Être plus efficace grâce au hasard

Vincent Jugé

Stage olympique de Valbonne

25/08/2018

Y a-t-il deux élèves avec la même date d'anniversaire ?

Y a-t-il deux élèves avec la même date d'anniversaire ?

Oui ! Il y a même 9 paires d'élèves dans ce cas !

Paradoxe des anniversaires

Problème considéré

On a k boules numérotées de 1 à n. Quelle est la probabilité qu'au moins 2 boules aient le même numéro ?

Théorème

Si $k \gg \sqrt{n}$, c'est très probable !

Paradoxe des anniversaires

Problème considéré

On a k boules numérotées de 1 à n. Quelle est la probabilité qu'au moins 2 boules aient le même numéro ?

Théorème

Si $k \gg \sqrt{n}$, c'est très probable !

Esquisse de preuve

1. On met $k/2$ boules dans le groupe A et $k/2$ boules dans le groupe B.

Paradoxe des anniversaires

Problème considéré

On a k boules numérotées de 1 à n. Quelle est la probabilité qu'au moins 2 boules aient le même numéro ?

Théorème

Si $k \gg \sqrt{n}$, c'est très probable !

Esquisse de preuve

1. On met $k/2$ boules dans le groupe A et $k/2$ boules dans le groupe B.
2. Chaque boule du groupe B a une probabilité $\mathbb{P} = k/(2n)$ d'avoir le même numéro qu'une boule du groupe A.

Paradoxe des anniversaires

Problème considéré

On a k boules numérotées de 1 à n. Quelle est la probabilité qu'au moins 2 boules aient le même numéro ?

Théorème

Si $k \gg \sqrt{n}$, c'est très probable !

Esquisse de preuve (avec arnaque)

1. On met $k/2$ boules dans le groupe A et $k/2$ boules dans le groupe B.
2. Chaque boule du groupe B a une probabilité $\mathbb{P} = k/(2n)$ d'avoir le même numéro qu'une boule du groupe A.
3. On a donc une probabilité $k\mathbb{P}/2 = k^2/(4n) \gg 1$ d'avoir deux boules des groupes A et B de même numéro.

Paradoxe des anniversaires

Esquisse de preuve (sans arnaque)

1. On répartit les boules en des groupes $G_1, G_2, \ldots, G_{2\ell}$ de taille $\sqrt{n}/2$, avec $\ell = k/\sqrt{n} \gg 1$.

Paradoxe des anniversaires

Esquisse de preuve (sans arnaque)

1. On répartit les boules en des groupes $G_1, G_2, \ldots, G_{2\ell}$ de taille $\sqrt{n}/2$, avec $\ell = k/\sqrt{n} \gg 1$.
2. La probabilité que deux boules des groupes G_{2i} et G_{2i+1} n'aient jamais le même numéro vaut $\mathbb{P} = (1 - 1/(2\sqrt{n}))^{\sqrt{n}/2}$.

Paradoxe des anniversaires

Esquisse de preuve (sans arnaque)

1. On répartit les boules en des groupes $G_1, G_2, \ldots, G_{2\ell}$ de taille $\sqrt{n}/2$, avec $\ell = k/\sqrt{n} \gg 1$.
2. La probabilité que deux boules des groupes G_{2i} et G_{2i+1} n'aient jamais le même numéro vaut $\mathbb{P} = (1 - 1/(2\sqrt{n}))^{\sqrt{n}/2}$.
3. Si $v > u$, alors on montre par récurrence sur v que :
   1. $(1 - 1/v)^u \geqslant 1 - u/v$;

Preuve de l'hérédité

$(1 - 1/v)^u - (1 - 1/v)^{u+1} = (1 - 1/v)^u/v \leqslant 1/v$

Paradoxe des anniversaires

Esquisse de preuve (sans arnaque)

1. On répartit les boules en des groupes $G_1, G_2, \ldots, G_{2\ell}$ de taille $\sqrt{n}/2$, avec $\ell = k/\sqrt{n} \gg 1$.
2. La probabilité que deux boules des groupes G_{2i} et G_{2i+1} n'aient jamais le même numéro vaut $\mathbb{P} = (1 - 1/(2\sqrt{n}))^{\sqrt{n}/2}$.
3. Si $v > u$, alors on montre par récurrence sur v que :
   1. $(1 - 1/v)^u \geqslant 1 - u/v$;
   2. $(1 - 1/v)^u \leqslant 1 - u(1 - u/v)/v = 1 - u/v + (u/v)^2$.

Preuve de l'hérédité

$$(1 - 1/v)^u - (1 - 1/v)^{u+1} = (1 - 1/v)^u/v \geqslant (1 - u/v)/v$$

Paradoxe des anniversaires

Esquisse de preuve (sans arnaque)

1. On répartit les boules en des groupes $G_1, G_2, \ldots, G_{2\ell}$ de taille $\sqrt{n}/2$, avec $\ell = k/\sqrt{n} \gg 1$.
2. La probabilité que deux boules des groupes G_{2i} et G_{2i+1} n'aient jamais le même numéro vaut $\mathbb{P} = (1 - 1/(2\sqrt{n}))^{\sqrt{n}/2}$.
3. Si $v > u$, alors on montre par récurrence sur v que :
   1. $(1 - 1/v)^u \geqslant 1 - u/v$;
   2. $(1 - 1/v)^u \leqslant 1 - u(1 - u/v)/v = 1 - u/v + (u/v)^2$.
4. Donc $\mathbb{P} \leqslant 1 - 1/4 + 1/16 = 13/16$, et la probabilité que nos k boules aient toutes des numéros distincts vaut au plus $(1 - \mathbb{P})^\ell \ll 1$.

Cas pratique

Pour $n \approx 400$ et $k \approx 80$, on a $\ell \approx 4$ et $(1 - \mathbb{P})^\ell \leqslant (13/16)^4 \approx 0.44$.

Combien cette chambre compte-t-elle de jouets ?

Combien cette chambre compte-t-elle de jouets ?

Méthode n°1 : Ranger la chambre.

Temps nécessaire $\approx n$

Combien cette chambre compte-t-elle de jouets ?

Méthode n°1 : Ranger la chambre.
Temps nécessaire $\approx n$

Méthode n°2 : Tirer des jouets au hasard jusqu'à retomber sur le même.
Temps nécessaire $\approx \sqrt{n}$ grâce au **paradoxe des anniversaires** !

Combien cette chambre compte-t-elle de jouets ?

Méthode n°1 : Ranger la chambre.
Temps nécessaire $\approx n$

Méthode n°2 : Tirer des jouets au hasard jusqu'à retomber sur le même.
Temps nécessaire $\approx \sqrt{n}$ grâce au **paradoxe des anniversaires** !

Esquisse de preuve (sans arnaque)

1. Tirer k jouets revient à tirer k boules numérotées de 1 à n.

Combien cette chambre compte-t-elle de jouets ?

Méthode n°1 : Ranger la chambre.
Temps nécessaire $\approx n$

Méthode n°2 : Tirer des jouets au hasard jusqu'à retomber sur le même.
Temps nécessaire $\approx \sqrt{n}$ grâce au **paradoxe des anniversaires** !

Esquisse de preuve (sans arnaque)

1. Tirer k jouets revient à tirer k boules numérotées de 1 à n.
2. Après $k \approx \sqrt{n}$ tirages, on a tiré deux fois la même boule.

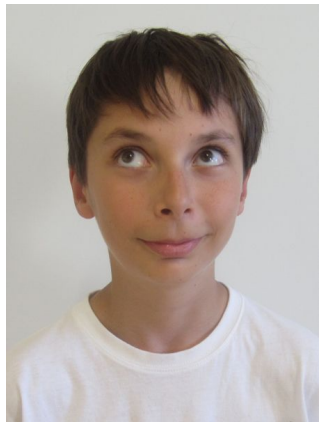
Combien cette chambre compte-t-elle de jouets ?

Méthode n°1 : Ranger la chambre.
Temps nécessaire $\approx n$

Méthode n°2 : Tirer des jouets au hasard jusqu'à retomber sur le même.
Temps nécessaire $\approx \sqrt{n}\log(n)$ grâce au **paradoxe des anniversaires** !

Esquisse de preuve (sans arnaque)

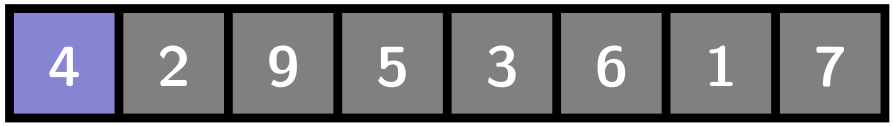
1. Tirer k jouets revient à tirer k boules numérotées de 1 à n.
2. Après $k \approx \sqrt{n}$ tirages, on a tiré deux fois la même boule.
3. Félix B. sait tester en temps $\log(n)$ si une boule figure déjà parmi les k premières que l'on a tirées.

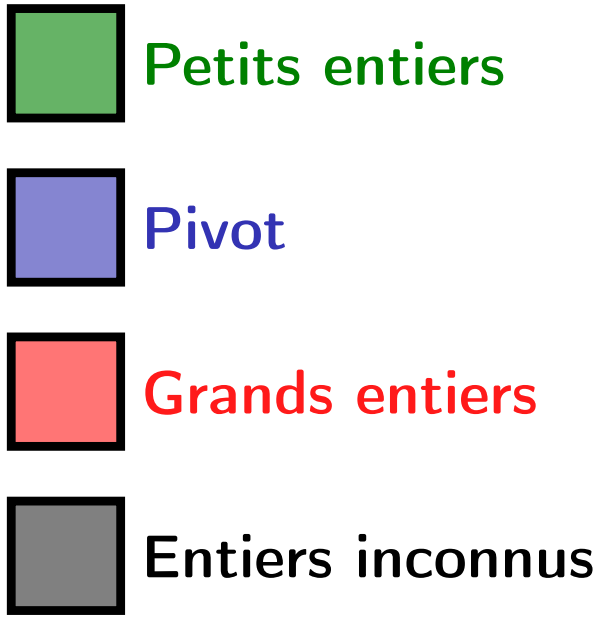


Trier efficacement des copies

Tri rapide

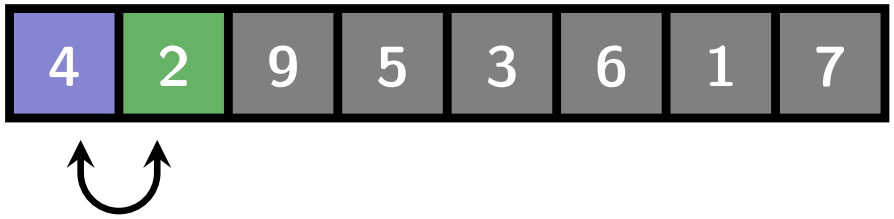
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

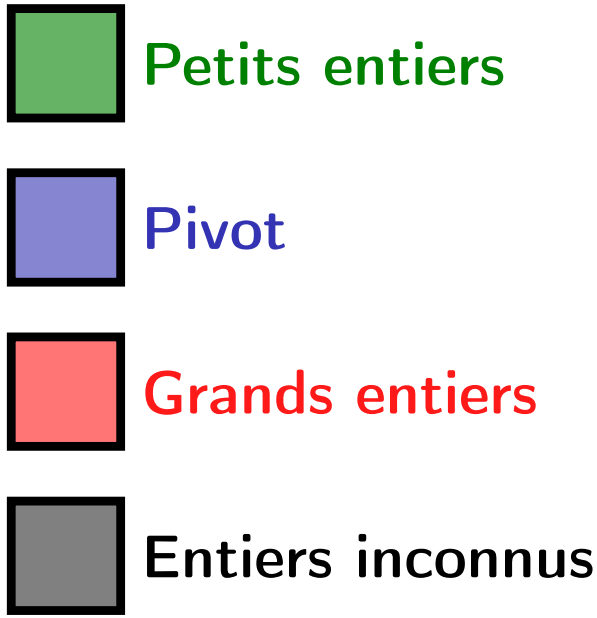




Tri rapide

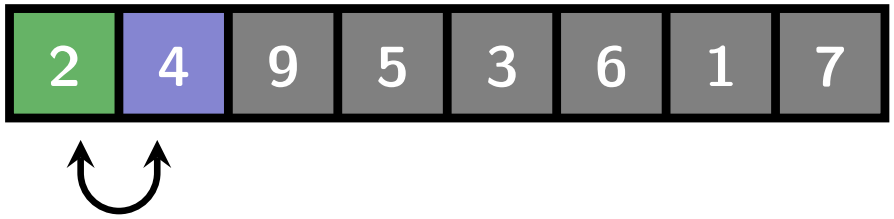
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

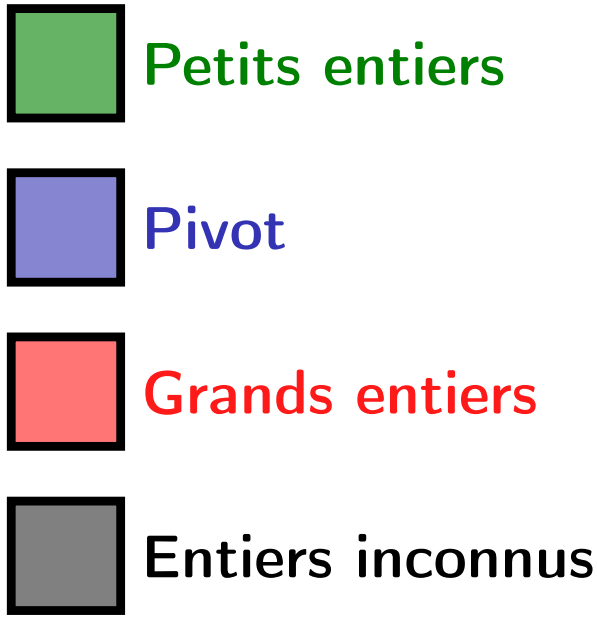




Tri rapide

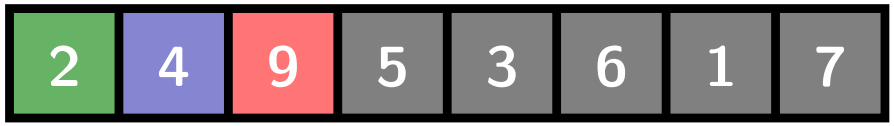
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

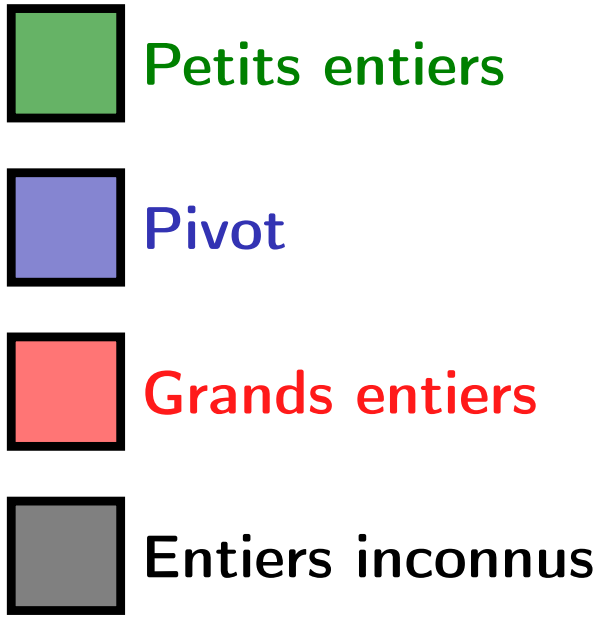




Tri rapide

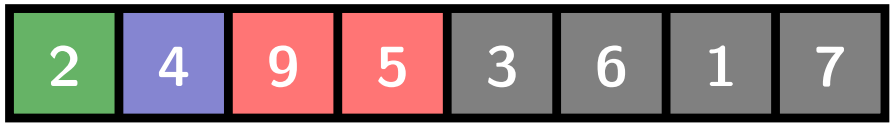
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

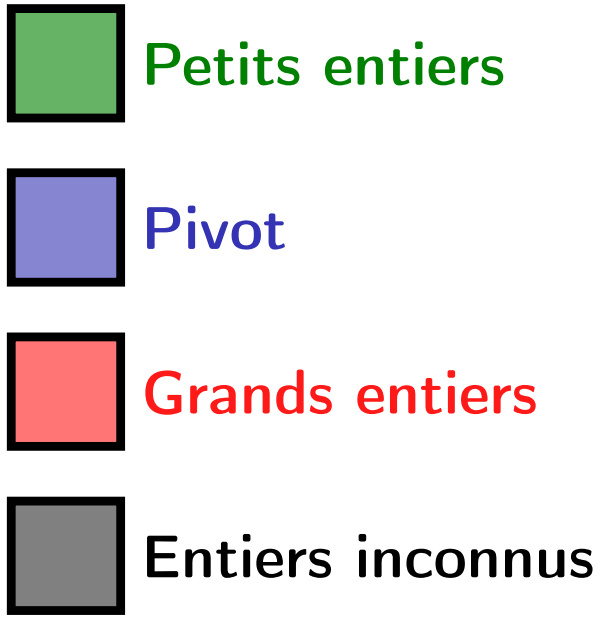




Tri rapide

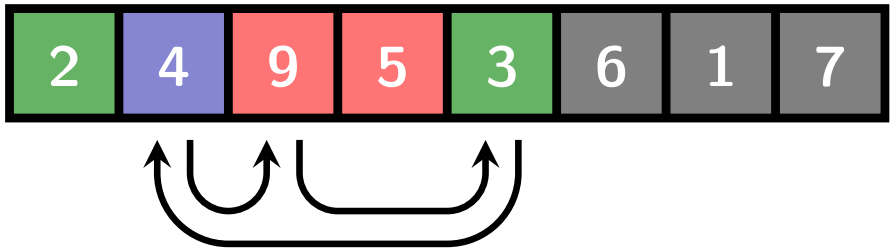
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

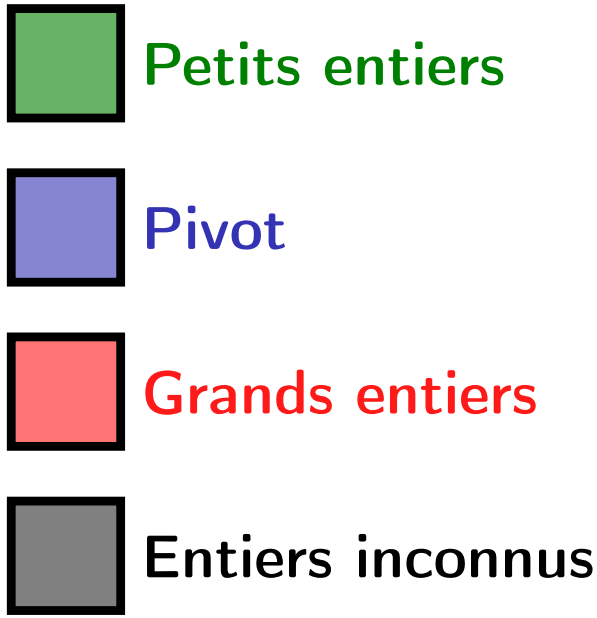




Tri rapide

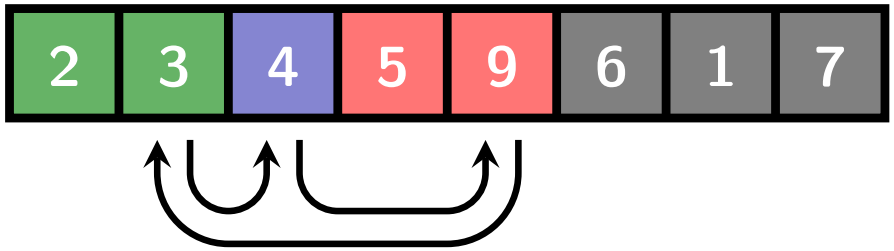
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.





Tri rapide

Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.



Petits entiers

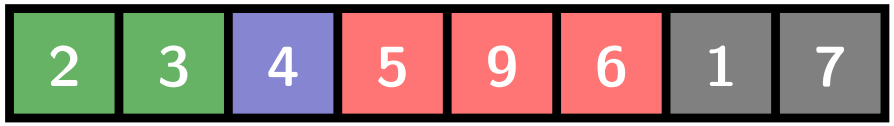
Pivot

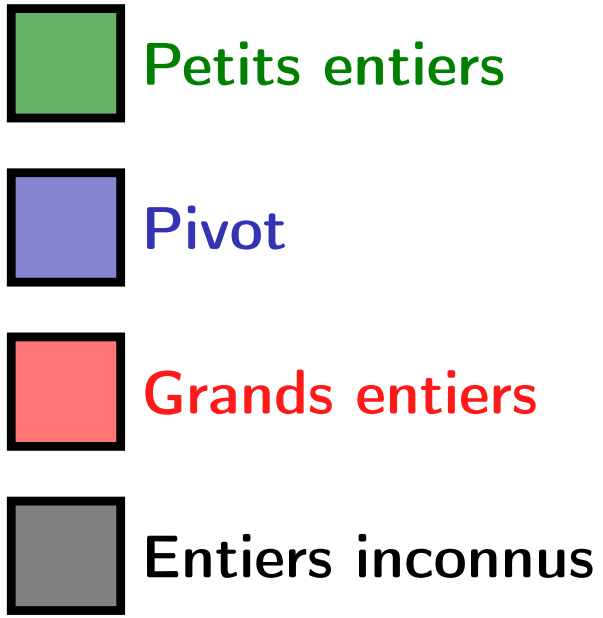
Grands entiers

Entiers inconnus

Tri rapide

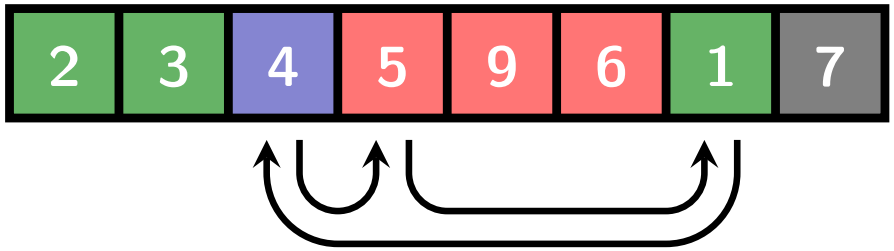
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

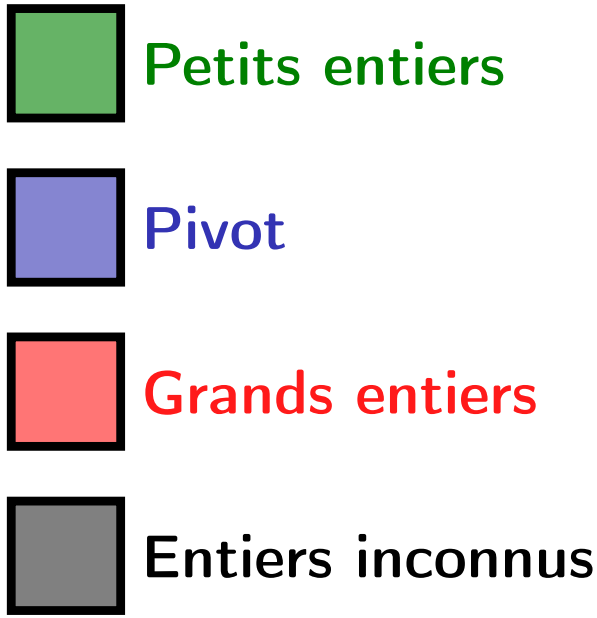




Tri rapide

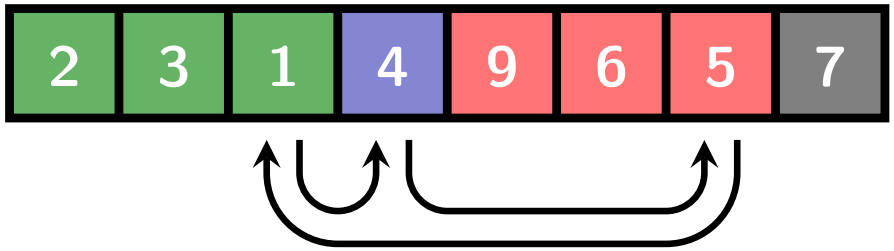
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

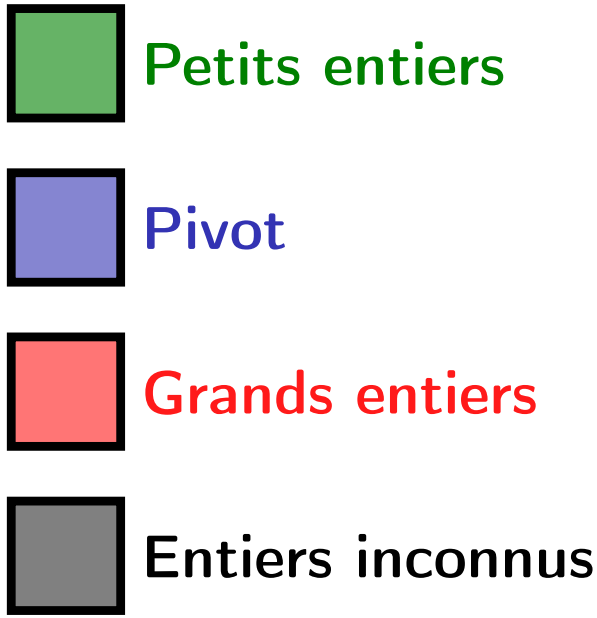




Tri rapide

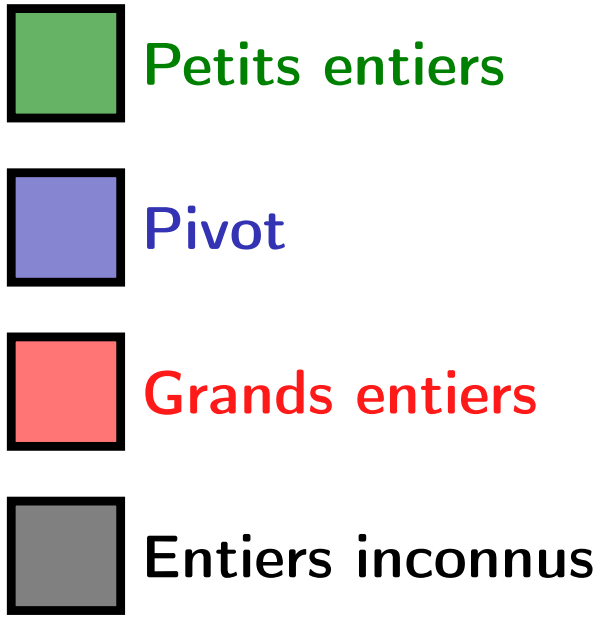
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.





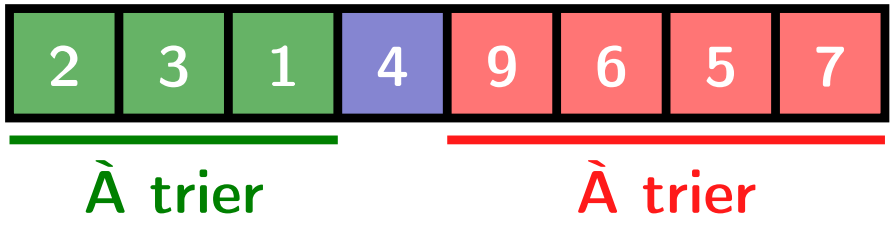
Tri rapide

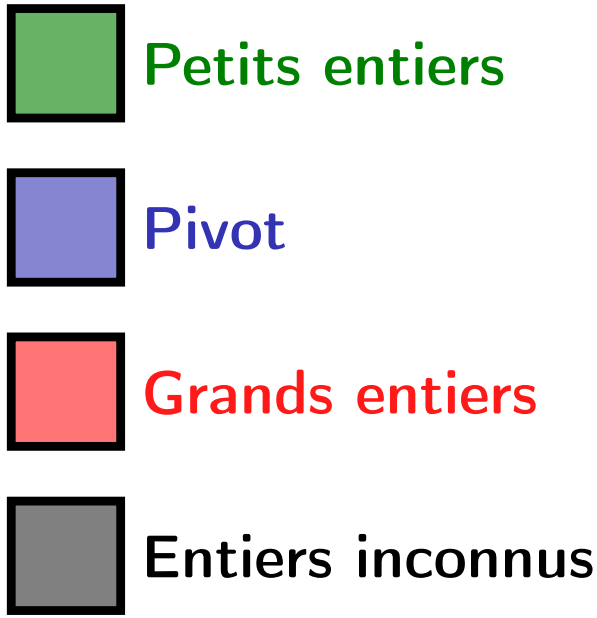
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.



Tri rapide

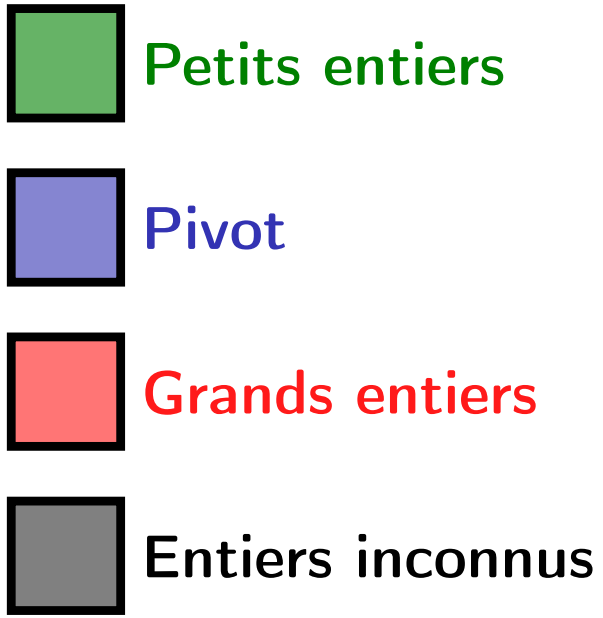
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.





Tri rapide

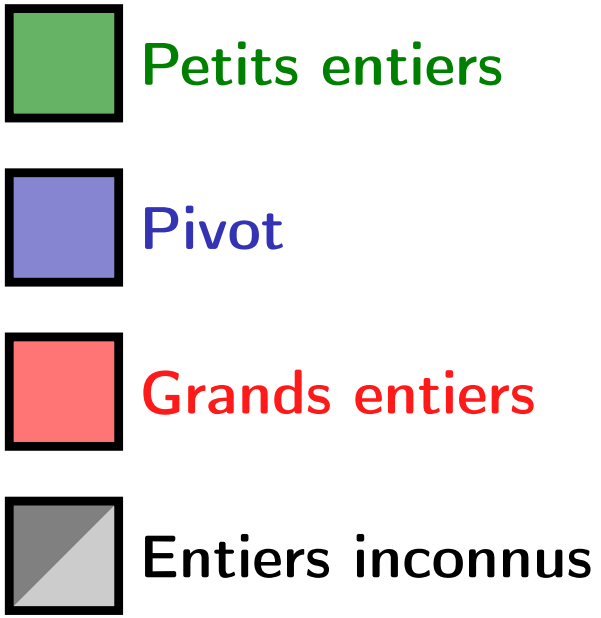
Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.



Tri rapide – Le bug de la copie oubliée

Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

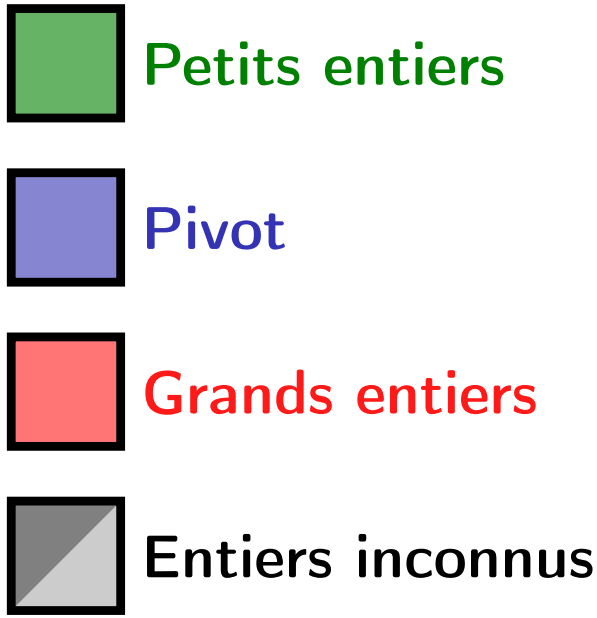
Complexité attendue $\approx n^2/4 \gg n \log(n)$



Tri rapide – Le bug de la copie oubliée

Un **algorithme** couramment utilisé pour trier un tableau est le **tri rapide**.
En moyenne, il utilise environ $n \log(n)$ opérations pour trier n entiers.

Complexité attendue $\approx n^2/4 \gg n \log(n)$



Ruses possibles pour contrer le bug

1. Choisir le pivot **au hasard** ! (facile)
2. **Mélanger** le tableau avant de le trier. (facile)
3. Choisir la **médiane** comme pivot. (très difficile à faire efficacement)

Preuve interactive de connaissance

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod{p}$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod{p}$ **sans découvrir de tel entier**.

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod p$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod p$ **sans découvrir de tel entier**.

Comment s'y prendre ?

Il suffit d'appliquer le protocole suivant :

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod{p}$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod{p}$ **sans découvrir de tel entier**.

Comment s'y prendre ?

Il suffit d'appliquer le protocole suivant :

1. Alice choisit un entier m **au hasard** et divulgue $g^m \pmod{p}$ à Bob.

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod p$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod p$ **sans découvrir de tel entier**.

Comment s'y prendre ?

Il suffit d'appliquer le protocole suivant :

1. Alice choisit un entier m **au hasard** et divulgue $g^m \pmod p$ à Bob.
2. Bob choisit **au hasard** un entier $\epsilon \in \{0, 1\}$ et demande $m + k\epsilon$ à Alice.

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod p$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod p$ **sans découvrir de tel entier**.

Comment s'y prendre ?

Il suffit d'appliquer le protocole suivant :

1. Alice choisit un entier m **au hasard** et divulgue $g^m \pmod p$ à Bob.
2. Bob choisit **au hasard** un entier $\epsilon \in \{0, 1\}$ et demande $m + k\epsilon$ à Alice.
3. Alice lui donne un entier n en guise de réponse.

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod p$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod p$ **sans découvrir de tel entier**.

Comment s'y prendre ?

Il suffit d'appliquer le protocole suivant :

1. Alice choisit un entier m **au hasard** et divulgue $g^m \pmod p$ à Bob.
2. Bob choisit **au hasard** un entier $\epsilon \in \{0, 1\}$ et demande $m + k\epsilon$ à Alice.
3. Alice lui donne un entier n en guise de réponse.
4. Bob vérifie que $g^n \equiv (g^k)^\epsilon g^m \pmod p$.

Preuve interactive de connaissance – logarithme discret

Cadre de travail

1. Alice a choisi trois entiers p, g et k.
2. Elle divulgue p, g et l'entier $g^k \pmod p$ à Bob.
3. Bob doit vérifier qu'Alice connaît bien **un** entier ℓ tel que $g^\ell \equiv g^k \pmod p$ **sans découvrir de tel entier**.

Comment s'y prendre ?

Il suffit d'appliquer le protocole suivant :

1. Alice choisit un entier m **au hasard** et divulgue $g^m \pmod p$ à Bob.
2. Bob choisit **au hasard** un entier $\epsilon \in \{0, 1\}$ et demande $m + k\epsilon$ à Alice.
3. Alice lui donne un entier n en guise de réponse.
4. Bob vérifie que $g^n \equiv (g^k)^\epsilon g^m \pmod p$.

Théorème : Si Alice ne connaît pas ℓ, le protocole échoue 1 fois sur 2.


MERCI POUR VOTRE ATTENTION !
NE POSEZ PAS DE QUESTIONS DIFFICILES S'IL VOUS PLAÎT !