Raisonnement par contraposition

Table des matières

1 Contraposition

En logique et en mathématiques, le **raisonnement par contraposition** est une technique très utile pour démontrer une implication.

Si l'on souhaite démontrer une implication de la forme : supposons que P soit une propriété qui implique Q

$$P \implies Q,$$

il est souvent plus simple de prouver la contraposée de cette implication, qui est :

$$\neg Q \implies \neg P.$$

La négation de P n'est, en bref, *pas* P. Les deux affirmations sont logiquement équivalentes, c'est-à-dire que si l'une est vraie, l'autre l'est aussi.

2 Pourquoi utiliser la contraposition ?

Prouver directement une implication $P \implies Q$ peut parfois être difficile. En revanche, démontrer que si Q est faux, alors P l'est aussi (c'est-à-dire $\neg Q \implies \neg P$) peut être plus simple. Cette méthode est appelée **raisonnement par contraposition**.

3 Exemple 1 : Nombre pair

On considère l'énoncé suivant :

Si n^2 est un nombre pair, alors n est un nombre pair.

Nous voulons démontrer l'implication :

$$P : n^2 \text{ est pair} \implies Q : n \text{ est pair.}$$

Plutôt que de démontrer cette implication directement, nous pouvons démontrer la contraposée :

$$\neg Q : n \text{ est impair} \implies \neg P : n^2 \text{ est impair.}$$

Démonstration

- Supposons que n soit impair. Par définition, cela signifie que $n = 2k + 1$ pour un entier k.
- Calculons n^2 :

$$n^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1.$$

- Donc n^2 est de la forme $2m + 1$, ce qui signifie que n^2 est impair.

Nous avons montré que si n est impair, alors n^2 est impair, ce qui prouve la contraposée et donc l'implication initiale.

4 Exemple 2 : Informatique

Dans le contexte de l'informatique, supposons que vous vouliez démontrer l'énoncé suivant :

> Si un programme est en temps constant, alors il ne dépend pas de la taille de l'entrée.

Ici, nous avons l'implication :

$$P : \text{Le programme est en temps constant} \implies Q : \text{Il ne dépend pas de la taille de l'entrée.}$$

Au lieu de démontrer directement $P \implies Q$, prouvons la contraposée :

$$\neg Q : \text{Le programme dépend de la taille de l'entrée} \implies \neg P : \text{Le programme n'est pas en temps consta}$$

Démonstration

- Supposons que le programme dépende de la taille de l'entrée. Cela signifie que le temps d'exécution augmente avec la taille de l'entrée.
- Puisque le temps d'exécution varie, il ne peut pas être constant.
- Ainsi, si le programme dépend de la taille de l'entrée, alors il n'est pas en temps constant.

Ceci prouve la contraposée et donc l'implication originale.

5 Conclusion

Le raisonnement par contraposition est un outil puissant, particulièrement utile lorsque prouver directement une implication est difficile. En informatique, cette méthode est fréquemment utilisée pour prouver des propriétés de programmes, d'algorithmes et de structures de données.