

JeuxCartes

Des cartes de Poker ou de Tarot ou de Uno,
simples ou en *mains* ou en *mini*,
avec possibilité de tirage aléatoire.

Cédric Pierquet

c pierquet - at - outlook . fr

<https://github.com/cpierquet/JeuxCartes>

Version 0.2.1 – 27 Décembre 2022

► Quelques commandes pour afficher des cartes à jouer, de type Poker/Tarot/Uno

Une commande pour créer une carte individuelle (insertion autonome ou via TikZ).

Une commande pour créer une main de cartes, avec possibilité d’affichage en éventail.

Une commande pour créer une main aléatoire de cartes, avec possibilité d’affichage en éventail.

Une commande pour des cartes en version « miniatures » (individuelle, main ou aléatoire).



Merci aux membres du groupe  du « Coin L^AT_EX » pour leur aide et leurs idées!

L^AT_EX

pdfL^AT_EX

LuaL^AT_EX

TikZ

T_EXLive

MiK_TE_X

Table des matières

I	Historique	2
1	Le package JeuxCartes	3
1.1	« Philosophie » du package	3
1.2	Chargement du package	3
2	Compléments	4
2.1	Le système de « clés/options »	4
2.2	Liste des commandes et jeux disponibles	4
II	Les commandes	5
3	Les commandes simples « AffCarteJeu » et « AffCartesJeu »	5
3.1	Introduction	5
3.2	Noms de cartes	6
3.3	Clés et options	6
3.4	Designs disponibles pour les cartes classiques	8
4	Main de cartes	9
4.1	Introduction	9
4.2	Clés et options	9
5	Mains aléatoires	11
5.1	Introduction	11
5.2	Clés et options	11
6	Mini-Cartes	13
6.1	Introduction	13
6.2	Noms des <i>mini</i> -cartes	14
6.3	Clés et options	14
6.4	<i>Mini</i> -Mains et <i>mini</i> -mains aléatoires	14
III	Compléments	16
7	Cartes disponibles	16
7.1	Poker/Bataille/Belote (v1)	16
7.2	Poker/Bataille/Belote (v2)	16
7.3	Poker/Bataille/Belote (v3)	17
7.4	Cartes Tarot	17
7.5	Cartes de Uno	18
7.6	MiniCartes	18
8	Carte simple dans un environnement TikZ	19

Première partie

Historique

- v0.2.1: Correction du nom d'une carte pokerv1...
- v0.2.0: Modification du design des cartes (pdf pour Poker/Uno + png pour Tarot) + alternatives pour le Poker
- v0.1.4: Réduction de la taille des images png
- v0.1.3: Ajout de cartes Uno (CC1.0 par AlexDer) + modification des commandes + rami
- v0.1.2: Modification du nom (et de la source) des images de Tarot (CC0)
- v0.1.1: Ajout de commandes pour des mini-cartes
- v0.1 : Version initiale

Introduction

1 Le package JeuxCartes

1.1 « Philosophie » du package

💡 Idée(s)

L'idée de ce package est de proposer, comme le fait `pst-poker`, des commandes pour intégrer des cartes de poker dans un document \LaTeX .

Les cartes sont des images :

- les cartes type Poker v1 (Licence : LGPL-2.1 license) <https://github.com/htdebeer/SVG-cards>
- les cartes type Poker v2 (Licence : Public Domain) https://tekeye.uk/playing_cards/svg-playing-cards
- les cartes type Poker v3 (Licence : Public Domain) <https://www.me.uk/cards/>
- les cartes type Tarot (Licence : Public Domain) <https://freesvg.org/deck-of-french-tarot-playing-cards>
- les cartes type Uno (Licence : MIT License) <https://eperezcosano.github.io/uno-part1/>

🧩 Information(s)

Le package `JeuxCartes` charge les packages :

- `tikz`, `pifont`, `xfp`, `pgffor`, `xinttools`;
- `listofitems`, `xstring`, `xparse` et `simplekv`;
- `randomlist`, `xcolor` avec les options `[table,svgnames]`.

🧩 Information(s)

Des packages ou solutions existent déjà pour des cartes en \LaTeX , le lecteur choisira donc la solution qui lui semblera être la meilleure pour son utilisation!

On peut citer par exemple :

- le package `pst-poker` en langage `pstricks`;
- les fichiers `postscript` disponibles sur <https://melusine.eu.org/syracuse/postscript/cartes01/>

⚠ Attention

Les images utiles sont proposées en format pdf pour les cartes type Poker et Uno (obtenues grâce à des fichiers svg!), pour une qualité optimale et une taille réduite, et en format png pour les cartes Type Tarot, donc une solution de compilation adaptée aux formats pdf/png est nécessaire.

Les fichiers eps ne sont pas intégrées dans le package, mais elles sont disponible sur le dépôt github <https://github.com/cpierquet/JeuxCartes>.

Pour une compilation avec \LaTeX (et la chaîne `dvi-ps-ps2pdf`), il faudra donc enregistrer (*manuellement*) les images au format eps!

1.2 Chargement du package

🔗 Code \LaTeX

```
%exemple de chargement pour une compilation en pdflatex
\documentclass[french]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{JeuxCartes}
\usepackage{babel}
...
```

</> Code L^AT_EX

```
%exemple de chargement pour une compilation en (xe/lua)latex
\documentclass[french]{article}
\usepackage{mathtools}
\usepackage{fontspec}
\usepackage{JeuxCartes}
\usepackage{babel}
...
```

2 Compléments

2.1 Le système de « clés/options »

💡 Idée(s)

L'idée est de conserver – autant que faire se peut – l'idée de **⟨Clés⟩** qui sont modifiables et relativement explicites. À noter que :

- les **⟨Clés⟩** peuvent être mises dans n'importe quel ordre, et être omises lorsque la valeur par défaut est conservée;
- les arguments doivent, eux, être positionnés dans le *bon ordre*.

🧩 Information(s)

Les commandes présentées seront explicitées via leur syntaxe avec les options ou arguments.

Autant que faire se peut, des exemples/illustrations/remarques seront proposés à chaque fois.

Les codes seront présentés dans des boîtes </> Code L^AT_EX, si possible avec la sortie dans la même boîte, et sinon la sortie sera visible dans des boîtes ➕ Sortie L^AT_EX. Les clés ou options seront présentées dans des boîtes 🗝 Clés.

2.2 Liste des commandes et jeux disponibles

🧩 Information(s)

Les *Jeux* disponibles sont :

- **⟨Poker⟩**, **⟨Pokerv2⟩**, **⟨Pokerv3⟩**, **⟨Tarot⟩** et **⟨Uno⟩** pour l'affichage des cartes;
- **⟨Poker⟩**, **⟨Pokerv2⟩**, **⟨Pokerv3⟩**, **⟨Tarot⟩**, **⟨Bataille⟩**, **⟨Rami⟩**, **⟨Belote⟩** et **⟨Uno⟩** pour les mains aléatoires;
- **⟨Poker⟩** et **⟨Tarot⟩** pour les *mini*-cartes.

</> Code L^AT_EX

```
%Affichage d'une seule carte (mode image pure ou mode tikz)
\AffCarteJeu[...]{...}

%Affichage en mode côte à côte de plusieurs cartes
\AffCartesJeu[...]{...}

%Affichage en main de plusieurs cartes (éventail possible)
\MainCartesJeu[...]{...}

%Affichage en main de cartes aléatoires (éventail possible)
\MainCartesJeuAleatoire[...]{...}

%affichage d'une mini-carte (mode 'inline')
\AffMiniCarteJeu[...]{...}

%affichage d'une main de mini-cartes (mode 'inline')
\MainMiniCartesJeu[...]{...}

%affichage d'une main de mini-cartes aléatoires (mode 'inline')
\MainMiniCartesJeuAleatoire[...]{...}
```

Deuxième partie

Les commandes

3 Les commandes simples « AffCarteJeu » et « AffCartesJeu »

3.1 Introduction

Information(s)

La commande `\AffCarteJeu` affiche une carte, avec un système de clés/options.

La commande `\AffCartesJeu` affiche une liste de cartes ne mode *côte à côte*, avec un système de clés/options.

Code LaTeX

```
\AffCarteJeu{VP}\AffCarteJeu{10K}\AffCarteJeu{AC}\AffCarteJeu{Dos}\AffCarteJeu{DT}\\ \\
\AffCarteJeu[TypeJeu=Tarot]{VP}\AffCarteJeu[TypeJeu=Tarot]{Dos}
\AffCarteJeu[TypeJeu=Tarot]{Exc}\AffCarteJeu[TypeJeu=Tarot]{1AT}
\AffCarteJeu[TypeJeu=Tarot]{10K}\AffCarteJeu[TypeJeu=Tarot]{18AT}\\ \\
\AffCartesJeu[TypeJeu=Uno]{P4 § PTR § Coul § 7B § 8V}
```



3.2 Noms de cartes

Information(s)

Les cartes étant des **images**, leur nom est fixé suivant le modèle suivant (typiquement `<hauteur>.<couleur>`) avec quelques spécificités :

- cartes classiques pour Belote/Bataille/Poker :
 - 2C, 3C, ..., DC, RC, AC `<Cœur>`
 - 2P, 3P, ..., DP, RP, AP `<Pique>`
 - 2K, 3K, ..., DK, RK, AK `<Carreau>`
 - 2T, 3T, ..., DT, RT, AT `<Trèfle>`
 - JN, JR `<Joker>`
 - Dos . . . (si possible, avec variantes données en fin de documentation).
- cartes de Tarot :
 - 2C, 3C, ..., CC, ..., AC `<Cœur>`
 - 2P, 3P, ..., CP, ..., AP `<Pique>`
 - 2K, 3K, ..., CK, ..., AK `<Carreau>`
 - 2T, 3T, ..., CT, ..., AT `<Trèfle>`
 - Exc, 1AT, 2AT, ..., 20T, 21AT `<Atouts>`
 - Dos `<Dos>`
- cartes de Uno :
 - 0B, 1B, ..., 8B, 9B `<Bleu>`
 - 0R, 1R, ..., 8R, 9R `<Rouge>`
 - 0J, 1J, ..., 8J, 9J `<Jaune>`
 - 0V, 1V, ..., 8V, 9V `<Vert>`
 - P2B, P2R, P2B, P2J, P2V `<+2>`
 - PTB, PTR, PTB, PTJ, PTV `<PasseTour>`
 - CSB, CSR, CSB, CSJ, CSV `<ChangeSens>`
 - P4 `<+4>`
 - Cou1 `<JokerCouleur>`

3.3 Clés et options

Code L^AT_EX

```
\AffCarteJeu[options]{<carte>}
\AffCartesJeu[options]{<liste de cartes>}
```

Clés et options

Concernant les `<options>` :

- la clé `<Hauteur>` correspond à la hauteur voulue, en cm, de la carte; défaut `<4.25>`
- la clé `<TypeJeu>` (parmi `<PokerXXX>` ou `<Tarot>` ou `<Uno>`) pour spécifier les cartes à utiliser; défaut `<Poker>`
- la clé `<Rotation>` pour une éventuelle rotation de la carte; défaut `<0>`
- la clé `<AlignementV>` pour l'alignement vertical (lié à `raisebox` ou à `TikZ`); défaut `<0.5>`
- le booléen `<Tikz>` pour préciser que la carte est *autonome*, dans un environnement `tikzpicture`; défaut `<false>`
- la clé `<DecalageX>` pour le décalage horizontal par rapport à l'origine (si `<Tikz=false>`); défaut `<0>`
- la clé `<DecalageY>` pour le décalage vertical par rapport à l'origine (si `<Tikz=false>`); défaut `<0>`
- le booléen `<TikzAutonome>` pour basculer d'un environnement `TikZ` à une commande `TikZ`. défaut `<true>`

Information(s)

Concernant la clé `<AlignementV>`, elle peut être :

- donnée sous forme décimale entre 0 et 1 pour l'utilisation standard ;
- choisie parmi 0, 0.5 et 1 pour l'utilisation via le mode `<Tikz>`.

Information(s)

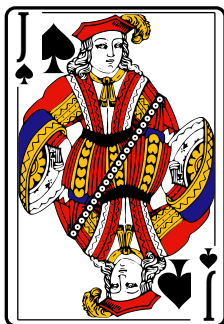
Concernant la clé `<TikzAutonome>` est plutôt une clé utilisée *interne*, elle peut être utilisée pour intégrer une carte dans un environnement TikZ déjà créé, pour une utilisation complètement personnalisé des cartes !

Il est à noter que la commande (en *interne*) est liée à un *sous-environnement* `scope`.

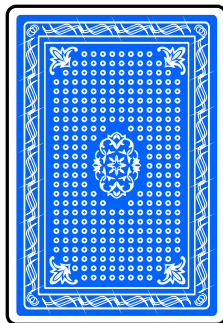
Code LaTeX

En mode normal, `\AffCarteJeu{VP}` ou `\AffCarteJeu[AlignementV=0]{Dos}` ou `\AffCarteJeu[AlignementV=1]{10K}` ou `\AffCarteJeu[TypeJeu=Tarot,AlignementV=0.33,Rotation=25]{10AT}`. \\ \\
 En mode `\TikZ`, `\AffCarteJeu[Hauteur=2,Tikz]{VP}`
 ou `\AffCarteJeu[Hauteur=2,AlignementV=0,Tikz]{AC}`
 ou `\AffCarteJeu[Hauteur=2,AlignementV=1,Tikz]{10K}`
 ou `\AffCarteJeu[Hauteur=2,TypeJeu=Tarot,AlignementV=0.33,Rotation=-25]{10AT}`. \\ \\
 On obtient la main `\AffCartesJeu[Hauteur=2,TypeJeu=Uno]{9B § 8R § P2J § P4 § 0J § CSR § PTJ}`.

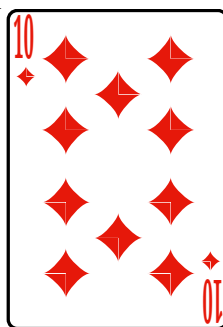
En mode normal,



ou



ou



ou



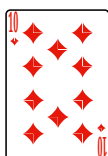
En mode TikZ,



ou



ou



ou



On obtient la main



3.4 Designs disponibles pour les cartes classiques

Information(s)

Il existe 3 versions de design des cartes *classiques*, accessibles via les clés :

- `<Poker>`;
- `<PokerV2>`;
- `<PokerV3>`.

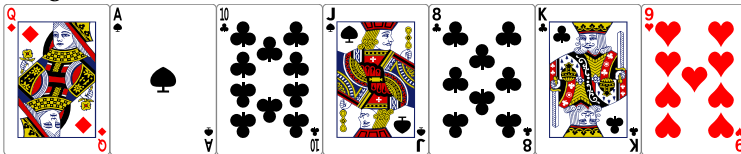
Code L^AT_EX

```
\textbf{Design v1} : \\
\AffCartesJeu[Hauteur=2]{DK § AP § 10T § VP § 8T § RT § 9C} \\ \\
\textbf{Design v2} : \\
\AffCartesJeu[Hauteur=2,TypeJeu=PokerV2]{DK § AP § 10T § VP § 8T § RT § 9C} \\ \\
\textbf{Design v3} : \\
\AffCartesJeu[Hauteur=2,TypeJeu=PokerV3]{DK § AP § 10T § VP § 8T § RT § 9C} \\ \\
```

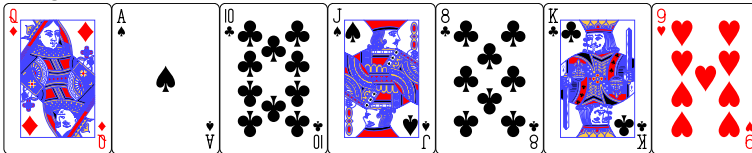
Design v1 :



Design v2 :



Design v3 :



4 Main de cartes

4.1 Introduction

💡 Idée(s)

La commande précédente, relativement simple, ne fait *que* placer une image dans un le document, il sera peut-être plus intéressant d'utiliser les commandes qui suivent, afin de créer des *main*s de cartes, aléatoires ou non!

⚠ Attention

Les commandes pour créer des *main*s sont obligatoirement liées à des environnement `tikzpicture` autonomes!

🔗 Code L^AT_EX

```
\MainCartesJeu[options]{<liste de cartes>}
```

4.2 Clés et options

🔗 Clés et options

L'argument *mandataire* correspond à la liste des cartes à afficher, sous la forme

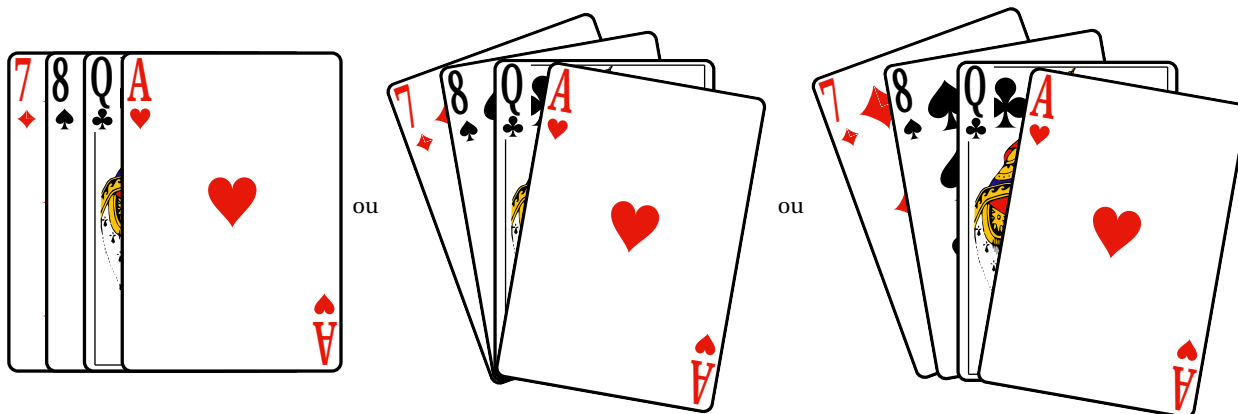
— <carte1> \$ <carte2> \$... \$ <carten>.

Les <options> sont les suivantes :

- la clé <Hauteur> correspond à la hauteur voulue, en cm, des cartes; défaut <4.25>
- la clé <TypeJeu> (parmi <PokerXXX> ou <Tarot> ou <Uno>) pour spécifier les cartes à utiliser; défaut <Poker>
- la clé <EspH> correspond à coefficient d'espacement horizontal (*proche* du cm en taille réelle) entre les cartes; défaut <1>
- la clé <EspV> correspond à coefficient d'espacement vertical (*proche* du cm en taille réelle) entre les cartes; défaut <0>
- le booléen <Eventail> pour une présentation en éventail; défaut <false>
- la clé <Rotation> pour l'angle entre les cartes en mode <Eventail>; défaut <10>
- la clé <AlignementV> pour l'alignement vertical lié à TikZ; défaut <0.5>

🔗 Code L^AT_EX

```
\MainCartesJeu{7K $ 8P $ DT $ AC}  
~ou \MainCartesJeu[Eventail,EspH=0,EspV=0.1]{7K $ 8P $ DT $ AC}  
~ou \MainCartesJeu[Eventail,EspH=0.5]{7K $ 8P $ DT $ AC}
```



📄 Information(s)

Pour une main *horizontale*, seule la clé <EspH> est importante, et une valeur proche de 1 donne des résultats intéressants.

Pour une main *éventail*, les deux espacements peuvent être modifiés, afin d'avoir un rendu satisfaisant pour l'utilisateur.

Les exemples proposés dans cette documentation permettent de se rendre compte de valeurs possibles pour un rendu satisfaisant.

```
\MainCartesJeu[TypeJeu=Uno,EspH=1.75]%
```

```
{9B $ P4 $ P2J $ 1R $ 7R $ PTV $ 2J $ 5J $ 9B $ PTR $ 5V}
```

```
\smallskip
```

```
\MainCartesJeu[TypeJeu=Tarot,EspH=1,EspV=-0.15,Hauteur=2.25]%
```

```
{Exc $ 1AT $ CC $ 8T $ 2K $ 5AT $ 2AT $ DP $ 7T $ 10C $ 19AT $ VP}
```

```
\smallskip
```

Ça c'est une belle poignée ! \MainCartesJeu[Eventail,Hauteur=3,TypeJeu=Tarot,EspH=0,EspV=0.1]%

```
{Exc $ 1AT $ 2AT $ 3AT $ 4AT $ 5AT $ 6AT $ %
```

```
10AT $ 11AT $ 15AT $ 16AT $ 19AT $ 20AT $ 21AT}
```

```
\smallskip
```

Et un chien qui en a : \MainCartesJeu[Eventail,Hauteur=2,TypeJeu=Tarot,EspH=0.5,EspV=0.15,Rotation=20]%

```
{DK $ 10AT $ 16AT $ VP $ 1AT}
```



Ça c'est une belle poignée!



Et un chien qui en a :



5 Mains aléatoires

5.1 Introduction

💡 Idée(s)

L'idée est ici de proposer une commande, similaire à la précédente, mais qui permet de **tirer au hasard** un certain nombre de cartes pour créer une main.

</> Code L^AT_EX

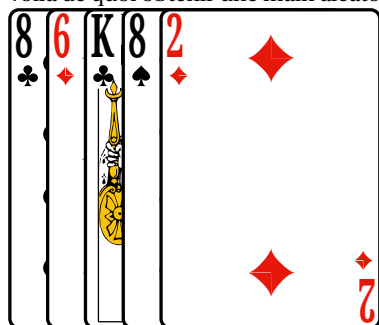
```
\MainCartesJeuAleatoire[<options>]{<nombre de cartes>}
```

</> Code L^AT_EX

Voilà de quoi obtenir une main aléatoire de Poker :

```
\MainCartesJeuAleatoire{5}
```

Voilà de quoi obtenir une main aléatoire de Poker :



5.2 Clés et options

🔑 Clés et options

Les **<options>** sont les suivantes :

- la clé **<Hauteur>** correspond à la hauteur voulue, en cm, des cartes; défaut **<4.25>**
- la clé **<TypeJeu>** (parmi **<PokerXXX>** ou **<Tarot>** ou **<Belote>** ou **<Bataille>** ou **<Rami>** ou **<Uno>**); défaut **<Poker>**
- la clé **<EspH>** correspond à coefficient d'espacement horizontal (*proche* du cm en taille réelle) entre les cartes; défaut **<1>**
- la clé **<EspV>** correspond à coefficient d'espacement vertical (*proche* du cm en taille réelle) entre les cartes; défaut **<0>**
- le booléen **<Eventail>** pour une présentation en éventail; défaut **<false>**
- la clé **<Rotation>** pour l'angle entre les cartes en mode **<Eventail>**; défaut **<10>**
- la clé **<AlignementV>** pour l'alignement vertical lié à TikZ; défaut **<0.5>**

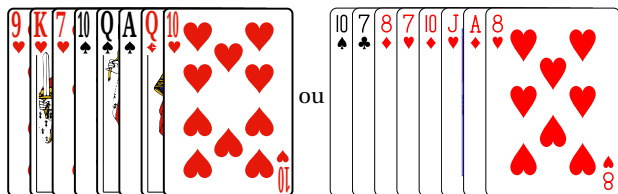
📌 Information(s)

En ce qui concerne les *jeux* disponibles (non modifiables) :

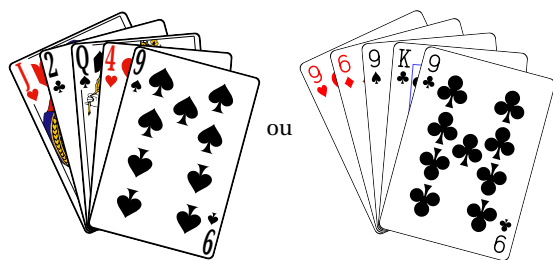
- **<Poker>** ou **<PokerV2>** ou **<PokerV3>** : 52 cartes (sans Joker);
- **<Belote>** ou **<BeloteV2>** ou **<BeloteV3>** : 32 cartes (sans Joker);
- **<Bataille>** ou **<BatailleV2>** ou **<BatailleV3>** : 54 cartes (avec Jokers);
- **<Rami>** ou **<Ramiv2>** ou **<Ramiv3>** : 108 cartes (avec Jokers);
- **<Tarot>** : 78 cartes;
- **<Uno>**.

```
\textbf{\large Belote v1 \& Belote v2 :} \\ \\
\MainCartesJeuAleatoire[Hauteur=2.5,TypeJeu=Belote]{8} ou
→ \MainCartesJeuAleatoire[Hauteur=2.5,TypeJeu=Belotev2]{8}.\\ \\
\textbf{\large Poker v1 \& Poker v3 :} \\ \\
\MainCartesJeuAleatoire[Hauteur=2.5,Eventail,EspH=0,EspV=0.1]{5} ou
→ \MainCartesJeuAleatoire[Hauteur=2.5,Eventail,EspH=0,EspV=0.1,TypeJeu=Pokerv3]{5}.\\ \\
\textbf{\large Bataille v1 :} \\ \\
\MainCartesJeuAleatoire[Hauteur=2.5,TypeJeu=Bataille]{27}\\ \\
\textbf{\large Tarot :} \\
\MainCartesJeuAleatoire[Hauteur=3.25,TypeJeu=Tarot]{10} ou~
\MainCartesJeuAleatoire[Hauteur=3.25,Eventail,TypeJeu=Tarot,EspH=0,EspV=0.1]{10}.
```

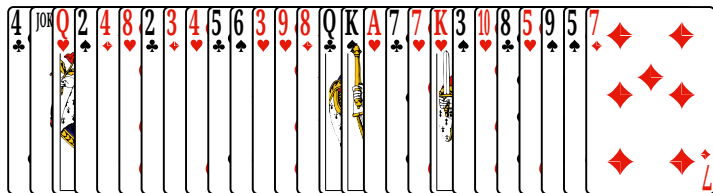
Belote v1 & Belote v2 :



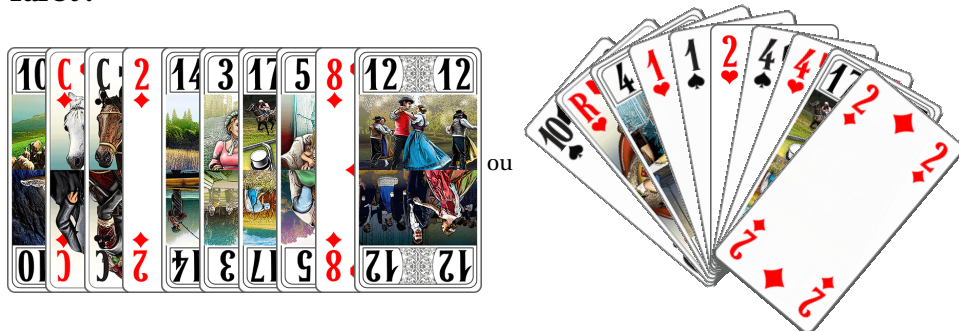
Poker v1 & Poker v3 :



Bataille v1 :



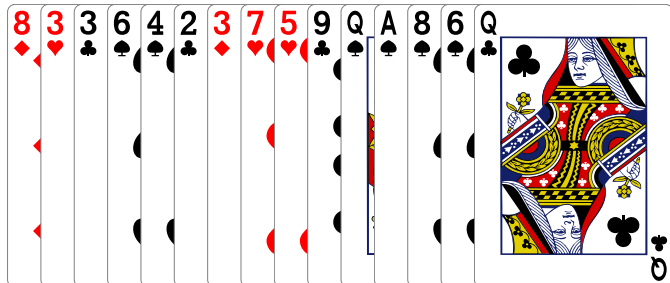
Tarot :



 Code \LaTeX

```
\textbf{\large Rami v2 :} \\ \\
\MainCartesJeuAleatoire[Hauteur=3.75,TypeJeu=Ramiv2]{15}\\ \\
\textbf{\large Uno :} \\ \\
\MainCartesJeuAleatoire[Hauteur=4,TypeJeu=Uno,Eventail,EspH=-0.1,Rotation=15,EspV=0.1]{7}
```

Rami v2 :



Uno :



6 *Mini-Cartes*

6.1 Introduction

 Idée(s)





L'idée est ici de proposer des commandes pour afficher des *mini-cartes*, utilisables en mode ligne, sans utiliser les images précédentes. Ces *mini-cartes* sont des figures TikZ, alignées verticalement sur leur *baseline*.

 Code \LaTeX

```
\AffMiniCarteJeu[<options>]{<carte>}
```

 Code \LaTeX

Si on met du texte sur la ligne du dessus, on peut voir le résultat.\\
Voilà des exemples de mini-cartes,
→ `\AffMiniCarteJeu{7.K}\AffMiniCarteJeu{1.AT}\AffMiniCarteJeu{V.K}\AffMiniCarteJeu{10.C}`, intégrables dans
→ un paragraphe.\\
Si on met du texte sur la ligne du dessous, on peut voir le résultat.

Si on met du texte sur la ligne du dessus, on peut voir le résultat.
Voilà des exemples de mini-cartes,    , intégrables dans un paragraphe.
Si on met du texte sur la ligne du dessous, on peut voir le résultat.

6.2 Noms des *mini*-cartes

Information(s)

Pour des raisons internes au code, les cartes doivent être saisies suivant la nomenclature (noter l'utilisation du  pour séparer la hauteur de la couleur!) :

- 2.C, 3.C, ..., C.C, D.C, R.C, A.C \langle Cœur \rangle
- 2.P, 3.P, ..., C.P, D.P, R.P, A.P \langle Pique \rangle
- 2.K, 3.K, ..., C.K, D.K, R.K, A.K \langle Carreau \rangle
- 2.T, 3.T, ..., K.T, D.T, R.T, A.T \langle Trèfle \rangle
- J.N, J.R \langle Joker \rangle
- Exc, 1.AT, 2.AT, ..., 20.T, 21.AT \langle Atouts \rangle

Par contre, il n'est pas prévu de *mini*-cartes pour le Uno.

Information(s)

Les *mini*-cartes d'atout pour le Tarot sont présentées avec un fond coloré, et avec un symbole en étoile.

6.3 Clés et options

Clés et options

Quelques \langle options \rangle pour les *mini*-cartes :

- la clé \langle Largeur \rangle pour gérer la largeur des miniatures; défaut \langle 0.55cm \rangle
- la clé \langle FondAtout \rangle pour gérer la couleur de fond pour les atouts. défaut \langle PeachPuff \rangle

Code LaTeX

```
\foreach \EECARTE in {2,3,4,5,6,7,8,9,10,V,C,D,R,A}{\AffMiniCarteJeu[Largeur=0.75cm]{\EECARTE.K}}\
\AffMiniCarteJeu[FondAtout=LightSkyBlue,Largeur=0.75cm]{Exc}%
\foreach \EECARTE in {1,2,...,21}{\AffMiniCarteJeu[FondAtout=LightSkyBlue,Largeur=0.75cm]{\EECARTE.AT}}
```



6.4 Mini-Mains et *mini*-mains aléatoires

Idée(s)

Comme pour les cartes *classiques*, il existe deux commandes pour des *mains* de *mini*-cartes, mais uniquement en présentation horizontale/côte-à-côte.

Code LaTeX

```
\MainMiniCartesJeu[<options>]{<liste de cartes>}
\MainMiniCartesJeuAleatoire[<options>]{<nb cartes>}
```

Clés et options

Les \langle Clés \rangle sont les mêmes que pour la commande individuelle, avec en plus :

- la clé \langle TypeJeu \rangle (parmi \langle Poker \rangle ou \langle Tarot \rangle ou \langle Belote \rangle ou \langle Bataille \rangle ou \langle Rami \rangle). défaut \langle Poker \rangle

```

\textbf{\large Saisie de mains : }\\
\MainMiniCartesJeu{7.K $ A.P $ D.T $ V.K $ 10.C $ C.T} et~
\MainMiniCartesJeu{Exc $ 1.AT $ C.C $ 8.T $ 2.K $ 5.AT $ 2.AT $ D.P $ 7.T $ 10.C $ 19.AT $ V.P}\\

\textbf{\large Poker : }\\
\MainMiniCartesJeuAleatoire{5} ou \MainMiniCartesJeuAleatoire{5} ou \MainMiniCartesJeuAleatoire{5} ou
- \MainMiniCartesJeuAleatoire{5}\\.\\

\textbf{\large Belote : }\\
\MainMiniCartesJeuAleatoire[TypeJeu=Belote]{8} ou \MainMiniCartesJeuAleatoire[TypeJeu=Belote]{8} ou
- \MainMiniCartesJeuAleatoire[TypeJeu=Belote]{8}\\.\\

\textbf{\large Bataille : }\\
\MainMiniCartesJeuAleatoire[TypeJeu=Bataille]{12} ou \MainMiniCartesJeuAleatoire[TypeJeu=Bataille]{12}\\.\\

\textbf{\large Tarot : }\\
\MainMiniCartesJeuAleatoire[TypeJeu=Tarot]{10} ou \MainMiniCartesJeuAleatoire[TypeJeu=Tarot]{10}\\.\\

\textbf{\large Rami : }\\
\MainMiniCartesJeuAleatoire[TypeJeu=Rami]{15} ou \MainMiniCartesJeuAleatoire[TypeJeu=Rami]{14}.

```

Saisie de mains :

7♦ A♠ D♣ V♦ 10♥ C♣ et EV♣ 1* C♥ 8♣ 2♦ 5* 2* D♣ 7♣ 10♥ 19* V♠

Poker :

V♣ D♥ 8♣ 7♥ 3♣ ou 9♥ 10♦ 10♣ 4♣ V♣ ou 10♦ 8♣ 3♣ 10♥ 10♣ ou 9♣ 2♦ 3♣ D♣ V♣.

Belote :

7♣ 7♦ 7♥ A♦ D♥ 8♣ V♠ A♥ ou R♣ R♥ D♦ 10♣ 10♦ 7♦ 10♣ 8♣ ou V♠ D♦ 9♣ 10♦ 7♣ V♥ 8♥ 7♦.

Bataille :

9♣ 3♣ 6♣ A♥ 9♥ 2♥ 8♥ 4♣ 10♥ A♠ 5♣ V♥ ou 2♦ 4♣ R♥ 6♥ 7♥ 5♦ R♣ 4♣ 9♣ R♦ 10♥ 9♣.

Tarot :

V♥ 6♦ R♦ 4♣ 8♣ 8♦ 15* R♣ 6♣ 6* ou 2* D♣ 7* 2♣ A♥ 5♦ 3♦ 9♣ 7♦ 5*.

Rami :

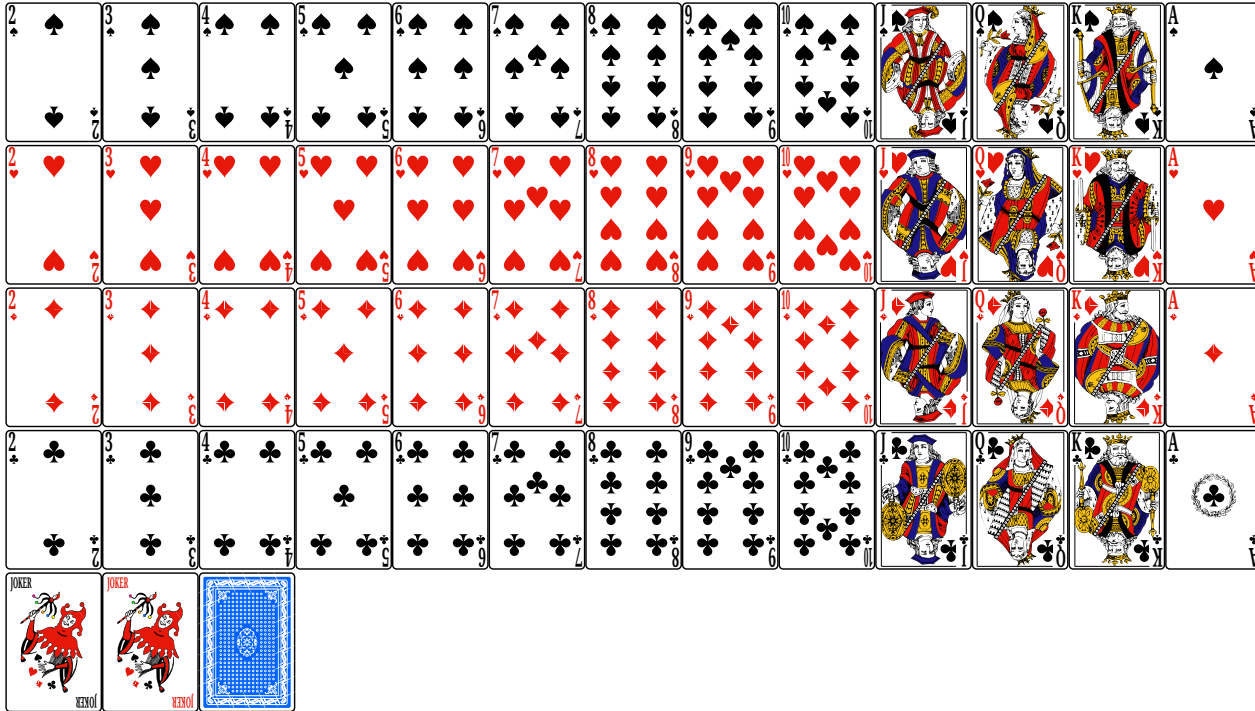
10♥ 6♥ 4♦ V♣ 5♣ A♠ 7♣ 9♦ 2♣ 5♥ 2♦ R♦ 10♣ 10♥ 10♣ ou 10♦ R♥ A♦ R♣ 2♦ V♣ 2♥ 9♣ 10♣ A♦ 2♣ R♣ V♦ V♥.

Troisième partie

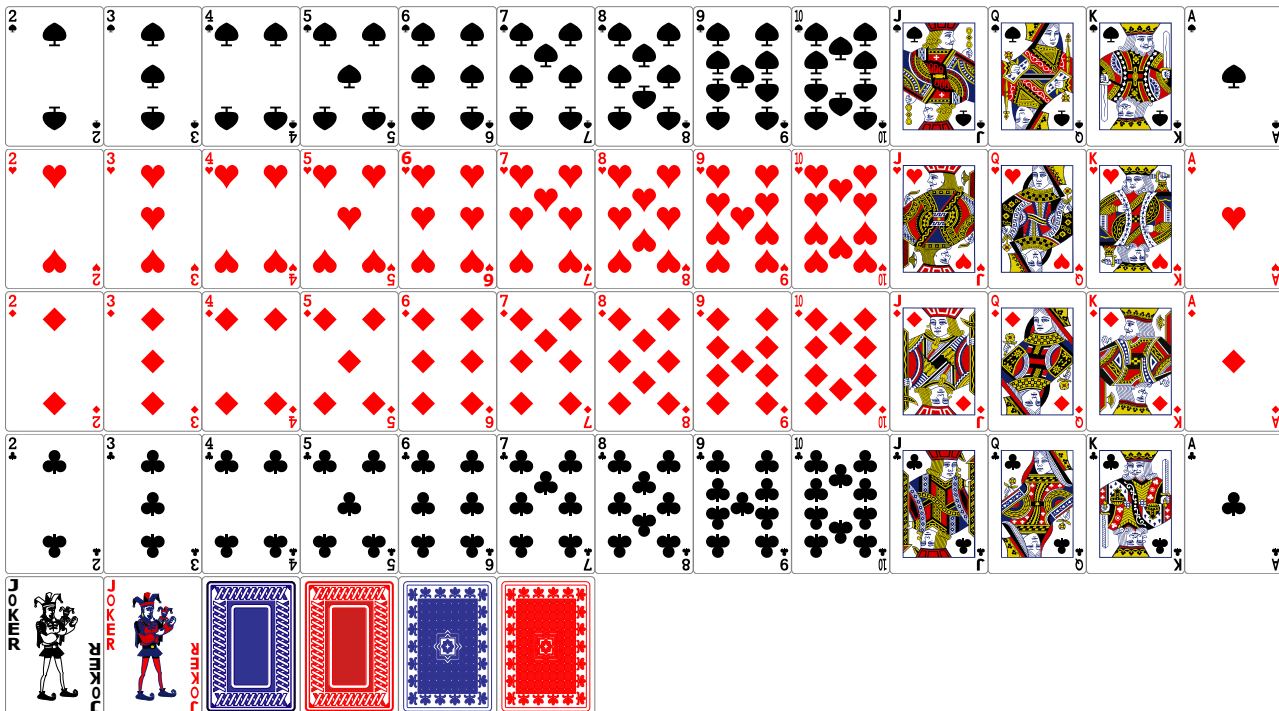
Compléments

7 Cartes disponibles

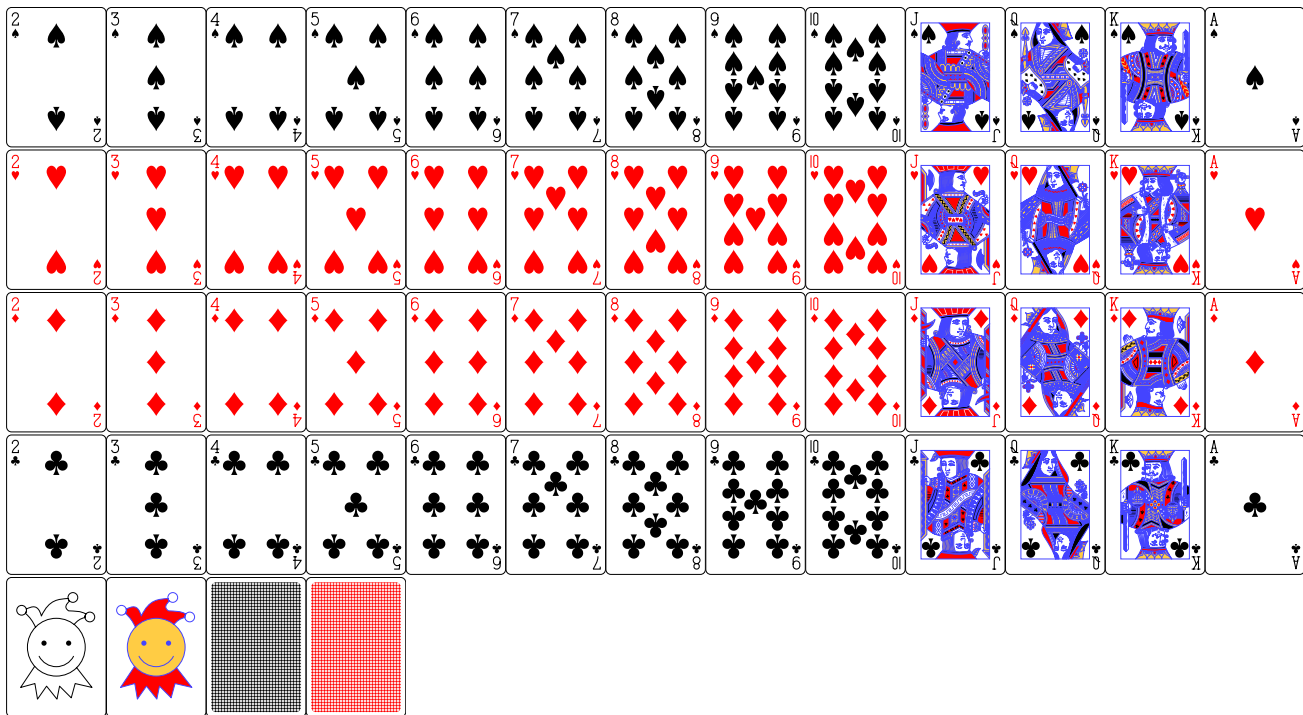
7.1 Poker/Bataille/Belote (v1)



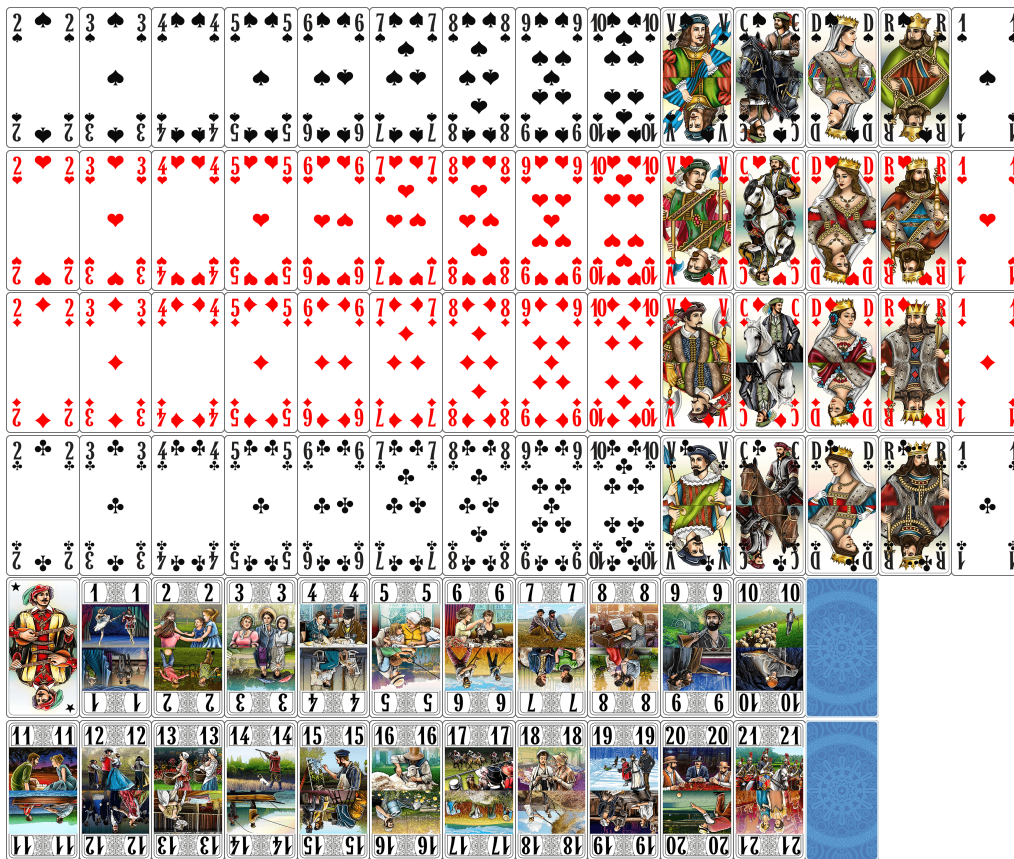
7.2 Poker/Bataille/Belote (v2)



7.3 Poker/Bataille/Belote (v3)



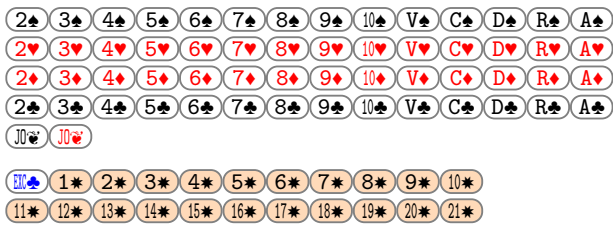
7.4 Cartes Tarot



7.5 Cartes de Uno



7.6 MiniCartes



8 Carte simple dans un environnement TikZ

💡 Idée(s)

La commande simple, avec l'option `<TikzAutonome=false>` permet à l'utilisateur de gérer les cartes dans un environnement `tikzpicture` indépendant.

L'origine des cartes est fixée, au coin bas-gauche, et la `<Rotation>` éventuelle est appliquée après la translation gérée par les `<DecalageXY>`.

🔗 Code L^AT_EX

```
\begin{tikzpicture}
  \draw[thin,lightgray] (-5,-5) grid (5,3) ; %grille rajoutée pour la lisibilité du code
  \foreach \ptsupport in {(0,0),(3,-4),(-3,-1),(-2,-3.5),(2,2.75),(-4.5,-4.5)}%
    {\filldraw[Gray] \ptsupport circle[radius=2pt] ;} %poits rajouté pour la lisibilité du code
  \foreach \x in {-5,-4,...,5} \draw[lightgray] (\x,-5) node[below,font=\small] {\x} ;
  \foreach \y in {-5,-4,...,3} \draw[lightgray] (-5,\y) node[left,font=\small] {\y} ;
  \AffCarteJeu[Hauteur=2,Tikz=true,TikzAutonome=false]%
    {10C}
  \AffCarteJeu[TypeJeu=Pokerv2,Hauteur=2.5,Tikz=true,TikzAutonome=false,DecalageX=3,DecalageY=-4]%
    {JN}
  \AffCarteJeu[Hauteur=3,Tikz=true,TikzAutonome=false,DecalageX=-3,DecalageY=-1,Rotation=20]%
    {JR}
  \AffCarteJeu[Hauteur=1.75,Tikz=true,TikzAutonome=false,DecalageX=-2,DecalageY=-3.5,Rotation=-50]%
    {Dos}
  \AffCarteJeu%
    [TypeJeu=Tarot,Hauteur=2.75,Tikz=true,TikzAutonome=false,DecalageX=2,DecalageY=2.75,Rotation=-90]%
    {18AT}
  \AffCarteJeu[TypeJeu=Uno,Hauteur=2,Tikz=true,TikzAutonome=false,DecalageX=-4.5,DecalageY=-4.5]%
    {P4}
\end{tikzpicture}
```

