

# Document beamer (présentation)

N. Carré

Louis-le-Grand – MPI

15 mars 2024



## 1 Introduction

- Schéma d'une diapo
- Pause

## 2 Quelques environnements supplémentaires

- Définition
- Théorème
- Du code
- Du pseudo-code

## 1 Introduction

- Schéma d'une diapo
- Pause

## 2 Quelques environnements supplémentaires

- Définition
- Théorème
- Du code
- Du pseudo-code

# Un titre

## Un contenu

- des items
- d'autres items

# Temporiser le contenu

Ce texte est sur la première diapo.

# Temporiser le contenu

Ce texte est sur la première diapo.

- ce texte est sur la suivante

# Temporiser le contenu

Ce texte est sur la première diapo.

- ce texte est sur la suivante
- ce texte arrive ensuite

# Temporiser le contenu

Ce texte est sur la première diapo.

- ce texte est sur la suivante
- ce texte arrive ensuite
- vous avez compris l'idée



# On peut ne mettre certains objets que sur certaines diapos

`\only` marche à l'intérieur d'autres environnements (par exemple `tikzpicture`).

- cet item va disparaître
- 
-

# On peut ne mettre certains objets que sur certaines diapos

`\only` marche à l'intérieur d'autres environnements (par exemple `tikzpicture`).

- 
- cet item restera
-

# On peut ne mettre certains objets que sur certaines diapos

`\only` marche à l'intérieur d'autres environnements (par exemple `tikzpicture`).

- 
- cet item restera
- dernier item

## ou cacher/faire apparaître

`\uncover` permet de faire ces modifications

- premier item

Ici, la différence avec `only` est que les objets sont présents mais invisibles.

## ou cacher/faire apparaître

`\uncover` permet de faire ces modifications

- premier item
- deuxième item

Ici, la différence avec `only` est que les objets sont présents mais invisibles.

## ou cacher/faire apparaître

`\uncover` permet de faire ces modifications

- deuxième item
- troisième item

Ici, la différence avec `only` est que les objets sont présents mais invisibles.

## 1 Introduction

- Schéma d'une diapo
- Pause

## 2 Quelques environnements supplémentaires

- Définition
- Théorème
- Du code
- Du pseudo-code

# Une définition

On a les définitions suivantes :

Définition 2.1

Une définition numérotée

Définition

Une définition non numérotée



Théorème 2.2

Un théorème numéroté

Attention, l'option [fragile] est nécessaire.

```
1 int main(void){
2     int x = 5; // Je pose x qui vaut 5.
3     printf("La_valeur_de_x_est_%d\n", x);
4     return EXIT_SUCCESS;
5 }
```

**Données** : Ce document

**Résultat** : Comment écrire un algorithme

On crée l'environnement

**tant que** *on n'est pas à la fin* **faire**

    On lit une phrase

**si** *on a compris* **alors**

$x \leftarrow x + 1$

        On passe à la section suivante.

**sinon**

**si** *on est perdu* **alors**

            On recommence depuis le début.

**Algorithme 1** : Comment écrire un algorithme

# Numéroter les lignes

**Données** : Ce document

**Résultat** : Comment écrire un algorithme

```
1 On crée l'environnement
2 tant que on n'est pas à la fin faire
3   | On lit une phrase
4   | si on a compris alors
5   |   |  $x \leftarrow x + 1$ 
6   |   | On passe à la section suivante.
7   | sinon
8   |   | si on est perdu alors
9   |   |   | On recommence depuis le début.
```

**Algorithme 2** : Comment écrire un algorithme