}%
\tkzInit[xmin=-1,xmax=1.1,xstep=.2,ymin=-1,ymax=1.1,ystep=.2]%
\tkzX%
\tkzY%
\tkzPoint(0,0){O}
\draw (0) circle (5cm);
\tkzGrid(-1,-1)(1,1)
\pgfmathparse{\sqrt(3)/2}
\let\My\pgfmathresult
\pgfmathparse{1/2}
\let\My\pgfmathresult%
\tkzPoint[pos=right,coord,%
name=$M$\left(\frac{\sqrt{3}}{2},\frac{1}{2}\right)](\My,\My){M}%
\tkzSegment[style=dashed](0/M)%
\end{tikzpicture}
```

Exemple n° 5 Définition du Radian

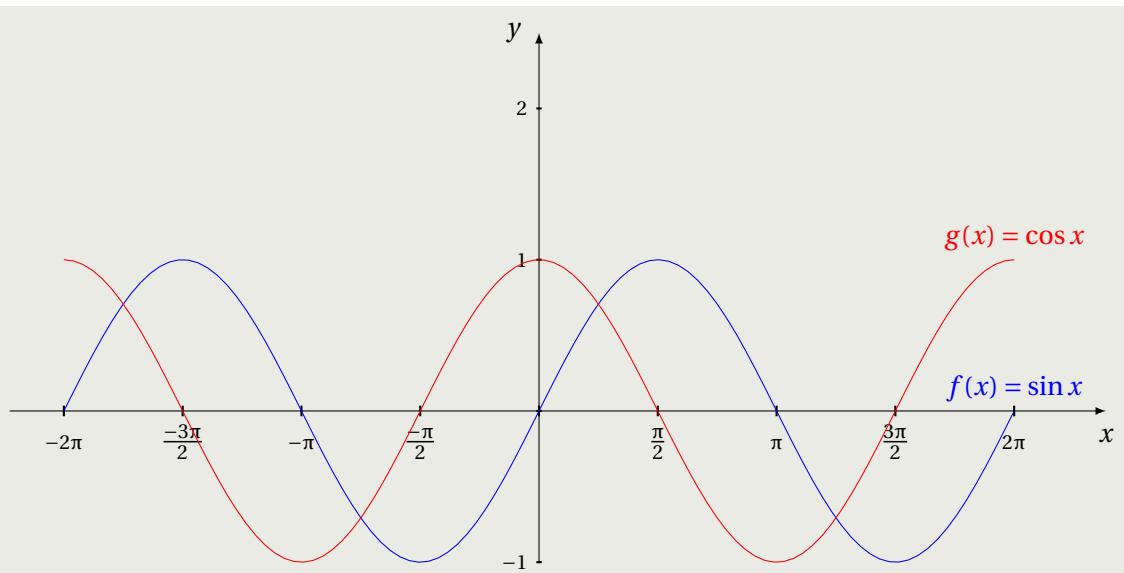
Fp.sty est utile pour calculer les abscisses du point l'angle \widehat{xOM} ayant pour mesure un radian.



```
\begin{tikzpicture}
  \tkzInit[xmin=-1,xmax=1.2,xstep=.2,ymin=-1,ymax=1.2,ystep=.2]
  \tkzX[gradsize=\scriptstyle]
  \tkzY[gradsize=\scriptstyle]
  \tkzPoint(0,0){O}
  \tkzPoint[pos=above right](1,0){A}
  \FPcos{\x}{1}\FPPsin{\y}{1}
  \tkzPoint[pos=above right](1,1){T}
  \tkzPoint[coord,%
            mark   = *,% 
            size   = 1pt,% 
            pos    = above right](\x,\y){M}
  \tkzSegment[color=red,colorlabel=red,label=1\text{ rad}](A/T,0/M)
  \draw[color=blue] (0,0) circle (5cm);
  \path (A) arc (0:40:5) node[rotate=-45,above,color=red] {1\text{ rad}};
  \begin{scope}
  \path[clip](0)--(A)--(M)--cycle;
  \draw[color=blue,fill=red] (0,0) circle (.5cm);
  \end{scope}
  \begin{scope}
  \path[clip](0)--(A)--(T)--(M)--cycle;
  \draw[color=red] (0,0) circle (5cm);
  \end{scope}
  \path[clip] (0,0) circle (5cm);\tkzGrid(-1,-1)(1,1)
  \tkzText[color= red](0.3,0.15){$1\text{ rad}$}
  \tkzText[style={draw},color= red](0.55,-0.15){$\scriptstyle\cos(1)$}
  \tkzText[style={draw},color= red](-0.23,0.83){$\scriptstyle\sin(1)$}
\end{tikzpicture}
```

Exemple n° 6 Sin et Cos avec Gnuplot

Je montre dans cet exemple comment utiliser Gnuplot avec mes repères. Attention cette fois, l'unité du repère doit être 1 cm.

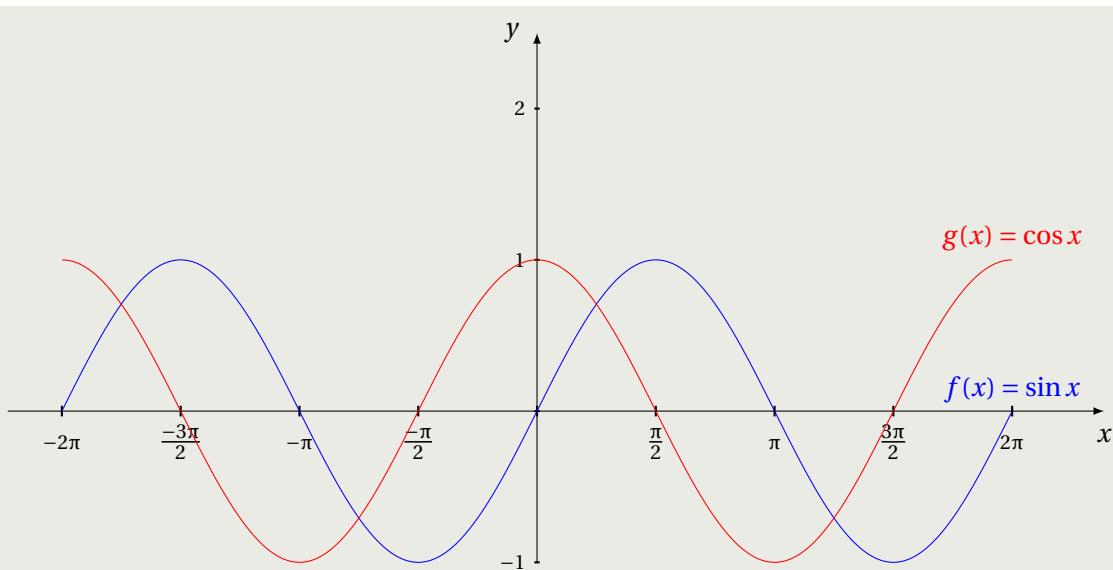


```
\twoptoff
\begin{tikzpicture}[yscale=2]
\tkzInit[xmin=-7,xmax=7,ymin=-1,ymax=2]
\tkzY[gradsize=\scriptstyle]
\tkzX[trig=2]
\draw[color=blue]%
  plot[id=sin,domain=-2*pi:2*pi,samples=100]%
    function{sin(x)}node[above] {$f(x) = \sin x$};
\draw[color=red]%
  plot[id=cos,domain=-2*pi:2*pi,samples=100]%
    function{cos(x)}node[above] {$g(x) = \cos x$};
\end{tikzpicture}
\twopton
```

Exemple n° 7 Sin et Cos sans Gnuplot

Avec Tikz 1.18, plus besoin de gnuplot.

Il faut noter que `domain=-2*pi:2*pi` devient `samples at={-6.28,-6.24,...,6.28}` ce qui permet de déterminer un peu plus de points car le nombre de points par défaut est 25 avec `pgfmath`.

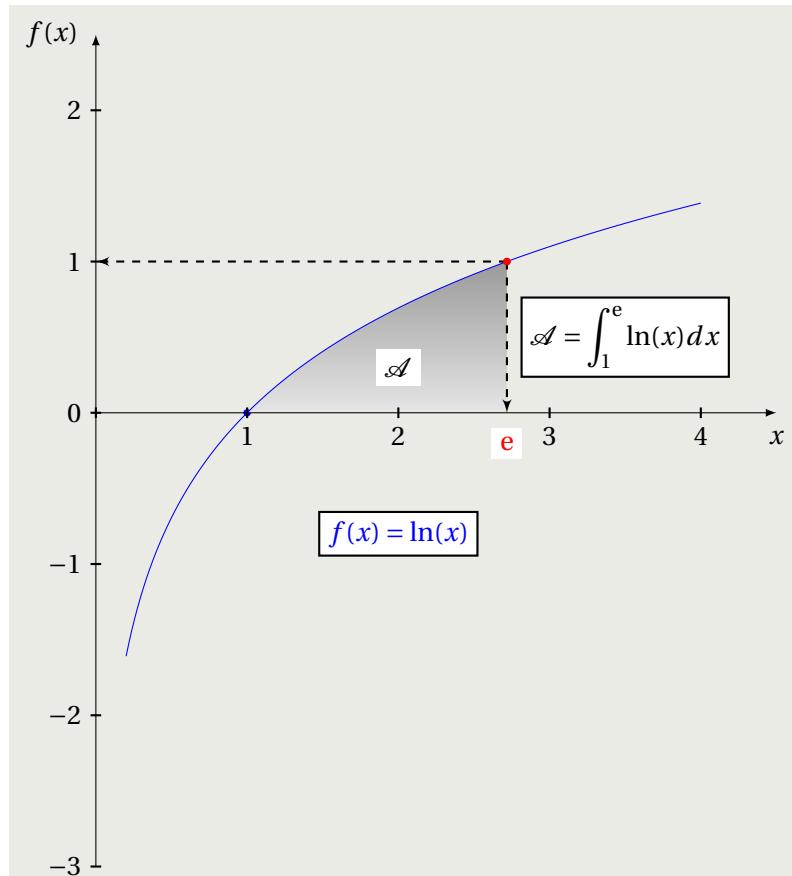


```
\twoptoff
\begin{tikzpicture}[yscale=2]
\tkzInit[xmin=-7,xmax=7,ymin=-1,ymax=2]
\tkzY[gradsize=\scriptstyle]
\tkzX[trig=2]
\draw[color=blue,samples at={-6.28,-6.24,...,6.28}] plot (\x,{sin(\x r)})%
  node[above] {$f(x) = \sin x$};
\draw[color=red,samples at={-6.28,-6.24,...,6.28}] plot (\x,{cos(\x r)})%
  node[above] {$g(x) = \cos x$};
\end{tikzpicture}
\twopton
```

Exemple n° 8 Fonction Logarithme

Un exemple un peu plus sophistiqué...Le module `tkz-fct.sty` facilite le travail avec des fonctions mais là il n'est pas utilisé.

La macro `\tkzText` est utilisée pour placer du texte. Vous trouverez dans la documentation du module `tkz-fct.sty` une méthode permettant de tracer la courbe de la fonction logarithme sans gnuplot, sans fp.sty mais avec pgfmath qui pourtant ne possède pas la fonction ln.



```
\begin{tikzpicture}[scale=2]
\tkzInit[xmin=0,xmax=4,ymin=-3,ymax=2]
\tkzY[label=$f(x)$,orig]
\tkzPoint[noname,color=blue,size=0.6pt](1,0){x}
\twoptoff
\shade[top color=gray!80,bottom color=gray!20] (1,0)%
  plot[id=ln, domain=1:2.718] function{log(x)} |- (1,0);
\draw[color=blue] plot[id=ln, domain=0.2:4, samples=200]%
  function{log(x)};
\twopton
\tkzText[style={draw},color= blue](2,-0.8){$f(x)=\ln(x)$}
\tkzText[color= red](2.718,-0.2){$\text{e}$}
\tkzX
\tkzPoint[coord,noname,color=red,size=0.6pt](2.718,1){x}
\tkzText[style={draw},color= black](3.5,0.5)%
  {$\displaystyle \mathcal{A}=\int_1^e \ln(x) dx$}
\tkzText(2,0.3){$\mathcal{A}$}
\end{tikzpicture}
```

IV. Initialisation \tkzInit

macro n° 1 Setup environnement avec \tkzInit

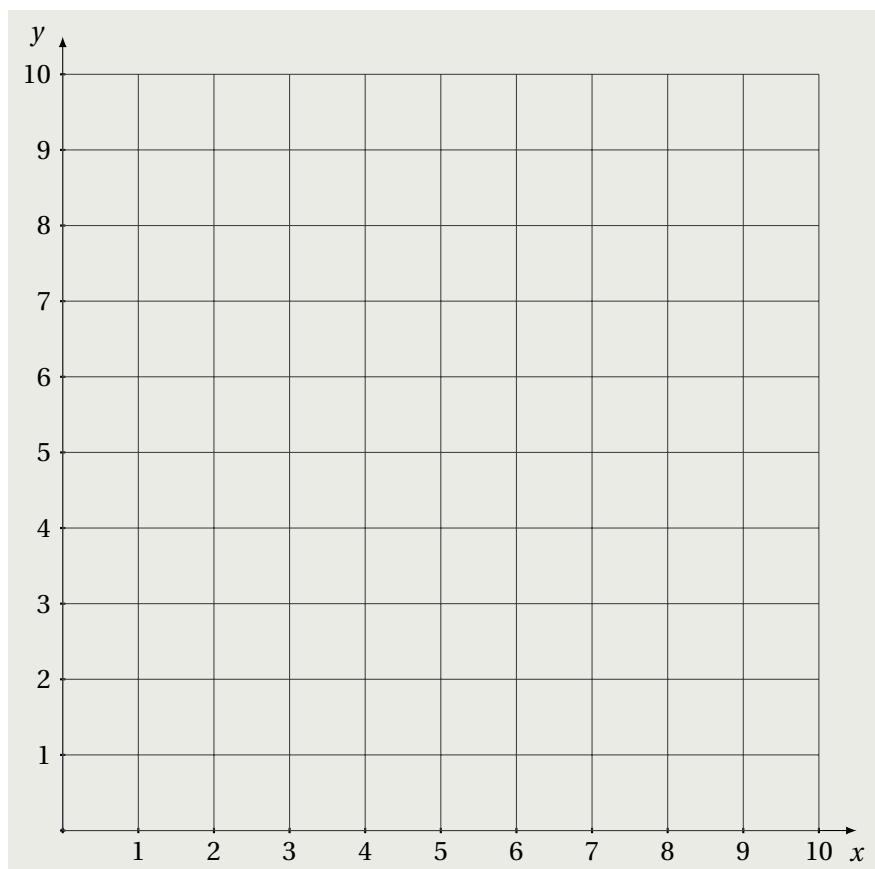
Syntaxe : `\tkzInit[<local options>]`

Le rôle de `\tkzInit` est de définir un repère orthogonal et une partie rectangulaire du plan dans laquelle vous aller placer vos dessins à l'aide de coordonnées cartésiennes. Le repère n'est pas obligatoirement normé. Cette macro permet de définir votre environnement de travail comme avec une calculatrice. Voici le tableau des valeurs des options par défaut :

options	défaut	définition
<code>xmin</code>	0	valeur minimum des abscisses
<code>xmax</code>	10	valeur maximum des abscisses
<code>xstep</code>	1	différence entre deux graduations en x
<code>ymin</code>	0	valeur minimum des ordonnées
<code>ymax</code>	10	valeur maximum des ordonnées
<code>ystep</code>	1	différence entre deux graduations en y

Pour visualiser exactement le rôle de ces options, voici deux macros essentielles `\tkzX` et `\tkzY`. Elles permettent de visualiser les axes de coordonnées. Par défaut, on obtient ceci. la première graduation est absente.

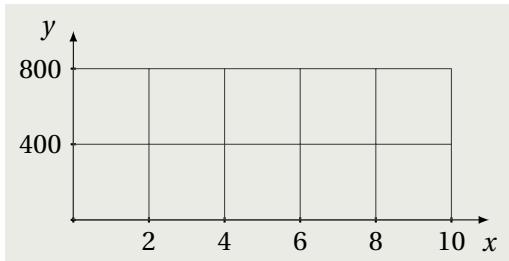
Exemple n° 9 Tout par défaut



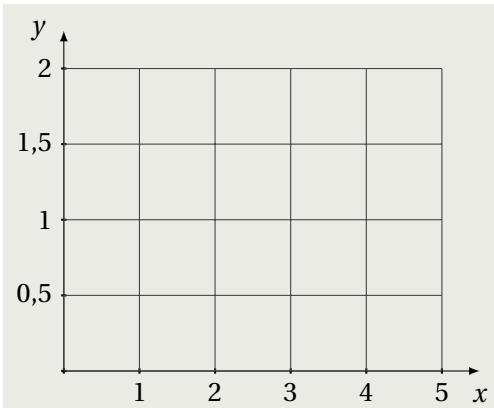
```
\begin{tikzpicture}
\tkzInit
\tkzX
\tkzY
\tkzGrid
\end{tikzpicture}
```

Exemple n° 10 Modification des options : Rôle de `xstep` et `ystep`

Attention, une graduation est représentée par 1 cm. Dans l'exemple ci-dessous `xstep` = 2 correspond à 1 cm, donc entre 2 et 8, il nous faudra 3 cm. De même `ystep`=400, il y a donc 2 cm entre 0 et 800.



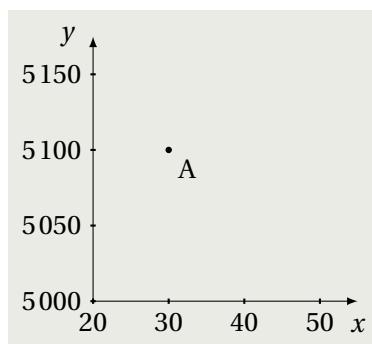
```
\begin{tikzpicture}
    \tkzInit[xmin=0,xmax=10,xstep=2,%
              ymin=0,ymax=800,ystep=400]%
    \tkzX
    \tkzY
    \tkzGrid
\end{tikzpicture}
```



```
\begin{tikzpicture}
    \tkzInit[xmin=0,xmax=5,xstep=1,%
              ymin=0,ymax=2,ystep=.5]
    \tkzX
    \tkzY
    \tkzGrid
\end{tikzpicture}
```

Exemple n° 11 Origine personnalisée.

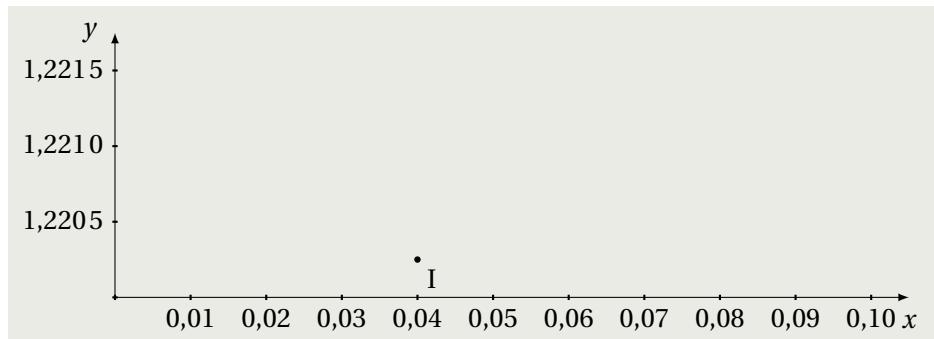
Il est important de remarquer que l'on peut placer un point sans rien calculer.



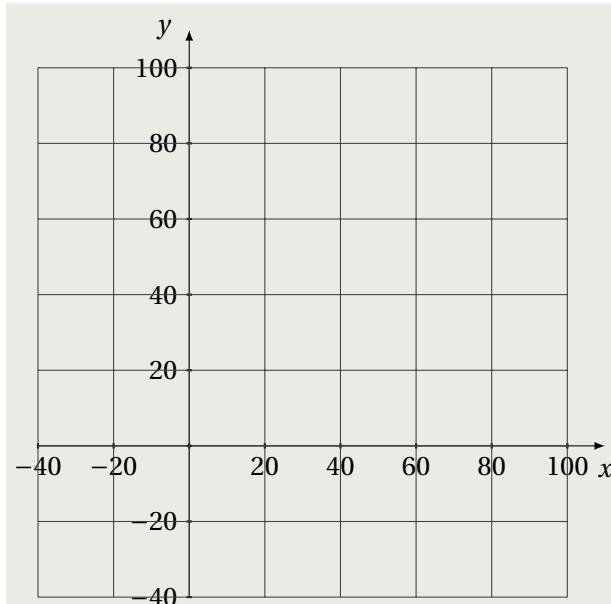
```
\begin{tikzpicture}
    \tkzInit[xmin=20,xmax = 50,xstep = 10,%
              ymin = 5000,ymax = 5150,ystep = 50]%
    \tkzX[orig]
    \tkzY[orig]
    \tkzPoint(30,5100){A}
\end{tikzpicture}
```

Exemple n° 12 Utilisation des décimaux

Il est préférable d'écrire les différents arguments relatifs à un axe avec le même nombre de décimales.



```
\begin{tikzpicture}
\tkzInit[xmin = 0.00, xmax = 0.10, xstep = 0.01,%
         ymin = 1.2200,ymax = 1.2215,ystep = 0.0005]
\tkzX\tkzY
\tkzPoint(.04,1.22025){I}
\end{tikzpicture}
```

Exemple n° 13 Valeurs négatives et grille par défaut.

```
\begin{tikzpicture}
\tkzInit[xmin = -40,xmax = 100,xstep = 20,%
         ymin = -40,ymax = 100,ystep = 20]
\tkzX\tkzY\tkzGrid
\end{tikzpicture}
```

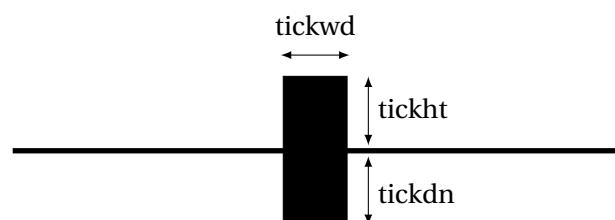
V. Utilisation de \tkzX, \tkzY

macro n° 2 Axe des abscisses tkzX

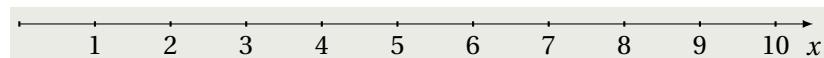
Syntaxe : \tkzX[<local options>]

options	défaut	définition
lw	0.4pt	line width définit la largeur du trait
color	black	couleur de l'axe
orig	false	booléen, donne l'origine
noticks	false	pas de ticks sur l'axe
nograd	false	pas de graduations
unit	false	montre la longueur unité avec un vecteur
pos	below = 3pt	position des ticks (graduations)
label	\$x\$	nom attribué au label
poslabel	3pt	écart vertical à l'axe (peut être négatif)
trig	0	0 pas d'axe trigonométrique sinon pi/trig est l'unité
tickwd	0.8pt	épaisseur du tick
tickup	1pt	hauteur du tick au-dessus de l'axe
tickdn	1pt	profondeur du tick en-dessous de l'axe
gradsize	\textstyle	taille de la graduation.

Exemple n° 14 Composition d'un tick

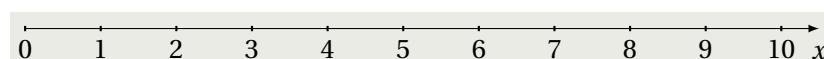


Exemple n° 15 Tout par défaut



```
\begin{tikzpicture}
  \tkzInit
  \tkzX
\end{tikzpicture}
```

Exemple n° 16 Ajout de l'origine orig



```
\begin{tikzpicture}
  \tkzInit
  \tkzX[orig]
\end{tikzpicture}
```

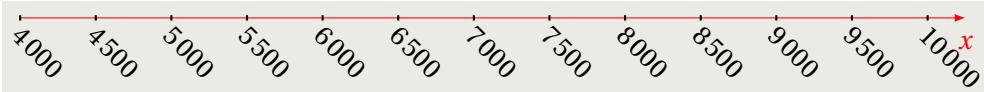
Exemple n° 17 options noticks, lw et label



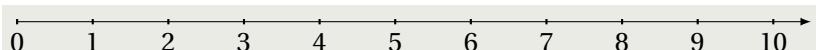
```
\begin{tikzpicture}
  \tkzInit
  \tkzX[noticks, lw = 2pt, label = $t$]
\end{tikzpicture}
```

Exemple n° 18 options **pos** et **label**

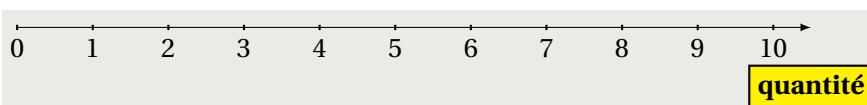
Orientation et positionnement des labels. L'option **pos** nécessite quelques connaissances sur **Tikz**.



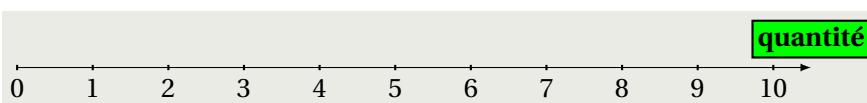
```
\begin{tikzpicture}
\tkzInit[xmin=4000,xmax=10000,xstep=500]
\tkzX[orig,color=red, pos={below right=1pt, rotate=-45}]
\end{tikzpicture}
```

Exemple n° 19 Pas de label **label={}**

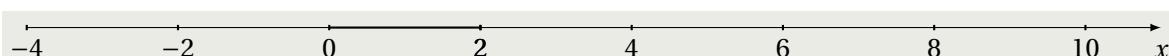
```
\begin{tikzpicture}
\tkzInit
\tkzX[orig,label={}]
\end{tikzpicture}
```

Exemple n° 20 Personnalisation d'un label 1 **poslabel**

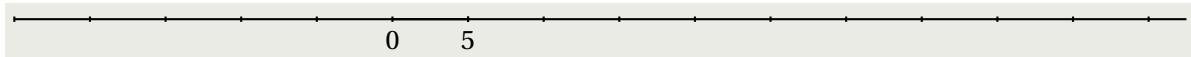
```
\begin{tikzpicture}
\tkzstyle{LabelXStyle}=[draw, fill=yellow]
\tkzInit
\tkzX[orig,label={\textbf{quantité}}, poslabel=14pt]
\end{tikzpicture}
```

Exemple n° 21 Personnalisation d'un label 2 **poslabel**

```
\begin{tikzpicture}
\tkzstyle{LabelXStyle}=[draw, fill=green, inner sep = 2pt]
\tkzInit
\tkzX[orig,label={\textbf{quantité}}, poslabel=-18pt]
\end{tikzpicture}
```

Exemple n° 22 Option **unit**

```
\begin{tikzpicture}[xscale=2]
\tkzInit[xmin=-4,xmax=10,xstep=2]
\tkzX[unit]
\end{tikzpicture}
```

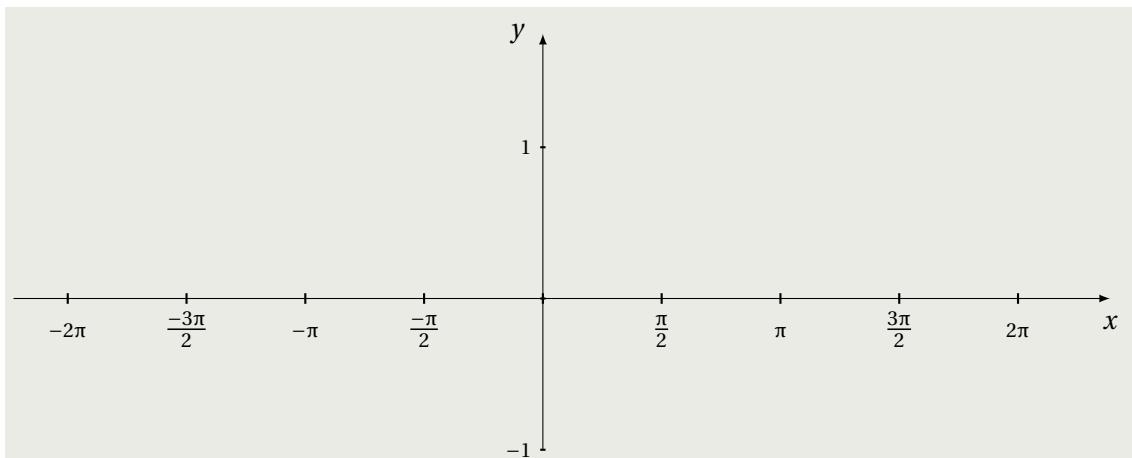
Exemple n° 23 Option `unit` et `nograd`

```
\begin{tikzpicture}
  \tikzstyle{AxeXStyle}=[-,line width=.8pt]
  \tkzInit[xmin=-25,xmax=50,xstep=5]
  \tkzX[unit,nograd,label={}]
\end{tikzpicture}
```

Exemple n° 24 Axe trigonométrique Option `trig=number`

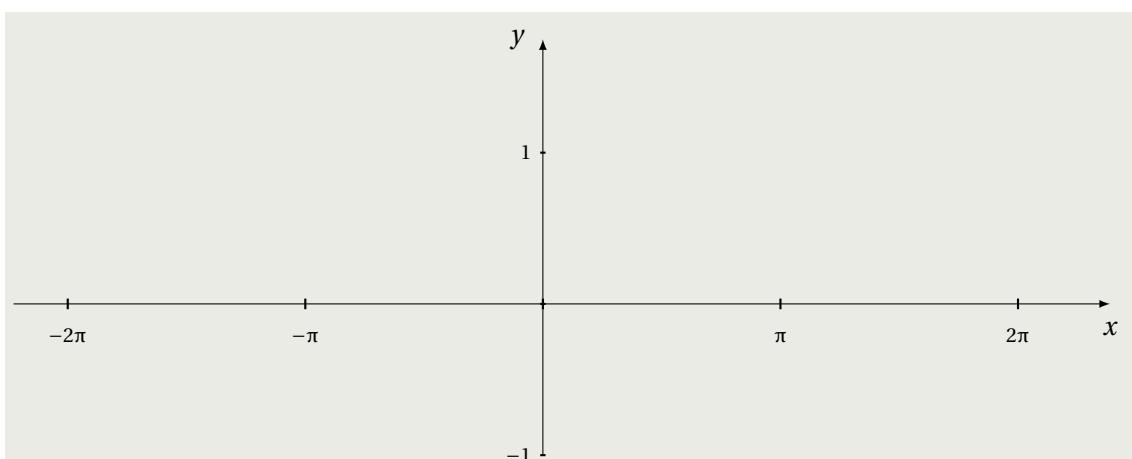
Si `number` = 0 alors l'axe est normal, si `number` est non nul alors l'axe est gradué à l'aide des multiples de $\frac{\pi}{number}$

`trig=2`



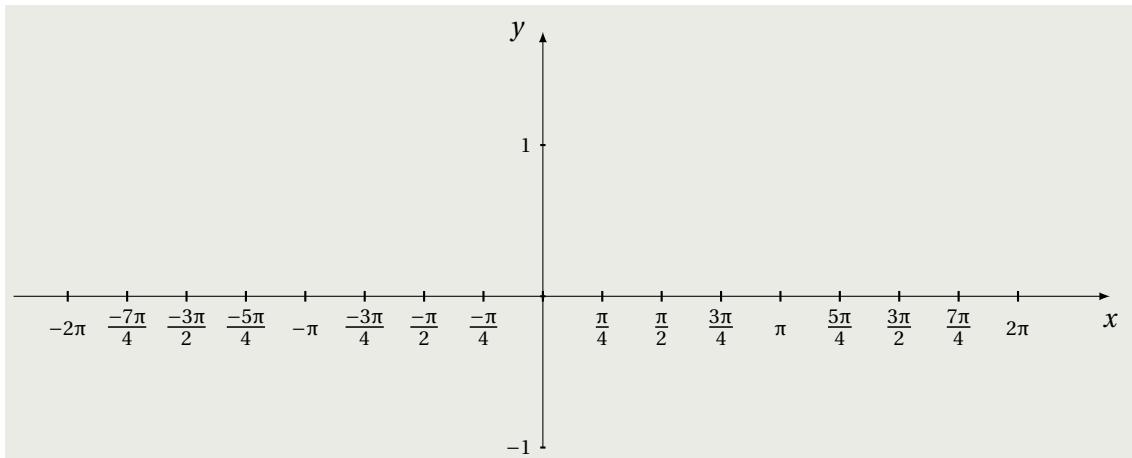
```
\begin{tikzpicture}[yscale=2]
  \tkzInit[xmin=-7,xmax=7,ymin=-1,ymax=1.25]
  \tkzY[gradsize=\scriptstyle]
  \tkzX[trig=2]
\end{tikzpicture}
```

`trig=1`



```
\begin{tikzpicture}[yscale=2]
  \tkzInit[xmin=-7,xmax=7,ymin=-1,ymax=1.25]
  \tkzY[gradsize=\scriptstyle]
  \tkzX[trig=1]
\end{tikzpicture}
```

trig=4



```
\begin{tikzpicture}[yscale=2]
\tkzInit[xmin=-7,xmax=7,ymin=-1,ymax=1.25]
\tkzY[gradsize=\scriptstyle]
\tkzX[trig=4]
\end{tikzpicture}
```

macro n° 3 Intervalle horizontal hachuré

Syntaxe : `\tkzH[<local options>]{<I1,I2,...,In>}` Un élément de la liste est un intervalle I donné sous la forme :

x_A /Boolean/Symbol/ x_B /Boolean/Symbol

Boolean est soit la lettre `F`, soit la lettre `T`(false or true).

Symbol est un crochet fermé ou ouvert `[` ou `]`.

Exemple n° 25 Intervalle hachuré



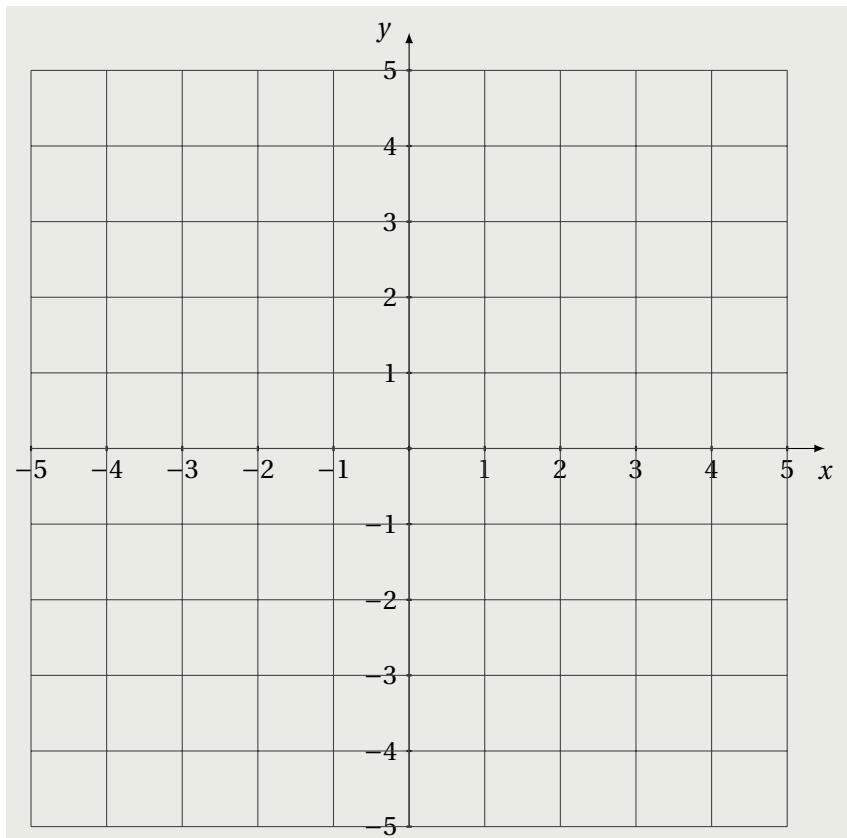
```
\begin{tikzpicture}
\tkzInit[xmax=1,xstep=.1]
\tkzX[label={},noticks,nograd]
\tkzH[0/F//.1/T/[,%
.2/T]/.3/T/],%
.5/T/]/1/F/}
\end{tikzpicture}
```

macro n° 4 Axe des ordonnées

Syntaxe : \tkzY[<local options>]

options	défaut	définition
lw	0.4pt	line width définit la largeur du trait
color	black	couleur de l'axe
orig	false	booléen, donne l'origine
nograd	false	pas de graduations
unit	false	montre la longueur unité avec un vecteur
pos	left=3pt	position des ticks (graduations)
label	\$y\$	nom attribué au label
poslabel	3pt	écart vertical à l'axe (peut être négatif)
noticks	false	pas de ticks sur l'axe
tickwd	0.8pt	épaisseur du tick
tickrt	1pt	partie droite du tick par rapport à l'axe
ticklt	1pt	partie gauche du tick par rapport à l'axe
gradsize	\textstyle	taille de la graduation.

Exemple n° 26 Exemple d'utilisation



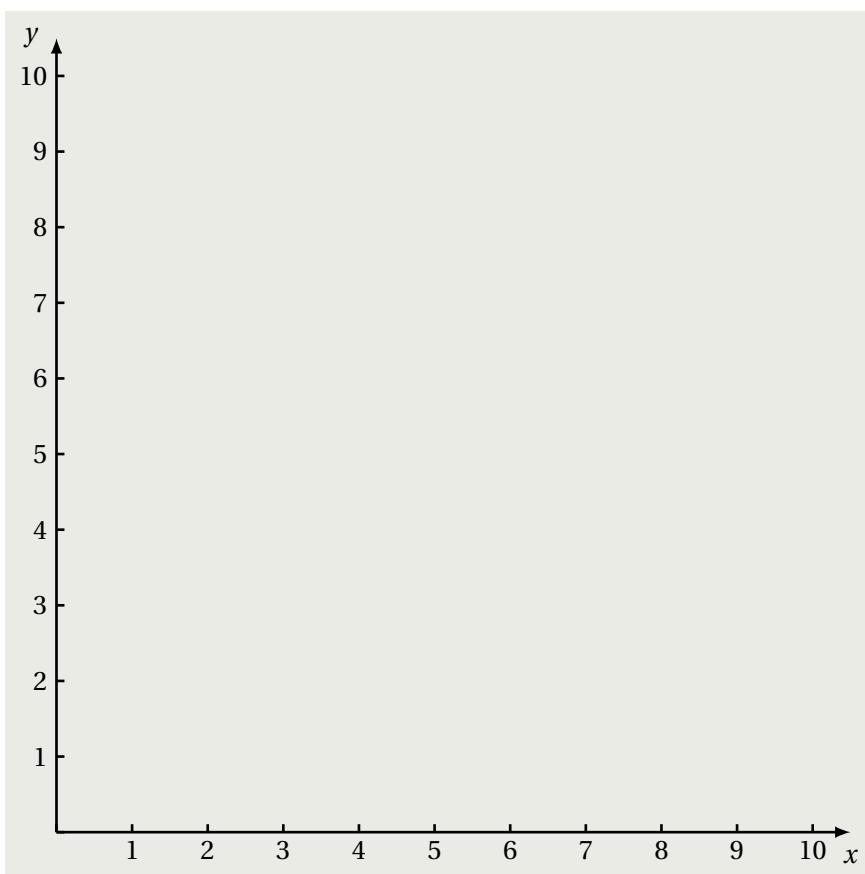
```
\begin{tikzpicture}
\tkzInit[xmin=-5,ymin=-5,xmax=5,ymax=5]
\tkzX
\tkzY
\tkzGrid
\end{tikzpicture}
```

macro n° 5 Modifier les valeurs par des défauts des axes

Syntaxe : `\SetUpAxis[<local options>]`

options	défaut	définition
<code>lw</code>	<code>0.4pt</code>	line width définit la largeur du trait
<code>tickwd</code>	<code>0.8pt</code>	épaisseur du tick
<code>ticka</code>	<code>1pt</code>	partie droite ou au-dessus du tick
<code>tickb</code>	<code>1pt</code>	partie gauche ou en-dessous du tick
<code>gradsize</code>	<code>\textstyle</code>	taille de la graduation.

Exemple n° 27 Modification des axes par défaut



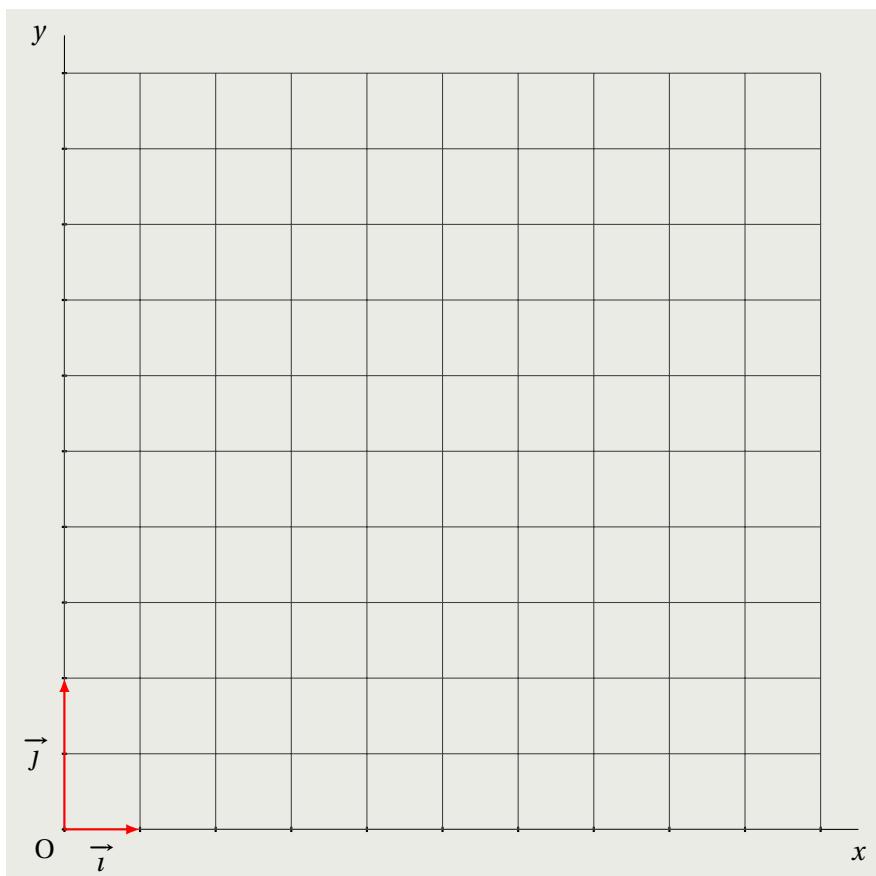
```
\begin{tikzpicture}
  \tkzInit
  \SetUpAxis[lw=1pt,tickwd=1pt,ticka=3pt,tickb=0pt]
  \tkzX\tkzY
\end{tikzpicture}
```

macro n° 6 Repère avec \tkzRep

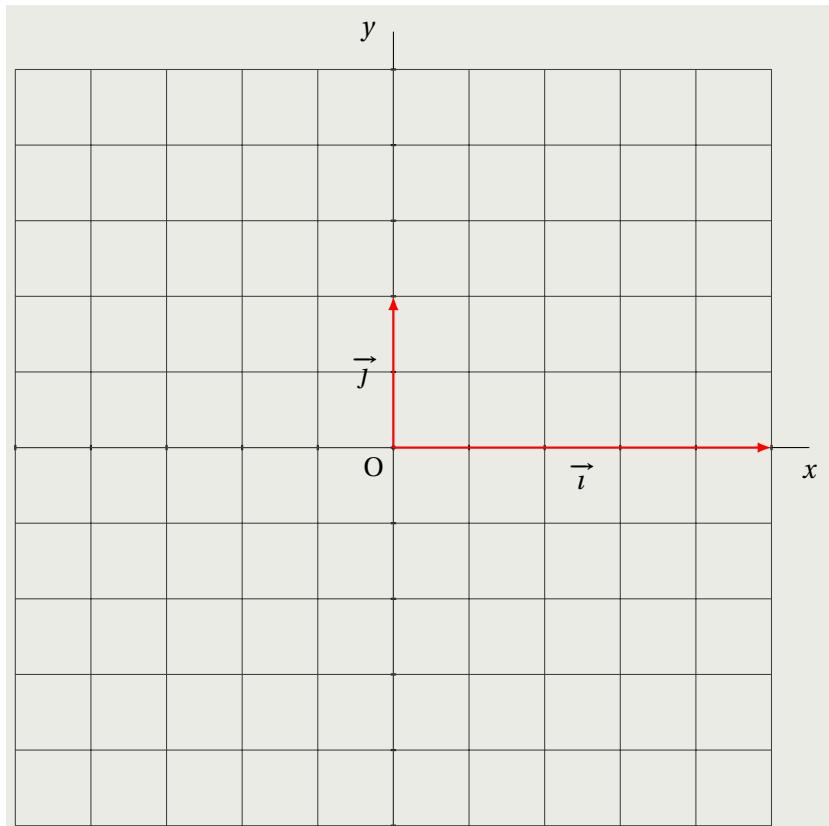
Syntaxe : \tkzRep [*local options*]

options	défaut	définition
lw	0.8pt	line width définit la largeur du trait
xlabel	$\vec{\imath}$	étiquette pour l'axe des abscisses
ylabel	$\vec{\jmath}$	étiquette pour l'axe des ordonnées
poslabel	2pt	Position d l'étiquette par rapport aux axes
xnorm	1	norme du vecteur en x
ynorm	1	norme du vecteur en y
color	black	couleur des traits
colorlabel	black	couleur des étiquettes
orig	true	si orig=false , pas d'origine
LO	\$0\$	2tiquette de l'origine

Exemple n° 28 Exemple d'utilisation



```
\begin{tikzpicture}
\tikzstyle{AxeXStyle}=[-]
\tikzstyle{AxeYStyle}=[-]
\tkzInit
\tkzX[nograd]
\tkzY[nograd]
\tkzGrid
\tkzRep[color=red,ynorm=2]
\end{tikzpicture}
```

Exemple n° 29 Exemple d'utilisation 2

```
\begin{tikzpicture}
\tikzstyle{AxeXStyle}=[-]
\tikzstyle{AxeYStyle}=[-]
\tkzInit[xmin=-5,ymin=-5,xmax=5,ymax=5]
\tkzX[nograd]
\tkzY[nograd]
\tkzGrid
\tkzRep[color=red,xnorm=5,ynorm=2]
\end{tikzpicture}
```

VI . Utilisation de \tkzGrid

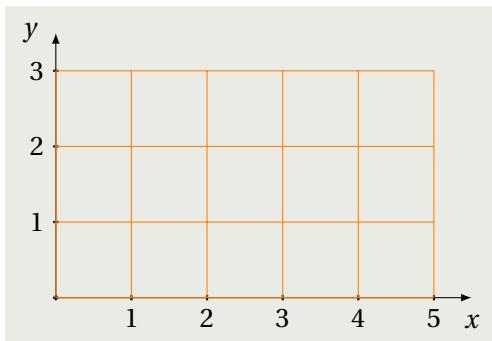
macro n° 7 Les grilles \tkzGrid

Syntaxe : `\tkzGrid[<local options>](<xA ; yA>)(<xB ; yB>)`

options	défaut	définition
sub	true	demande une sous grille
color	darkgray	couleur de la première grille
subcolor	lightgray	couleur de la sous-grille
subxstep	0.2	le pas des sous-graduations
subystep	0.2	le pas des sous-graduations

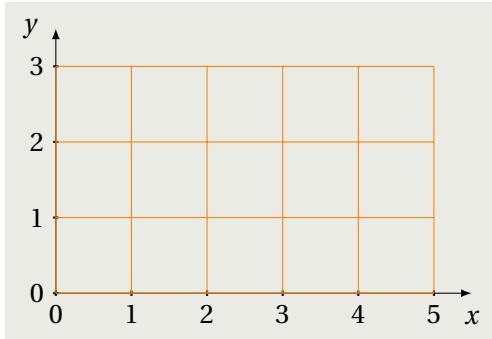
⚠ `xstep` et `ystep` ne sont plus des options. La grille principale dépend des valeurs passées à l'initialisation.

Exemple n° 30 Presque par défaut

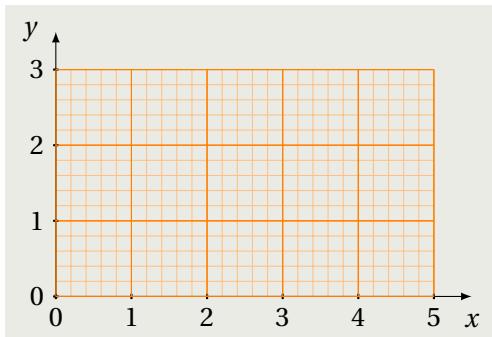


```
\begin{tikzpicture}
  \tkzInit[xmax=5,ymax=3]
  \tkzX
  \tkzY
  \tkzGrid[color=orange]
\end{tikzpicture}
```

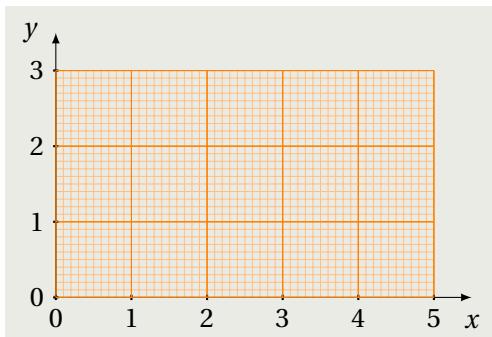
Exemple n° 31 origine en plus



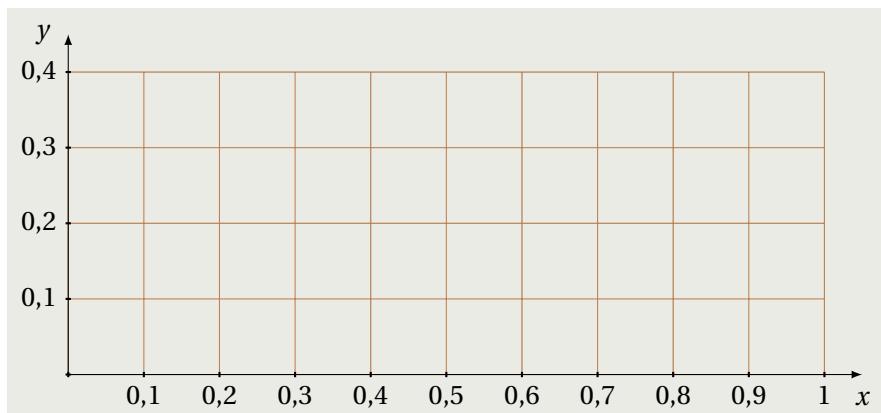
```
\begin{tikzpicture}
  \tkzInit[xmax=5,ymax=3]
  \tkzX[orig]
  \tkzY[orig]
  \tkzGrid[color=orange]
\end{tikzpicture}
```

Exemple n° 32 Sous grille en plus, option `sub`

```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=3]
\tkzGrid[color=orange,sub,subcolor=orange!50]
\tkzX [orig]
\tkzY [orig]
\tkzGrid[color=orange]
\end{tikzpicture}
```

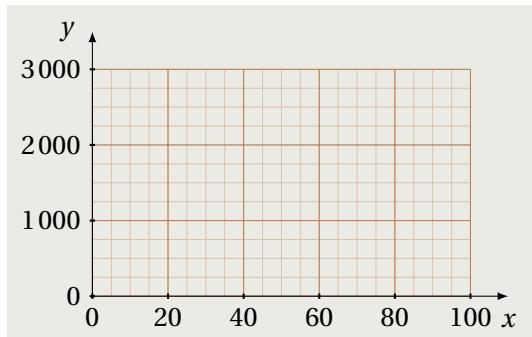
Exemple n° 33 Changement de maille

```
\begin{tikzpicture}
\tkzInit[xmax=5,ymax=3]
\tkzGrid[color      = orange,%
         sub,%
         subcolor = orange!50,%
         subxstep = 0.1,%
         subystep = 0.1]
\tkzX[orig]\tkzY[orig]
\tkzGrid[color=orange]
\end{tikzpicture}
```

Exemple n° 34 Option `xstep` et `ystep`

```
\begin{tikzpicture}
\tkzInit[xmax=1,xstep=.1,ymax=.4,ystep=.1]
\tkzGrid[sub,color=bistre]%
\tkzX
\tkzY
\end{tikzpicture}
```

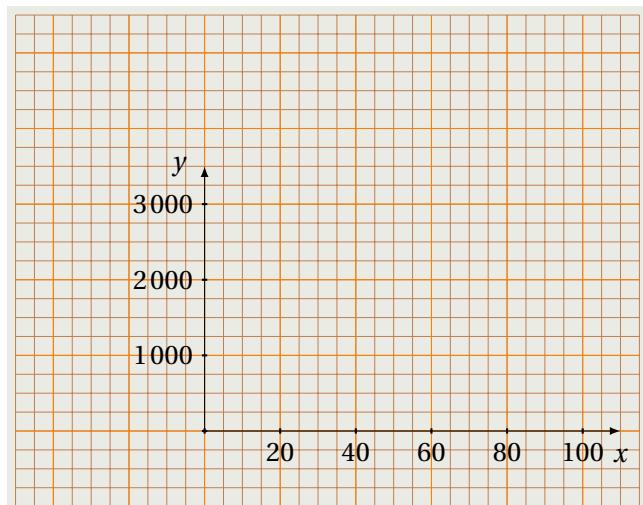
Exemple n° 35 Avec des intervalles importants



```
\begin{tikzpicture}
\tkzInit[xmax=100,xstep=20,ymax=3000,ystep=1000]
\tkzGrid[sub,subxstep=5,subystep=250,color=bistre,subcolor=bistre!50]%
\tkzX[orig]
\tkzY[orig]
\end{tikzpicture}
```

Exemple n° 36 Débordement

La grille peut avoir une taille quelconque.



```
\begin{tikzpicture}
\tkzInit[xmax=100,xstep=20,ymax=3000,ystep=1000]
\tkzGrid[sub,subxstep=5,subystep=250,color=orange,subcolor=bistre]%
(-50,-1000)(115,5500)%
\tkzX
\tkzY
\end{tikzpicture}
```

VII . Placer un point \tkzPoint

macro n° 8 Placer un point

Syntaxe : `\tkzPoint[<local options>](<xA ; yA`

ou encore

Syntaxe : `\tkzPoint* [<local options>](<xA ; yA>){<name>}`

Avec la version étoilée, le point est défini mais non marqué.

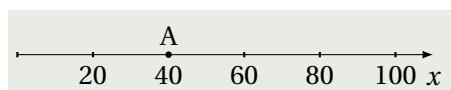
Attention, l'argument `name` est le nom du node. Il doit être composé de caractères autorisés par Tikz. C'est aussi celui du point lorsque cela est valide exemples : a, A, a1 mais A₁ n'est pas accepté. Dans ce cas, vous nommez le node a1 par exemple, et vous nommez le point ainsi : `name= A_1`.

Voici la liste des autres options :

options	défaut	définition
<code>noname</code>	<code>false</code>	si true pas de nom
<code>name</code>	<code>empty</code>	si non vide alors c'est le nom attribué au point
<code>color</code>	<code>black</code>	couleur du point
<code>nomark</code>	<code>false</code>	si true pas de marque
<code>mark</code>	<code>*</code>	représentation du point
<code>size</code>	<code>1pt</code>	représentation du point
<code>namecolor</code>	<code>black</code>	couleur du label
<code>size</code>	<code>\normalsize</code>	taille du point
<code>pos</code>	<code>above right</code>	position du nom
<code>coord</code>	<code>false</code>	booléen pour indiquer si on représente les coordonnées
<code>xlabel</code>	<code>empty</code>	nom de l'abscisse si coord = true
<code>ylabel</code>	<code>empty</code>	nom de l'ordonnée si coord = true
<code>pos xlabel</code>	<code>0pt</code>	écart par rapport l'axe des abs. de xlabel
<code>pos ylabel</code>	<code>0pt</code>	écart par rapport l'axe des ord. de ylabel

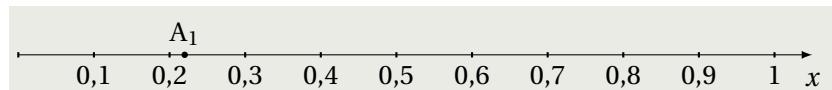
Dans l'exemple suivant, le point A₁ est référencé par A et est nommé A₁, mais le point B est référencé par B et n'est pas nommé

Exemple n° 37 Point sur un axe



```
\begin{tikzpicture}
  \tkzInit[xmax=100,xstep=20,ymax=.5]
  \tkzX
  \tkzPoint[pos=above](40,0){A}
\end{tikzpicture}
```

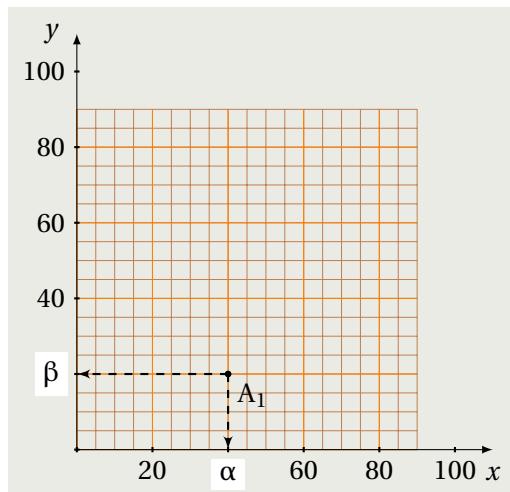
Exemple n° 38 Nom différent entre le point et le node, position du nom.



```
\begin{tikzpicture}
  \tkzInit[xmax=1,xstep=0.1,ymax=.5]
  \tkzX
  \tkzPoint[pos=above,name=$A_1$](0.22,0){A}
\end{tikzpicture}
```

Exemple n° 39 Avec un intervalle hachuré

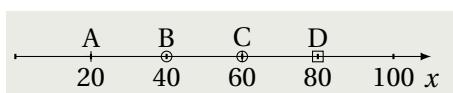
```
\begin{tikzpicture}
\tkzInit[xmax=1,xstep=0.1,ymax=.5]
\tkzX[pos={below=6pt}]
\tkzPoint[pos=above,name=$A_1$](0.22,0){A}
\tkzXH{0.3/F/[.5/F/]}
\end{tikzpicture}
```

Exemple n° 40 Option `coord`, `xlabel` et `ylabel`

```
\begin{tikzpicture}
\tkzInit[xmax=100,xstep=20,ymax=100,ystep=20]
\tkzGrid[sub,subxstep=5,subystep=5,color=orange,subcolor=bistre]%
(0,0)(90,90)%
\tkzX\tkzY
\tkzPoint[coord,name=$A_1$, xlabel=$\alpha$,ylabel=$\beta$](40,20){A}
\end{tikzpicture}
```

Exemple n° 41 Mark option `mark` et `size`

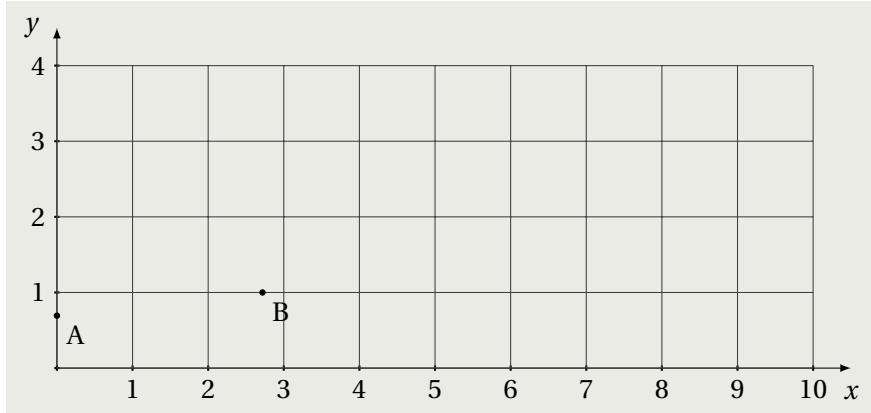
Il est possible d'utiliser la librairie de Tikz.



```
\begin{tikzpicture}
\SetUpPoint[pos={above}]
\tkzInit[xmax=100,xstep=20,ymax=.5]
\tkzX
\tkzPoint[size=2pt,mark=+](20,0){A}
\tkzPoint[size=2pt,mark=o](40,0){B}
\tkzPoint[size=2pt,mark=oplus](60,0){C}
\tkzPoint[size=2pt,mark=square](80,0){D}
\end{tikzpicture}
```

Exemple n° 42 Point et FP.sty

Il est possible d'utiliser FP.



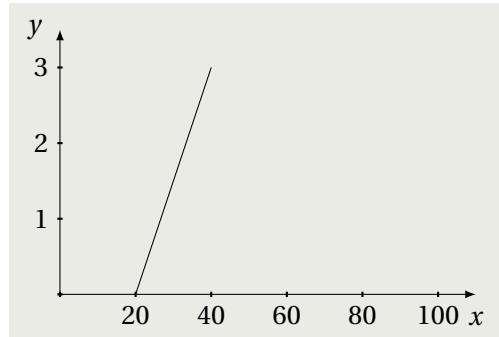
```
\begin{tikzpicture}
  \tkzInit[ymax=4]
  \tkzX\tkzY\tkzGrid
  \tkzPoint(ln(1),ln(2)){A}
  \tkzPoint(exp(1),ln(exp(1))){B}
\end{tikzpicture}
```

macro n° 9 Définir un point sans le tracer

Syntaxe : `\tkzPoint*[(local options)](xA ; yA){name}`

Avec la version étoilée, le point est défini mais non marqué.

Exemple n° 43 Points non marqués

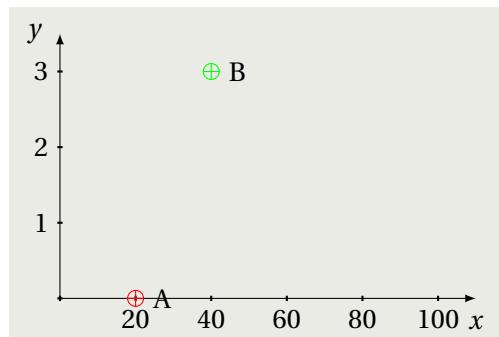


```
\begin{tikzpicture}
  \SetUpPoint[pos={above}]
  \tkzInit[xmax=100,xstep=20,ymax=3]
  \tkzX\tkzY
  \tkzPoint*(20,0){A}
  \tkzPoint*(40,3){B}
  \tkzSegment(A/B)
\end{tikzpicture}
```

macro n° 10 Tracer un point déjà défini

Syntaxe : `\tkzDrawPoint[(local options)](name)`Un point peut être défini comme l'intersection de deux droites. On peut alors le faire apparaître. Les options sont celles de la macro `\tkzPoint`.

Exemple n° 44 Marqué un point défini

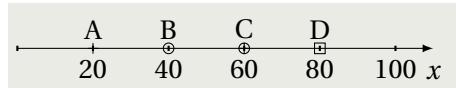


```
\begin{tikzpicture}
  \SetUpPoint[pos={right=3pt}]
  \tkzInit[xmax=100,xstep=20,ymax=3]
  \tkzX\tkzY
  \tkzPoint*(20,0){A}
  \tkzPoint*(40,3){B}
  \tkzDrawPoint[size=3pt,mark=oplus,color=red](A)
  \tkzDrawPoint[size=3pt,mark=oplus,color=green](B)
\end{tikzpicture}
```

macro n° 11 Modifier les options par défaut pour les points

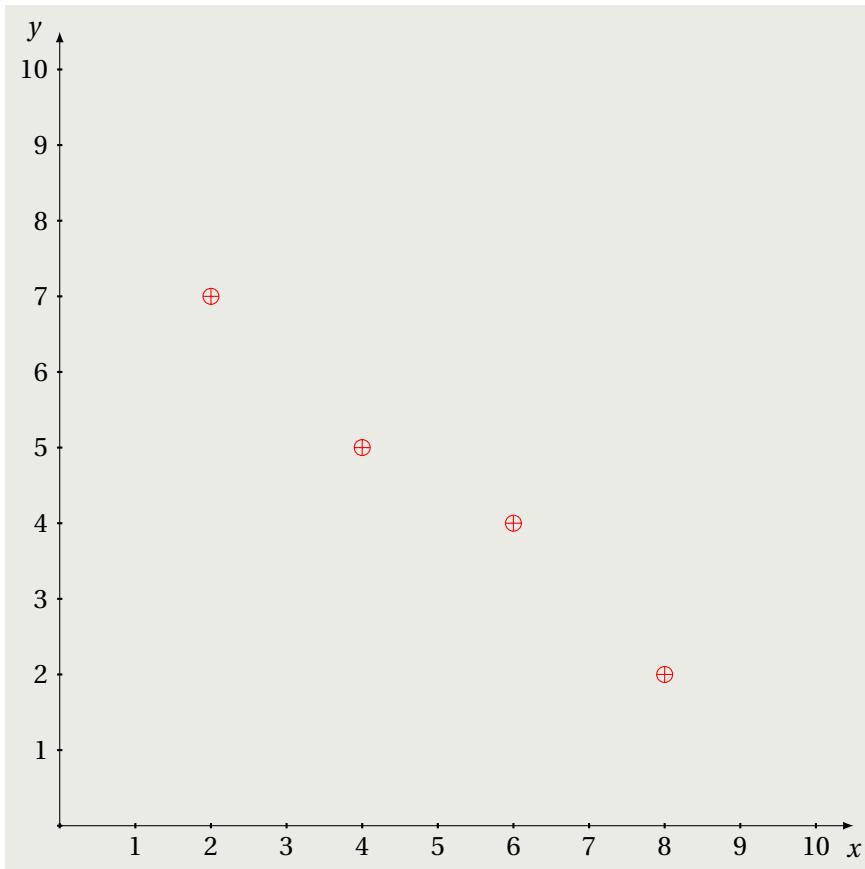
Syntaxe : \SetUpPoint[<local options>]

Exemple n° 45 Modification des options par défaut : place du nom



```
\begin{tikzpicture}
  \SetUpPoint[pos={above}]
  \tkzInit[xmax=100,xstep=20,ymax=.5]
  \tkzX
  \tkzPoint[size=2pt,mark=+](20,0){A}
  \tkzPoint[size=2pt,mark=o](40,0){B}
  \tkzPoint[size=2pt,mark=oplus](60,0){C}
  \tkzPoint[size=2pt,mark=square](80,0){D}
\end{tikzpicture}
```

Exemple n° 46 Modification des autres options par défaut



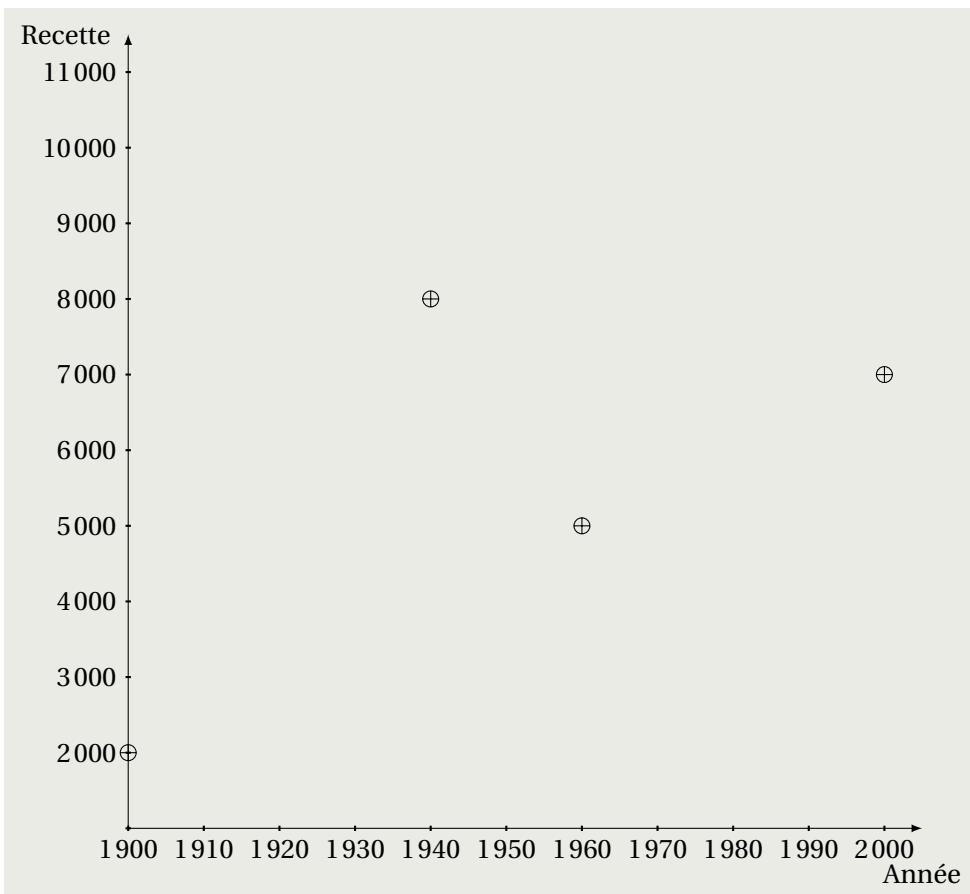
```
\begin{tikzpicture}
  \SetUpPoint[mark=oplus,size=3pt,noname,color=red]
  \tkzInit
  \tkzX
  \tkzY
  \tkzPoint(2,7){A}
  \tkzPoint(4,5){B}
  \tkzPoint(6,4){C}
  \tkzPoint(8,2){D}
\end{tikzpicture}
```

macro n° 12 Placer des points

Syntaxe : `\tkzSetOfPoints[<local options>]{<x_A/y_A/N_A, ...>}`

Les éléments de la liste de points sont composés des deux coordonnées et du nom de chaque point.

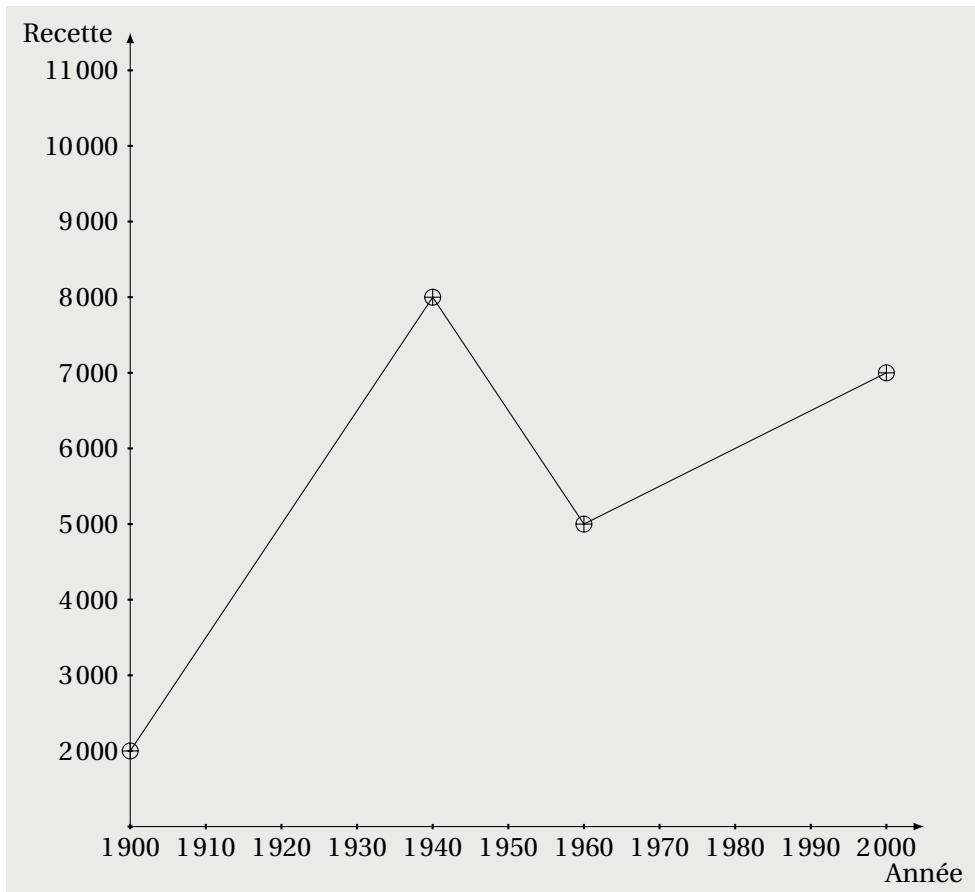
Exemple n° 47 Nuage de points



```
\begin{tikzpicture}
  \tkzInit[xmin=1900,xmax=2000,xstep=10, ymin=1000,ymax=11000,ystep=1000]
  \SetUpPoint[mark=oplus,size=3pt,noname]
  \tkzX[label=Année, poslabel=10pt, orig]
  \tkzY[label=Recette]
  \tkzSetOfPoints[noname,mark=oplus,size=3pt]{%
    1900/2000/A,%
    1940/8000/B,%
    1960/5000/C,%
    2000/7000/D}
\end{tikzpicture}
```

Exemple n° 48 Nuage de points sans nom

Si le nom n'est pas indiqué, alors une référence est attribuée automatiquement au point de la forme `tkzPt0`, `tkzPt1` etc ...



```
\begin{tikzpicture}
  \tkzInit[xmin=1900,xmax=2000,xstep=10,ymin=1000,ymax=11000,ystep=1000]
  \SetUpPoint [mark=oplus,size=3pt,noname]
  \tkzX[label = Année,%
        poslabel = 10pt,orig]
  \tkzY[label = Recette]
  \tkzSetOfPoints[]{%
    1900/2000/,
    1940/8000/,
    1960/5000/,
    2000/7000/}
  \tkzSegment(tkzPt1/tkzPt2,tkzPt2/tkzPt3,tkzPt3/tkzPt4)
\end{tikzpicture}
```

VIII . Tracer des segments \tkzSegment

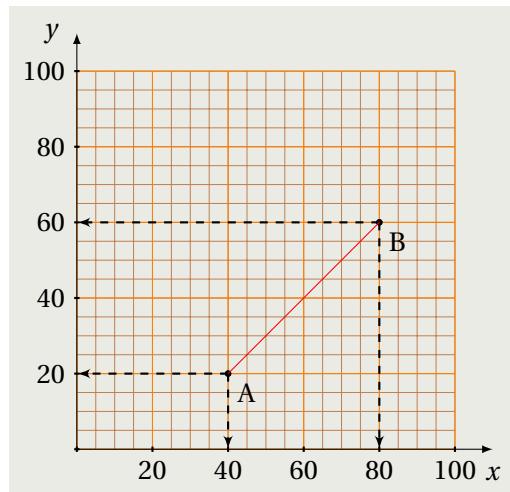
macro n° 13 Tracer un segment ou des segments\tkzSegment

Syntaxe : \tkzSegment[<local options>](<first node/second node, . . . >)

Voici la liste des autres options :

options	défaut	définition
color	black	couleur du trait
lw	0.4pt	épaisseur du trait
style	solid	style du trait dashed,dotted etc...
symbol	{}	marque sur le segment
colorsymbol	black	couleur du symbol
namecolor	black	couleur du label
label	{}	étiquette du segment
colorlabel	black	nom de l'ordonnée si coord = true
poslabel	1pt	écart par rapport aux axes des coordonnées
time	.5	positionne le symbole et le label entre 0 et 1

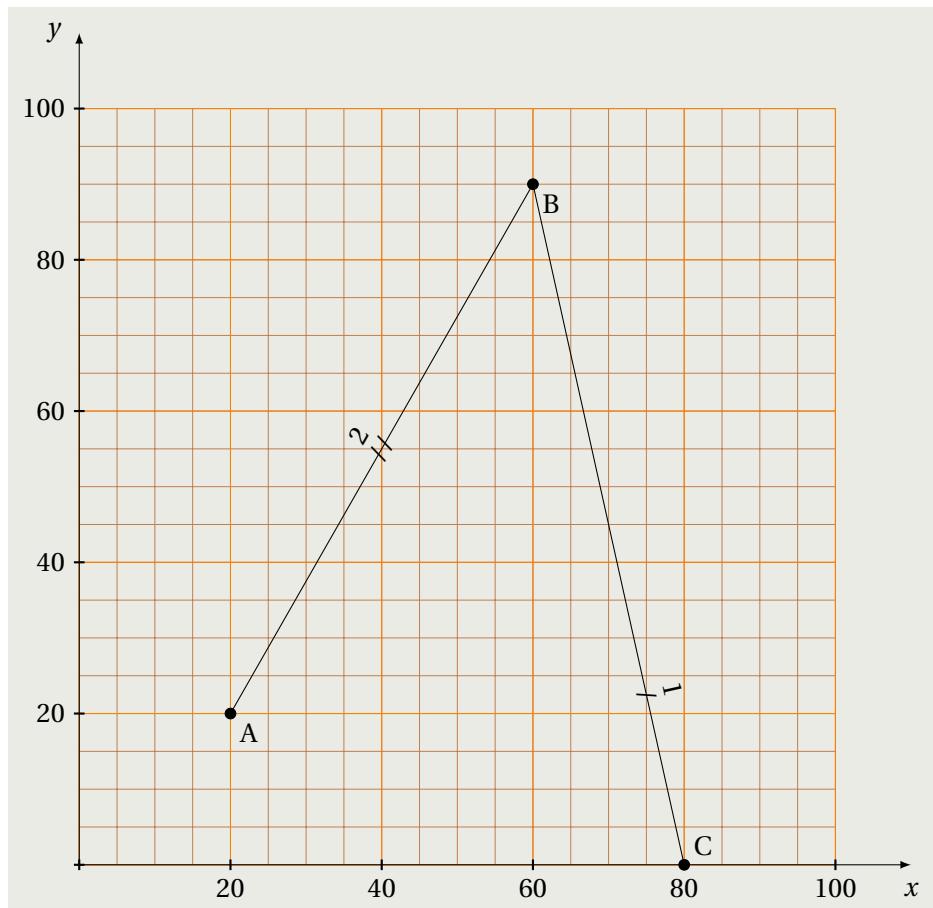
Exemple n° 49 Un simple segment



```
\begin{tikzpicture}
  \tkzInit[xmax=100,xstep=20,ymax=100,ystep=20]
  \tkzGrid[sub,subxstep=5,subystep=5,color=orange,subcolor=bistre]
  \tkzX
  \tkzY
  \tkzPoint[coord](40,20){A}
  \tkzPoint[coord](80,60){B}
  \tkzSegment[color=red](A/B)
\end{tikzpicture}
```

Exemple n° 50 Segment et options `symbol` `label` et `time`

`time` permet de déplacer sur le segment un symbole défini par `symbol`; c'est un nombre compris entre 0 et 1. `label` est une étiquette que l'on peut éloigner du segment par `poslabel`



```
\begin{tikzpicture}[scale=2]
\tkzInit[xmax=100,xstep=20,ymax=100,ystep=20]
\tkzGrid[sub,subxstep=5,subystep=5,color=orange,subcolor=bistre]%
\tkzX
\tkzY
\tkzPoint(20,20){A}
\tkzPoint(60,90){B}
\tkzPoint[pos=above right](80,0){C}
\tkzSegment[symbol={//},label=2,poslabel = 3pt](A/B)
\tkzSegment[symbol={/},label=1,time=.75,poslabel = 3pt](B/C)
\end{tikzpicture}
```

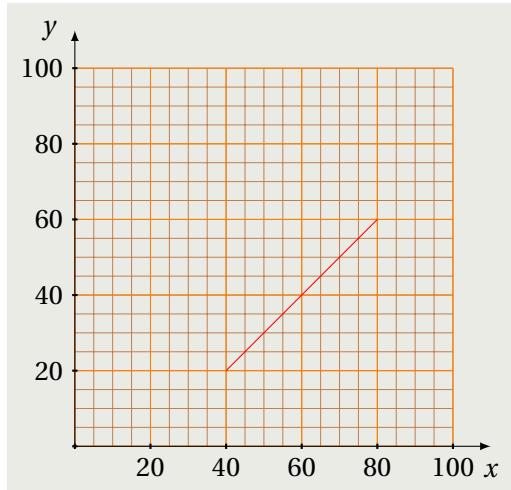
macro n° 14 Tracer un segment avec des coordonnées \tkzSegment*

Syntaxe : \tkzSegment*[, local options](x_A ; y_A)(x_B ; y_B)

Les options sont les mêmes que précédemment

Exemple n° 51 Utilisation de la versin étoilée

Un exemple qui reprend les données de l'exemple 49



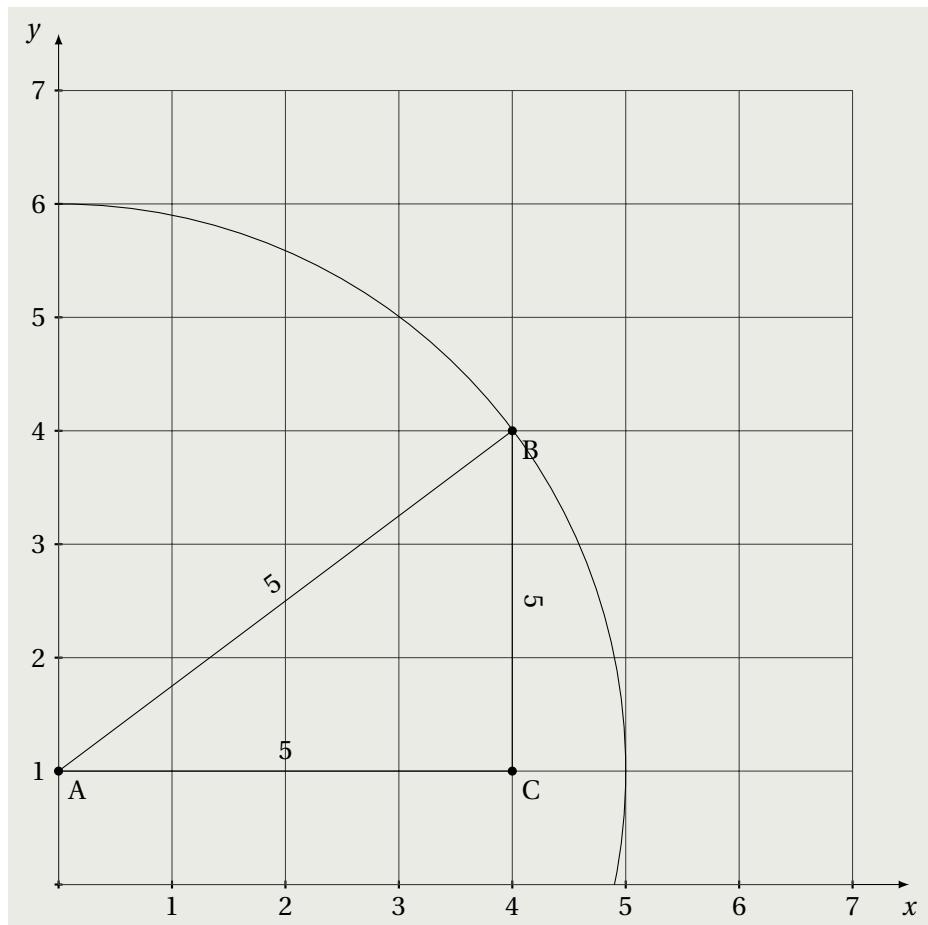
```
\begin{tikzpicture}
\tkzInit[xmax=100,xstep=20,ymax=100,ystep=20]
\tkzGrid[sub,subxstep=5,subystep=5,color=orange,subcolor=bistre]
\tkzX
\tkzY
\tkzSegment*[color=red] (40,20) (80,60)
\end{tikzpicture}
```

macro n° 15 Distance \tkzDistance

Syntaxe : `\tkzDistance(<Point A,Point B>)`

Exemple n° 52 Segment et mesure de distance (voir plot2d)

La macro `\tkzDistance` permet d'avoir la distance en cm entre deux nodes.



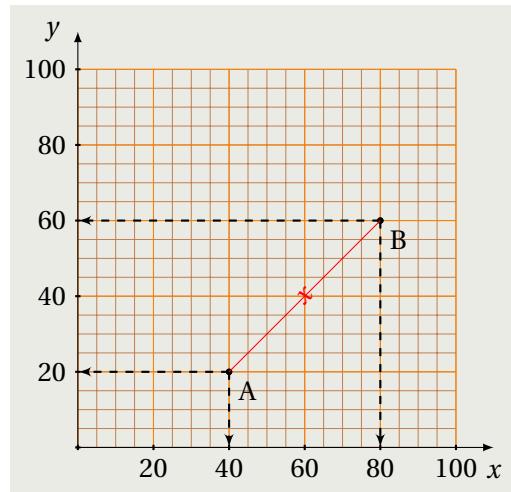
```
\begin{tikzpicture}[scale=1.5]
  \tkzInit[xmax=7,ymax=7]\tkzX\tkzY\tkzGrid
  \tkzPoint(0,1){A}\tkzPoint(4,4){B}\tkzPoint(4,1){C}
  \tkzDistance(A,B)
  \tkzSegment[label1=$5$](A/B,A/C,B/C)
  \tkzClip
  \draw (A) circle (\tkzLen);
\end{tikzpicture}
```

macro n° 16 Marquer un segment \tkzSegmentMark

Syntax: `\tkzSegmentMark[<local options>](<first node/second node>)`

On peut marquer un segment après l'avoir obtenu. Contrairement aux points, les marques ne sont pas celles définies par Tikz mais ce sont des symboles existants dans les polices que vous utilisez.

Exemple n° 53 Marques sur des segments



```
\begin{tikzpicture}
  \tkzInit[xmax=100,xstep=20,ymax=100,ystep=20]
  \tkzGrid[sub,subxstep=5,subystep=5,color=orange,subcolor=bistre]
  \tkzX
  \tkzY
  \tkzPoint[coord](40,20){A}
  \tkzPoint[coord](80,60){B}
  \tkzSegment[color=red](A/B)
  \tkzSegmentMark[colorsymbol=red,symbol=$x$](A/B)
\end{tikzpicture}
```

IX. Divers macros

macro n° 17 Placer une marque \tkzMark

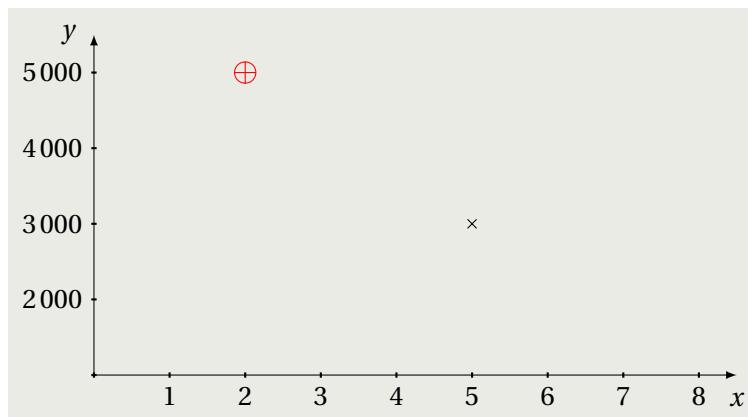
Syntaxe : `\tkzMark[<local options>](<xA ; yA>){<name>}`

Voici la liste des options :

options	défaut	définition
mark	x	tikz possède une librairie de marques
size	.5ex	taille de la marque
options	{}	permet de définir les arguments de mark options (voir le manuel)

Il s'agit d'une notion plus simple que le point, on place une marque de point anonyme. L'argument passé est utiliser pour repérer cette marque

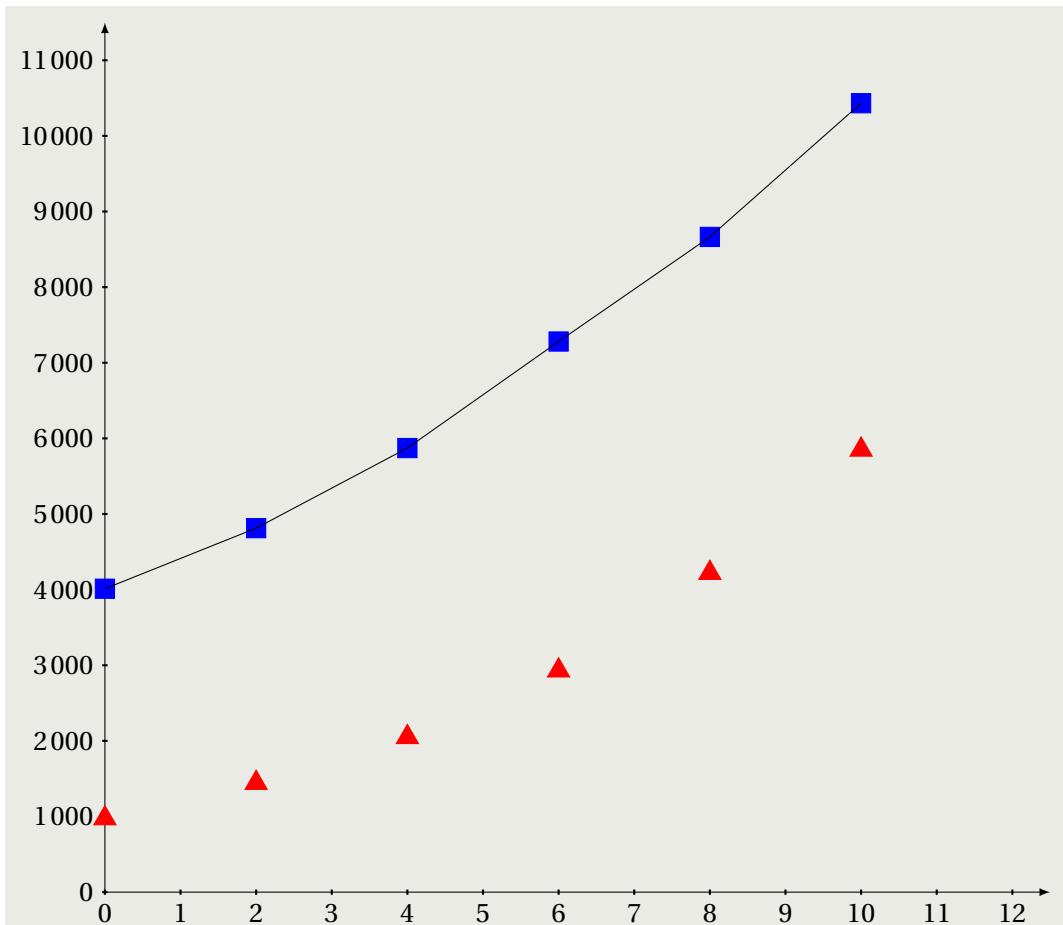
Exemple n° 54 Placer une marque



```
\begin{tikzpicture}
  \tkzInit[xmax=8, ymin=1000, ymax=5000, ystep=1000]
  \tkzX
  \tkzY
  \tkzMark(5,3000){M1}
  \tkzMark[mark=oplus, size=4pt, color=red](2,5000){M1}
\end{tikzpicture}
```

macro n° 18 Tracer un diagramme `\tkzDiagram`Syntaxe : `\tkzDiagram[<local options>]{<xA/yA,...>}`La différence avec le nuage de points, c'est qu'il n'y a pas de noms à donner car ils sont attribués automatiquement : tkzMark1, tkzMark2 etc... Voici la liste des options est la même que pour `\tkzMark`.

Exemple n° 55 Un diagramme



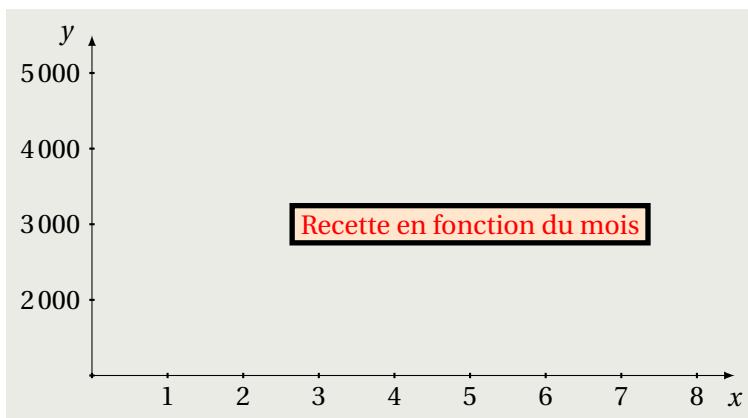
```
\begin{tikzpicture}
  \tkzInit[xmax=12,ymax=11000,ystep=1000]
  \tkzX[orig,label={}]
  \tkzY[orig,label={}]
  \tkzDiagram[mark=triangle*,size=1ex,color=red]%
  {0/975,2/1443,4/2049,6/2930,8/4220,10/5850}
  \tkzDiagram[mark=square*,size=.75ex,color=blue]%
  {0/4012,2/4813,4/5872,6/7281,8/8664,10/10432}
  \tkzSegment(tkzMark1/tkzMark2,tkzMark2/tkzMark3,tkzMark3/tkzMark4,tkzMark4/tkzMark5,tkzMark5/tkzMark6)
\end{tikzpicture}
```

macro n° 19 Placer du texte `\tkzText`

Syntaxe : `\tkzText[<local options>]{<text>}` Voici la liste des options :

options	défaut	définition
color	black	couleur du point
bkgcolor	white	couleur du fond
opacity	1	opacité
style	{}	permet d'utiliser des options de Tikz

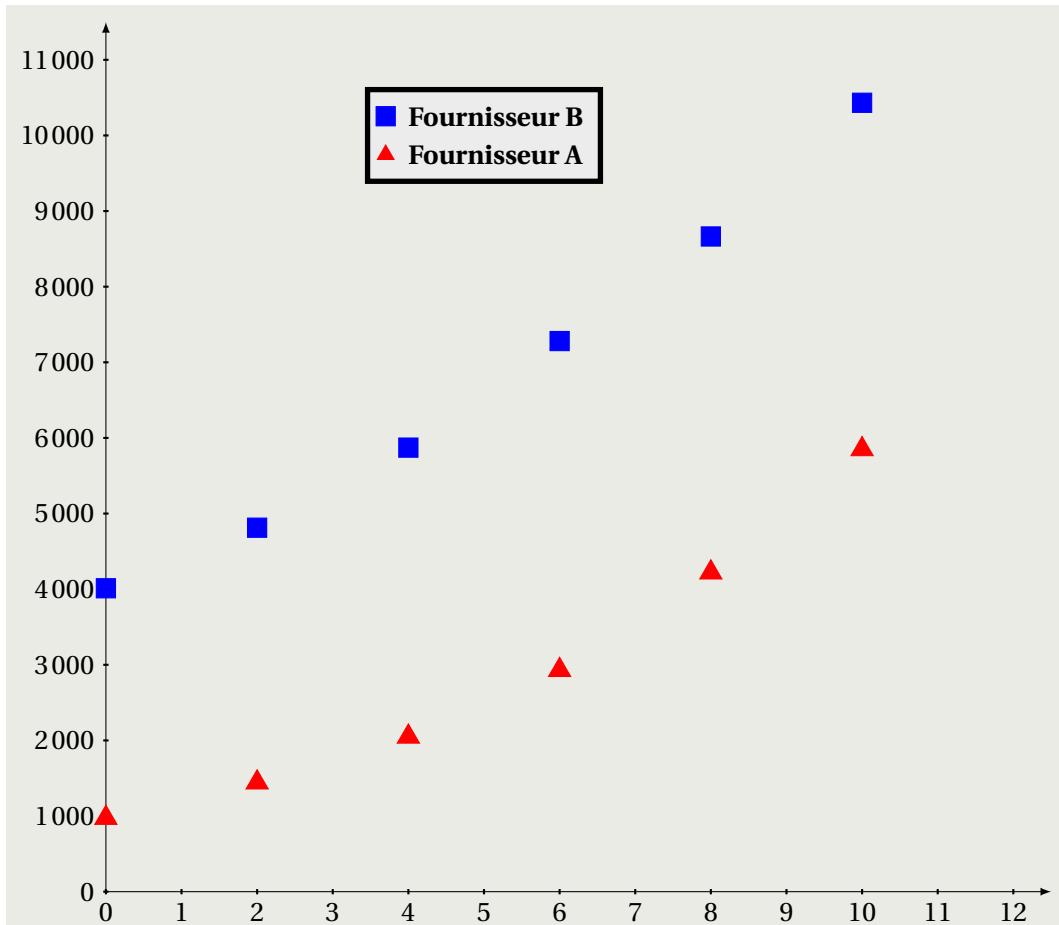
Exemple n° 56 Placer un texte



```
\begin{tikzpicture}
  \tkzInit[xmax=8,ymax=1000,ymax=5000,ystep=1000]
  \tkzX
  \tkzY
  \tkzText[style = {draw,line width=2pt},%
           color = red,%
           bkgcolor = orange!20](5,3000)%
  {Recette en fonction du mois}
\end{tikzpicture}
```

macro n° 20 Mettre une légende \tkzLegend

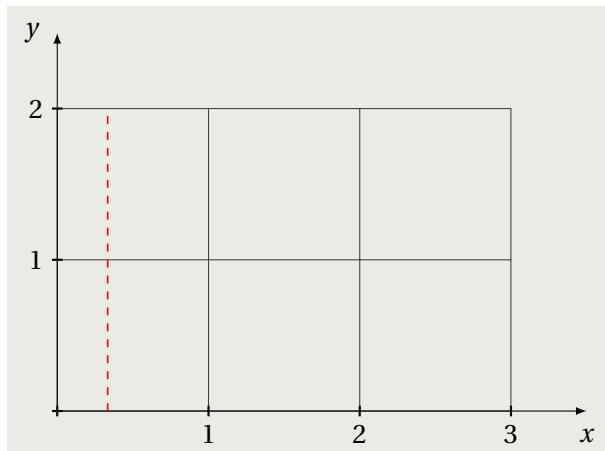
Exemple n° 57 Légende pour un diagramme



```
\begin{tikzpicture}
\tkzInit[xmax=12,ymax=11000,ystep=1000]
\tkzX[orig,label={}]
\tkzY[orig,label={}]
\tkzDiagram[mark=triangle*,size=1ex,color=red]%
{0/975,2/1443,4/2049,6/2930,8/4220,10/5850}
\tkzDiagram[mark=square*,size=.75ex,color=blue]%
{0/4012,2/4813,4/5872,6/7281,8/8664,10/10432}
\tkzLegend[color=lightgray!30,lw=2pt]%
(5,10000)%
{triangle*/1ex/red/\textbf{Fournisseur A},%
square*/0.75ex/blue/\textbf{Fournisseur B}}
\end{tikzpicture}
```

macro n° 21 Tracer une ligne verticale `\tkzVLine`Syntaxe : `\tkzVLine[<local options>](<list of values>)`

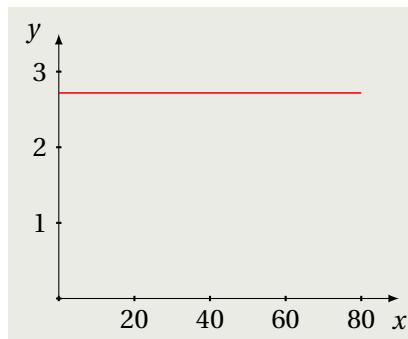
options	défaut	définition
color	black	couleur du trait
lw	0.6pt	épaisseur du point
style	dashed	style du trait

Exemple n° 58 Ligne verticale $x = \frac{1}{3}$ 

```
\begin{tikzpicture}[scale=2]
  \tkzInit[xmax=3,ymax=2]
  \tkzGrid
  \tkzX%
  \tkzY%
  \tkzVLine[color=red,style=dashed]{1/3}
\end{tikzpicture}
```

macro n° 22 tracer une ligne horizontale `\tkzHLine`Syntaxe : `\tkzHLine[<local options>](<list of values>)`

voir les options pour les lignes verticales

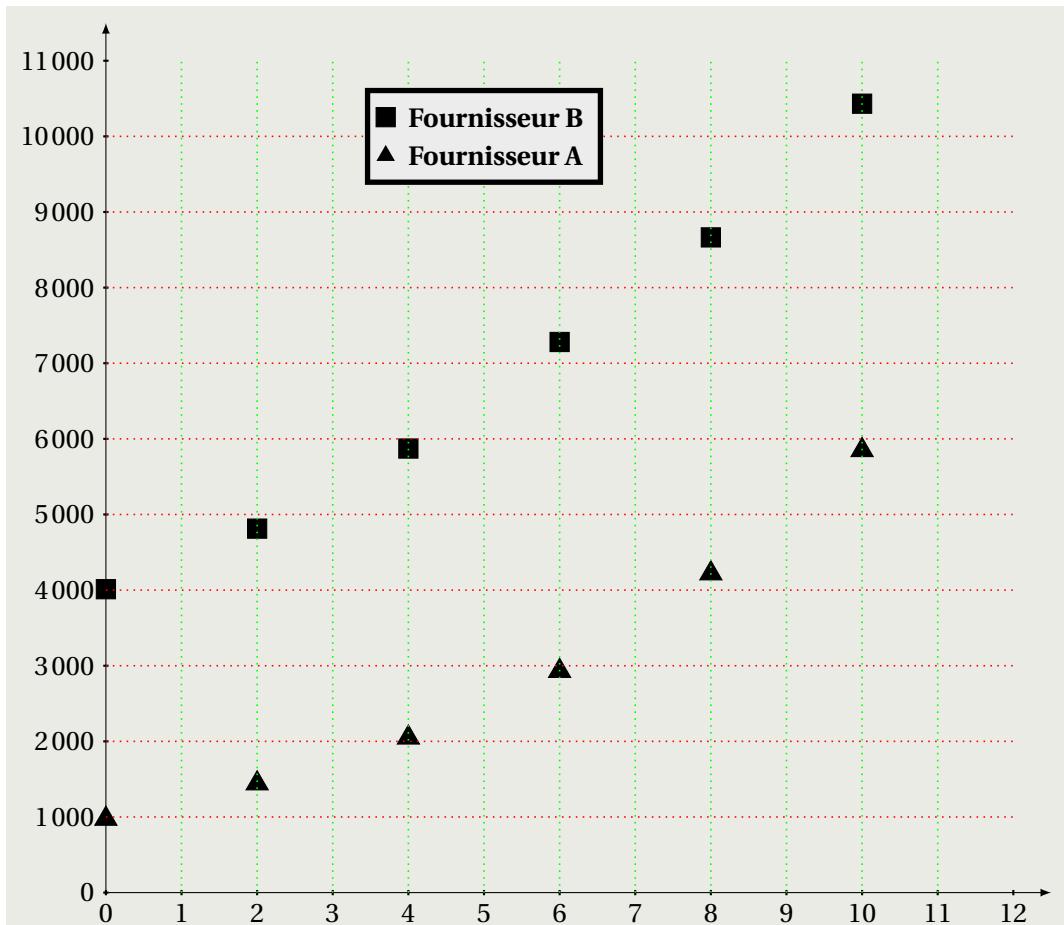
Exemple n° 59 Ligne horizontale $y = e$ 

```
\begin{tikzpicture}
  \tkzInit[xmax=80,xstep=20,ymax=3]
  \tkzX%
  \tkzY%
  \tkzHLine[color=red]{exp(1)}
\end{tikzpicture}
```

Exemple n° 60 Diagramme avec lignes verticales et horizontales

Utilisation d'une liste.

```
\tkzHLine[style=dotted,color=red]{1000,2000,...,10000}
```



```

\begin{tikzpicture}
\tkzInit[xmax=12,ymax=11000,ystep=1000]
  \tkzX[orig,label={}]
  \tkzY[orig,label={}]
\tkzDiagram[mark=triangle*,size=1ex]%
  {0/975,2/1443,4/2049,6/2930,8/4220,10/5850}
\tkzDiagram[mark=square*,size=.75ex]%
  {0/4012,2/4813,4/5872,6/7281,8/8664,10/10432}
\tkzHLine[style=dotted,color=red]{1000,2000,...,10000}
\tkzVLine[style=dotted,color=green]{1,2,...,11}
\tkzLegend[color=lightgray!30,lw=2pt]{5,10000}%
  {triangle*/1ex/black/\textbf{Fournisseur A},%
  square*/0.75ex/black/\textbf{Fournisseur B}}
\end{tikzpicture}

```

