**Chapitre II**

**Propositions (assertions)**

**Définition** : Une proposition ( assertion) est une phrase soit vraie, soit fausse, pas les deux en même temps.

Exemples

* « Je suis au bureau»
* « 12- 2 =10 »
* « 2 × 3 = 7 »
* « Pour tout *x* I∈R, on a *x*2 > 0. »
* « Pour tout *z* ∈ C, on a |*z*| = 1. »

Si *P* est une proposition( assertion) et *Q* est une autre proposition, nous allons définir de nouvelles assertions construites à partir de P et de *Q*.

**L’opérateur logique « et »**

La proposition « *P* et *Q* » est vraie si *P* est vraie et *Q* est vraie. La proposition « *P* et *Q* » est fausse sinon. « *P* et *Q* » est notée .

« *P*  *Q* »

On résumé ceci en une table de vérité :

|  |  |  |
| --- | --- | --- |
| ***P*** | ***Q*** | ***P***  ***Q*** |
| V | V | V |
| F | V | F |
| V | F | F |
| F | F | F |

Table de vérité de « *P*  *Q* »

**Exemple:**

si *P* est la proposition ( l’assertion) « Khadija est une etudiante » et *Q* la proposition

«Aymen est un etudiant » alors la proposition « *P* et *Q* » est vraie si Aymen et Khadija sont des étudiants et est fausse si l'un des deux n'est pas un étudiant.

**L’opérateur logique « ou »**

La proposition « *P* ou *Q* » est vraie si l’une (au moins) des deux propositions *P* ou *Q* est vraie. La proposition « *P* ou *Q* » est fausse si les deux propositions *P* et *Q* sont fausses. « *P* ou *Q* » est notée « *P* V *Q* »

On reprend ceci dans la table de vérité :

|  |  |  |
| --- | --- | --- |
| ***P*** | ***Q*** | ***P*** V ***Q*** |
| V | V | V |
| F | V | V |
| V | F | V |
| F | F | F |

Table de vérité de « ***P*** ou ***Q*** »

Si *P* est la proposition( l’assertion) « *IN* est un ensemble dénombrable » et *Q* la proposition « *IR* est un ensemble dénombrable » alors la proposition « *P* ou *Q* » est vraie si *IN* est dénombrable ou bien *IR* est dénombrable .

**Remarque**. Pour définir les opérateurs « ou », « et » on fait appel à une phrase en utilisant les mots ou, et ! Les tables de vérités permettent d’éviter ce problème.

**La négation « non »**

La proposition « non *P* » est. vraie si *P* est. fausse, et fausse si *P* est vraie. « non *P* » est. notée 

|  |  |
| --- | --- |
| ***P*** |  |
| V | F |
| F | V |

Table de vérité de « »

**L’implication "⇒ "**

**Définition:**

La proposition « (non *P*) ou *Q* » est notée « *P* ⇒ *Q* ».

Sa table de vérité est donc la suivante :

|  |  |  |  |
| --- | --- | --- | --- |
| ***P*** | ***Q*** |  | ***P*** ⇒ ***Q*** (V*Q*) |
| V | V | F | V |
| V | F | F | F |
| F | V | V | V |
| F | F | V | V |

Table de vérité de « ***P*** ⇒ ***Q*** »

La proposition « *P* ⇒ *Q* » se lit « *P* implique *Q* ».

Elle se lit souvent aussi « si *P* est vraie alors *Q* est vraie » ou « si *P* alors *Q* ». Par

**Exemples :**

* «  » est vraie
* « *sinθ* = 0 ⇒ *θ* = 0 » est fausse (pour *θ* = 2π par exemple).
* « 24 = 8 ⇒ 2 = 3 » est vraie,car si *P* est fausse alors la proposition « *P* ⇒ *Q* » est toujours vraie.

**L’équivalence **

L’équivalence est définie par :

«  » est la proposition « (*P* ⇒ *Q*) et (*Q* ⇒ *P*) »

. On dira « *P* est équivalent à *Q* » ou « *P* équivaut à *Q* » ou « *P* si et seulement si *Q* ». Cette proposition est vraie lorsque *P* et *Q* sont vraies ou lorsque *P* et *Q* sont fausses.

|  |  |  |
| --- | --- | --- |
| *P* | *Q* |  |
| V | V | V |
| F | F | V |
| V | F | F |
| F | V | F |

Table de vérité de « »

Exemples :

• Pour , l’équivalence « ) » est vraie.

**Remarque**:

* La proposition « »est toujours fausse (quelle que soit la proposition *P*)
* On s’intéresse davantage aux propositions vraies qu’aux fausses,
* dans la pratique et en general on écrira « » ou « » uniquement lorsque ce sont des propositions vraies.
* si l’on écrit « » cela sous-entend « est vraie ».
* Attention rien ne dit que *P* et *Q* soient vraies. Cela signifie que *P* et *Q* sont vraies en même temps ou fausses en même temps.

**Proposition 1**:

Soient *P,Q, R* trois assertions. Nous avons les équivalences (vraies) suivantes :

1. *P*  non(non(*P*))

2. (*P* et *Q*)  (*Q* et *P*)

3. (*P* ou *Q*)  (*Q* ou *P*)

4. non(*P* et *Q*)  (non *P*) ou (non *Q*)

5. non(*P* ou *Q*)  (non *P*) et (non *Q*)

6. *P* et (*Q* ou *R*)  (*P* et *Q*) ou (*P* et *R*)

7. *P* ou (*Q* et *R*)  (*P* ou *Q*) et (*P* ou *R*)

8. « *P* ⇒ *Q* »  « non(*Q*) ⇒ non(*P*) »

**Démonstration:**

Montrons :

4 non(*P* et *Q*)  (non *P*) ou (non *Q*) [ ()]

Pour montrer cette equivalence , il suffit de comparer les deux propositions (assertions) « ) » et « () » pour toutes les valeurs possibles de *P* et *Q*. Par exemple en utilisant la table de virité si P est vrai et *Q* est vrai alors « *P* et *Q* » est vrai donc « non(*P* et *Q*) » est faux ; d’autre part (non *P*) est faux, (non *Q*) est faux donc « (non *P*) ou (non *Q*) » est faux. Ainsi dans ce premier cas les propositions sont toutes les deux fausses. On dresse ainsi les deux tables de vérités et comme elles sont égales les deux propositions sont equivalents

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***P*** | ***Q*** |  |  |  |  |  |
| V | V | F | F | V | F | F |
| F | F | V | V | F | V | V |
| V | F | F | V | F | V | V |
| F | V | V | F | F | V | V |

Table de vérité de « ****» et de « ****»

**Quantificateurs**

**Le quantificateur ∀ :**  « pour tout » La proposition(assertion) *P* peut dépendre d’un paramètre *x*, par exemple « *x* 2 > 1 », La proposition *P*(*x*) est vraie ou fausse selon la valeur de *x*. La proposition ∀*x* ∈ *E* *P*(*x*) est une proposition vraie lorsque les propositions ( assertions) *P*(*x*) sont vraies pour tous les éléments *x* de l’ensemble *E*. On lit « Pour tout *x* appartenant à *E*, *P*(*x*) », sous-entendu « Pour tout *x* appartenant à *E*, *P*(*x*) est vraie ».

**Exemples :**

* « ∀*x* ∈ [0,+∞[ (*x* 2 > 0) » est une proposition vraie.
* « ∀*x* ∈ *IR* (*x* < 5) » est une proposition fausse.
* « ∀*n* ∈ *IN* *n*(*n* + 1) est divisible par 2 » est vraie

**Le quantificateur ∃** : « il existe » La proposition ∃*x* ∈ *E* *P*(*x*) est une proposition (assertion) vraie lorsque l’on peut trouver au moins un *x* de *E* pour lequel *P*(*x*) est vraie. On lit « il existe *x* appartenant à *E* tel que *P*(*x*) (soit vraie) ».

**Exemples :**

* « ∃*x* ∈ *IR* : (*x* − 1) < 0) » est vraie (par exemple *x* = 0 vérifie bien la propriété).
* « ∃*y* ∈ ***Z*** : *y* – *y*3 < 1 » est vraie (il y a plein de choix, par exemple *y* = 2 convient, mais aussi *y* = 5 il suffit de trouver un seul suffit pour dire que la propositon est vraie).
* « ∃*x* ∈ *IN* (*x*+ 2 = 0) » est fausse (aucun nombre naturel si on l'additionne à 2 donnera un nombre négatif).

**La négation des quantificateurs**

La négation de « ∀*x* ∈ *E* *P*(*x*) » est « ∃*x* ∈ *E* non *P*(*x*) »

Par exemple la négation de « ∀*x* ∈ [1,+∞[ (*x* 2 > 1) » est la proposition « ∃*x* ∈ [1,+∞[ (*x* 2 < 1) ». En effet la négation de *x* 2 > 1 est non(*x* 2 > 1) mais s’écrit plus simplement

*x* 2 < 1.

La négation de « ∃*x* ∈ *E* *P*(*x*) » est « ∀*x* ∈ *E* non *P*(*x*) »

**Exemples :**

* La négation de « ∃*n* ∈ *IN* (*n* - 1 = 0) » est

« ∀*n* ∈ IN (*n* - 1 **** 0) ».

* La négation de « ∀*x* ∈ *IR* (*x* + 1 ∈ ***Z***) » est « ∃*x* ∈ *IR* (*x* + 1 **** ***Z***) ».
* La negation de ∀*x* ∈ *IR* ∃*y* **** (*x* + *y* > 5) est« ∃*x* ∈ *IR* ∀ *y* **** (*x* + *y* ****5) »

**Remarques**

L’ordre des quantificateurs est très important. Par exemple les deux phrases logiques :

∀*x* ∈ *IR* ∃*y* ∈ *IR* (*x* + *y* > 0) et

∃*y* ∈ *IR* ∀*x* ∈ *IR* (*x* + *y* > 0). sont différentes.

La première est vraie, la seconde est fausse. En effet une phrase logique se lit de gauche à droite, ainsi la première phrase affirme « Pour tout réel *x*, il existe un réel *y* (qui peut donc dépendre de *x*) tel que *x* + *y* > 0. » (par exemple on peut prendre *y* = |*x*| + 1). C’est donc une phrase vraie. Par contre la deuxième se lit : « Il existe un réel *y*, tel que pour tout réel *x*, *x* + *y* > 0. » Cette phrase est fausse, cela ne peut pas être le même *y* qui convient pour tous les *x* ! On retrouve la même différence dans les phrases suivantes. Voici une phrase vraie « Pour tout étudiant, il existe un numéro de matriculassions », bien sûr le numéro dépend de la personne. Par contre cette phrase est fausse : « Il existe un numéro de matriculassions, pour tous les étudiants ». Ce serait le même numéro pour tous les étudiants!

Remarques importantes.

* Quand on écrit « ∃*x* *∈* *IR* ( *f* (*x*) = 0) » cela signifie juste qu’il existe un réel pour lequel *f* s’annule on peut lire la phrase ainsi : « il existe au moins un réel *x* tel que *f* (*x*) = 0 ». pour préciser que *f* s’annule en une unique valeur, on écrit: ∃! *x* ∈ *IR* (*f* (*x*) = 0).
* Pour la négation d’une phrase logique, il n’est pas nécessaire de savoir si la phrase est fausse ou vraie. Le procédé est algorithmique : on change le « pour tout » en « il existe » et inversement, puis on prend la négation de la proposition *P*.
* Pour la négation d’une proposition, il faut être précis : la négation de l’inégalité stricte « < » est l’inégalité large « **** », et inversement.

Exercices:

1. Écrire la table de vérité du « ou exclusif ». (C’est le ou dans la phrase « fromage ou dessert », l’un ou l’autre mais pas les deux.)

2. Écrire la table de vérité de « non (*P* et *Q*) ». Que remarquez vous ?

3. Écrire la négation de « *P* ⇒ *Q* ».

4. Démontrer les assertions restantes de la proposition ?.

5. Écrire la négation de « *P* et (*Q* ou *R*) ».

6. Écrire à l’aide des quantificateurs la phrase suivante :

« Pour tout nombre réel, son carré est positif ». Puis écrire la négation.

7. Mêmes questions avec les phrases : « Pour chaque réel, je peux trouver un entier relative tel que leur produit soit strictement plus grand que 1 ». Puis « Pour tout entier *n*, il existe un unique réel *x* tel que **** ».

**2. Raisonnements**

**Des méthodes classiques de raisonnements.**

2.1. **méthode direct:**

Si on veut montrer que laproposition « *P* ⇒ *Q* » est vraie. On suppose que *P* est vraie et on montre qu’alors *Q* est vraie.

Exemple 1.

Montrer que si *a*, *b* ∈ *Q* alors *a* + *b* ∈ *Q*.

**Démonstration.**

Soient *a* ∈ *Q*, *b* ∈ *Q*. donc ils existent *p* ∈ **Z** et *q* ∈ IN ∗ et

****∈ **Z** et ****∈ IN ∗ telques *a* = **** et *b* = **** d'ou

*a* + *b* = ****+****= **** avec le numérateur est bien un élément de **Z**; le dénominateur  est un élément de . Donc *a* + *b* s’écrit bien de la forme *a* + *b* = avec ∈ Z, ∈ . Ainsi *a* + *b* ∈ *Q*

.**Méthode : Cas par cas**

La méthode du cas par cas la s'apelle aussi méthode de disjunction.

Si l’on souhaite vérifier une proposition  *P*(*x*) pour tous les *x* dans un ensemble *E*, on montre proposition pour les *x* dans une partie *A* de *E*, puis pour les *x* n’appartenant pas à *A*.

. **Exemple 2.**

Montrer que pour tout *x* ∈ IR,  Démonstration.

Soit *x* ∈ IR. Nous distinguons deux cas. Premier cas :

*x* = 1.Alors on obtient

Deuxième cas *x* > 1. Alors . Calculons alors 

 donc .

Troisième cas: x < 1. Alors . Nous obtenons 

Conclusion. Dans tous les cas 

**2.3. Contraposée**

Le raisonnement par contraposition est basé sur l’équivalence suivante (voir la proposition ?) : La propostion « *P* ⇒ *Q* » est équivalente à « non(*Q*) ⇒ non(*P*) ». Donc si l’on souhaite montrer l’assertion « *P* ⇒ *Q* », on montre en fait que si non(*Q*) est vraie alors non(*P*) est vraie.

Exemple 3. Soit *n* ∈ *IN*. Montrer que si *n* 2 est pair alors *n* est pair.

Démonstration. Nous supposons que *n* n’est pas pair. Nous voulons montrer qu’alors *n* 2 n’est pas pair. Comme *n* n’est pas pair, il est impair et donc il existe *k* ∈ *IN* tel que *n* = 2*k*+1. Alors *n* 2 = (2*k*+1) 2 = 4*k* 2+4*k*+1 = 2*l*+1 avec *l* = 2*k* 2 + 2*k* ∈ *IN*. Et donc *n* 2 est impair. Conclusion : nous avons montré que si *n* est impair alors *n* 2 est impair. Par contraposition ceci est équivalent à : si *n* 2 est pair alors n est pair.

**2.4. Absurde**

Le raisonnement par l’absurde pour montrer « *P* ⇒ *Q* » repose sur le principe suivant : on suppose à la fois que *P* est vraie et que *Q* est fausse et on cherche une contradiction. Ainsi si *P* est vraie alors *Q* doit être vraie et donc « *P* ⇒ *Q* » est vraie.

Exemple 4. Soient *a*, *b* > 0. Montrer que si .

Démonstration. Nous raisonnons par l’absurde en supposant que et .

Comme alors *a*(1+ *a*) = *b*(1+ *b*) donc *a* + *a* 2 = *b* + *b* 2 d’où *a* 2 − *b* 2 = *b* − *a*. Cela conduit à (*a* − *b*)(*a* + *b*) = −(*a* − *b*). Comme alors et donc en divisant par *a* − *b* on obtient *a* + *b* = −1. La somme des deux nombres positifs *a* et *b* ne peut être négative. Nous obtenons une contradiction.

Conclusion : si alors a = b.

Dans la pratique, on peut choisir indifféremment entre un raisonnement par contraposition ou par l’absurde. Attention cependant de bien préciser quel type de raisonnement vous choisissez et surtout de ne pas changer en cours de rédaction ! 2.5.

**Contre-exemple**

Si l’on veut montrer qu’une proposition du type « ∀*x* ∈ *E* *P*(*x*) » est vraie alors pour chaque *x* de *E* il faut montrer que *P*(*x*) est vraie. Par contre pour montrer que cette proposition est fausse alors il suffit de trouver *x* ∈ *E* tel que *P*(*x*) soit fausse. (Rappelez-vous la négation de

« ∀*x* ∈ *E* *P*(x) » est « ∃*x* ∈ *E* non *P*(*x*) ».) Trouver un tel *x* c’est trouver un contre-exemple à la proposition « ∀*x* ∈ *E* *P*(*x*) ».

Exemple 5. Montrer que la proposition suivante est fausse

«la fonction **** est paire

Démonstration. Un contre-exemple: on prend  ****

****

D’où **** alors  *f* pas paire

**Récurrence**

Le principe de récurrence permet de montrer qu’une proposition *P*(*n*), dépendant de *n*, est vraie pour tout *n* ∈IN. La démonstration par récurrence se déroule en trois étapes : lors de l’initialisation on prouve *P*(0). Pour l’étape d’hérédité, on suppose *n* > 0 donné avec *P*(*n*) vraie, et on démontre alors que l’assertion *P*(*n* + 1) au rang suivant est vraie. Enfin dans la conclusion, on rappelle que par le principe de récurrence *P*(*n*) est vraie pour tout *n* ∈*IN*.

**Exemple 6**. Montrer que pour tout *n* ∈*IN*, 2*n* > *n*.

**Démonstration**. Pour *n* > 0, notons P(*n*) l’assertion suivante : 2 *n* > *n*. Nous allons démontrer par récurrence que *P*(*n*) est vraie pour tout *n* > 0.

Initialisation. Pour *n* = 0 nous avons 20 = 1 > 0. Donc *P*(0) est vraie. Hérédité. Fixons *n* > 0. Supposons que *P*(*n*) soit vraie. Nous allons montrer

que *P*(*n* + 1) est vraie. 2 *n*+1 = 2*n* + 2*n* > n + 2*n* car par *P*(*n*) nous savons 2*n* > n, et on a 2 *n*+1 > *n* + 1 car 2*n* > 1. Donc *P*(*n* + 1) est vraie.

Conclusion. Par le principe de récurrence *P*(*n*) est vraie pour tout

*n* > 0, c’est-à-dire 2 *n* > *n* pour tout n > 0.

Remarques :

• Si on doit démontrer qu’une propriété est vraie pour tout *n* > *n*0 , alors on commence l’initialisation au rang *n*0 .

• Le principe de récurrence est basé sur la construction de l’ensemble *IN*. En effet un des axiomes pour définir *IN* est le suivant : « Soit *A* une partie de *IN* qui contient 0 et telle que si *n* ∈ *A* alors *n* + 1 ∈ *A*. Alors *A* = *IN* ».

**Formule atomique**

**Définition** Une formule propositionnelle est dite formule atomique si elle ne peut pas s’écrire en fonction d’autres formules propositionnelles à l’aide des connecteurs fondamentaux.

**Exemple** 1.7 La formule propositionnelle *A* : ”4 est un carré parfait" est une formule atomique.

La formule "4 est un carré parfait alors 4 est un nombre pair " n’est pas une formule atomique. Car si on pose: *B* "4 est un nombre pair " La formule "4 est un carré parfait alors 4 est un nombre pair " est: *A* ⇒ *B*

**Définition** Une formule propositionnelle se définit par :

i) Toute formule atomique est une formule propositionnelle.

ii) Si *A* est une formule alors  est aussi une formule.

iii) Si *A* et *B* sont des formules alors (*A* ∧ *B*) , (*A* ∨ *B*) , (*A* ⇒ *B*) sont aussi des formules.

iv) Rien d’autres n’est une formule.

**Système complet de connecteurs**

Définition : On dit qu’un ensemble de connecteurs **C** est un système complet de

connecteurs si toute formule propositionnelle est équivalente à une formule n’utilisant

que les connecteurs de **C**.

Proposition :L’ensemble {, ∨} est un système complet de connecteurs.

Preuve. Il suffit de montrer que les propositions (*A*∧*B*) , (*A* ⇒ *B*) , (*A* ⇔ *B*) peuvent

s’écrire en utilisant seulement les connecteurs: «, ∨»

i) On a (*A*∧*B*) ⇔ (∨) .

ii) On a (*A* ⇒ *B*) ⇔∨*B*.

## iii) On a (*A* ⇔ *B*) ⇔ (∨ *B*) ∧ (∨*A*) .

**Tautologie et Antilogie**

**Une tautologie**

Une tautologie est une proposition (ou énoncé) (ou une combinaison de propositions) qui est toujours vraie, quelle que soit la valeur des différentes propositions.

**Exemple** :

*A* : *x* > 2

*B* : *x* > 1

« *A* ⇒ *B* »est vraie pour toute valeur de *x*, c’est donc une tautologie.

**Une antilogie (ou contradiction)**

Une antilogie ou contradiction est une combinaison de propositions qui est toujours fausse, quelle que soit la valeur des différentes propositions.

**Exemple** :

*A* : *x* > 7

*B* : *x* < 1

«*A*∧ *B*» est fausse pour toute valeur de *x*, c’est donc une antilogie.

Remarque : on dira que deux propositions *A* et *B* sont logiquement équivalentes ssi la bi-implication *A* ⇔ *B* est une tautologie.

**Un** **système logique**

Un système logique est un [système formel](https://fr.wikipedia.org/wiki/Syst%C3%A8me_formel) dédié au raisonnement et aux déductions logiques. Il est constitué :

* d'un ensemble de formules, y compris un ensemble d'axiomes donnés pour vrais et comme point de départ du raisonnement ;
* d'un ensemble de règles de déduction permettant de définir le type de raisonnement applicables dans ce système ;
* d'une [interprétation](https://fr.wikipedia.org/wiki/Interpr%C3%A9tation_(logique)) des formules, permettant de préciser le sens des formules. Les deux premiers items de cette liste font d'un système logique un système formel, ce troisième item est spécifique aux systèmes logiques.

**Règles d'inférences ou de déduction**

Dans un système [*logique*](https://fr.wikipedia.org/wiki/Logique), les **règles d'**[***inférence***](https://fr.wikipedia.org/wiki/Inf%C3%A9rence) sont les règles qui fondent le processus de [*déduction*](https://fr.wikipedia.org/wiki/D%C3%A9duction_logique), de [*démonstration*](https://fr.wikipedia.org/wiki/D%C3%A9monstration_(logique_et_math%C3%A9matique)). L'application des règles sur les [*axiomes*](https://fr.wikipedia.org/wiki/Axiome) du système permet d'en démontrer les[*théorèmes*](https://fr.wikipedia.org/wiki/Th%C3%A9or%C3%A8me).

  A→B  
  A          
  ∴B

La **déduction logique** est un type de relation que l'on rencontre en [*logique mathématique*](https://fr.wikipedia.org/wiki/Logique_math%C3%A9matique). qui relie des [*propositions*](https://fr.wikipedia.org/wiki/Logique_math%C3%A9matique) dites *prémisses* à une proposition dite *conclusion* et préserve la [*vérité*](https://fr.wikipedia.org/wiki/V%C3%A9rit%C3%A9). Prémisses et conclusion qui sont ainsi reliées par une [*règle de déduction*](https://fr.wikipedia.org/wiki/D%C3%A9duction_naturelle), assurent que si la règle est [*valide*](https://fr.wikipedia.org/wiki/Valide) et si les prémisses sont [vraies](https://fr.wikipedia.org/wiki/V%C3%A9rit%C3%A9), la conclusion est elle aussi vraie. On dit alors que la conclusion est une conséquence des prémisses*. Donc elles fondent la base de la démonstration mathématique*.

## Exemples:

Parmi les règles d'inférence de base de la [*logique propositionnelle*](https://fr.wikipedia.org/wiki/Logique_propositionnelle) est le *modus ponens*, à côté d'elle on trouve le [*modus tollens*](https://fr.wikipedia.org/wiki/Modus_tollens) .

**La règle du**  **modus ponens**,

Le modus ponens, ou détachement, est une figure du [*raisonnement*](https://fr.wikipedia.org/wiki/Raisonnement)[*logique*](https://fr.wikipedia.org/wiki/Logique) concernant *l'*[*implication*](https://fr.wikipedia.org/wiki/Implication_(logique)). Elle consiste à affirmer une [*implication*](https://fr.wikipedia.org/wiki/Implication_(logique)) (« si *A* alors *B* ») et à poser ensuite l'antécédent (« or *A* ») pour en déduire le conséquent (« donc *B* »).

## Formalisation

La règle du *modus ponens* ou de *détachement* est une règle primitive du raisonnement. On l'écrit formellement (suivant le contexte) :

*A* et de *A* ⇒ *B* ⊢*B*

et on peut lire : « de *A* et de *A* ⇒ *B* on déduit *B* », ou encore « *A* et *A* ⇒ *B* donc *B* », c'est-à-dire que l'on affirme *A* et *A* ⇒ *B*, et on en déduit que l'on peut affirmer *B*.

Bien que le connecteur implication (notée généralement « ⇒ » ou « → ») et la relation de déduction (notée « ⊢ ») soient fortement liées, elles ne sont pas de même nature et ne peuvent s'identifier, cette distinction est nécessaire pour formaliser le raisonnement. Ainsi la [tautologie](https://fr.wikipedia.org/wiki/Tautologie) propositionnelle [*A* ∧ (*A* ⇒ *B*)] ⇒ *B* n'est pas une règle, et ne peut représenter le modus ponens, pour le connecteur implicatif désigné par « ⇒ ». Le modus ponens peut en ce sens être vu comme la règle indiquant comment utiliser une implication lors d'une [démonstration](https://fr.wikipedia.org/wiki/D%C3%A9monstration_(logique_et_math%C3%A9matique)).

*A*, {\displaystyle {\frac {P\to Q,\neg Q}{\therefore \neg P}}}*A*→*B*,

\_\_\_\_\_\_\_  
   ∴*B*

**La règle du *modus tollens***

La règle du *modus tollens* Celui-ci est une application de la vérité générale selon laquelle, si une proposition est vraie, alors il en est de même pour sa [proposition contraposée](https://fr.wikipedia.org/wiki/Proposition_contrapos%C3%A9e).

La règle d'inférence *modus tollens* est l'inférence selon laquelle « *P* implique *Q* » et la négation du conséquent *Q* entraînent la négation de l'antécédent *P*.

La règle du *modus tollens* peut être formellement énoncée comme suit :

{\displaystyle {\frac {P\to Q,\neg Q}{\therefore \neg P}}}*A*→*B*, 

\_\_\_\_\_\_\_  
   ∴

**Système complet de connecteurs**

Définition : On dit qu’un ensemble de connecteurs **C** est un système complet de

connecteurs si toute formule propositionnelle est équivalente à une formule n’utilisant

que les connecteurs de **C**.

Proposition :L’ensemble {, ∨} est un système complet de connecteurs.

Preuve. Il suffit de montrer que les propositions (*A*∧*B*) , (*A* ⇒ *B*) , (*A* ⇔ *B*) peuvent

s’écrire en utilisant seulement les connecteurs: «, ∨»

i) On a (*A*∧*B*) ⇔ (∨) .

ii) On a (*A* ⇒ *B*) ⇔∨*B*.

## iii) On a (*A* ⇔ *B*) ⇔ (∨ *B*) ∧ (∨*A*) .

**Calcul des prédicats**

**. Quantificateurs**

**Définition**: Un prédicat est un énoncé dépendant d’une ou plusieurs variables. La

valeur de vérité du prédicat dépend ainsi de la (les) variable(s) le (les) composant(s).

Exemple *p* (*x*) : *x*2- 1 > 0 est un prédicat. On ne peut connaitre sa valeur de

vérité qu’en remplaçant la valeur de *x*.

Ainsi le prédicat est vrai pour *x* = 2 et faux pour *x* = 0.

7.2 Définition Soit *p*(*x*) un prédicat dépendant de la variable *x*. On introduit les

propositions:

1- ∀*x* (*p*(*x*)) : Par définition cette proposition est vraie si toute valeur de *x* rend vrai le

prédicat p(x).

2- ∃*x* (*p*(*x*)) : Par définition cette proposition est vraie s’il existe au moins une valeur de

*x* pour laquelle le prédicat *p*(*x*) est vrai.

Les symboles ∀ et ∃ s’appellent respectivement quantificateur universel et quantificateur

existentiel.

Exemple 1.11 "tous les étudiants possèdent un tél portable" peut se formaliser de la

façon suivante :

∀*x* (*x* est un étudiant ⇒ *x* possède un tél portable).

**7.1 Le quantificateur ∀ :**  « pour tout » La proposition(assertion) *P* peut dépendre d’un paramètre *x*, par exemple « *x* 2 > 1 », La proposition *P*(*x*) est vraie ou fausse selon la valeur de *x*. La proposition ∀*x* ∈ *E* *P*(*x*) est une proposition vraie lorsque les propositions ( assertions) *P*(*x*) sont vraies pour tous les éléments *x* de l’ensemble *E*. On lit « Pour tout *x* appartenant à *E*, *P*(*x*) », sous-entendu « Pour tout *x* appartenant à *E*, *P*(*x*) est vraie ».

**Exemples :**

* « ∀*x* ∈ [0,+∞[ (*x* 2 > 0) » est une proposition vraie.
* « ∀*x* ∈ *IR* (*x* < 5) » est une proposition fausse.
* « ∀*n* ∈ *IN* *n*(*n* + 1) est divisible par 2 » est vraie

**7.2 Le quantificateur ∃** : « il existe » La proposition ∃*x* ∈ *E* *P*(*x*) est une proposition (assertion) vraie lorsque l’on peut trouver au moins un *x* de *E* pour lequel *P*(*x*) est vraie. On lit « il existe *x* appartenant à *E* tel que *P*(*x*) (soit vraie) ».

**Exemples :**

* « ∃*x* ∈ *IR* : (*x* − 1) < 0) » est vraie (par exemple *x* = 0 vérifie bien la propriété).
* « ∃*y* ∈ ***Z*** : *y* – *y*3 < 1 » est vraie (il y a plein de choix, par exemple *y* = 2 convient, mais aussi *y* = 5 il suffit de trouver un seul suffit pour dire que la propositon est vraie).
* « ∃*x* ∈ *IN* (*x*+ 2 = 0) » est fausse (aucun nombre naturel si on l'additionne à 2 donnera un nombre négatif).

**La négation des quantificateurs**

La négation de « ∀*x* ∈ *E* *P*(*x*) » est « ∃*x* ∈ *E* non *P*(*x*) »

Par exemple la négation de « ∀*x* ∈ [1,+∞[ (*x* 2 > 1) » est la proposition

« ∃*x* ∈ [1,+∞[ (*x* 2 < 1) ». En effet la négation de *x* 2 > 1 est non(*x* 2 > 1) mais s’écrit plus simplement

*x* 2 < 1.

La négation de « ∃*x* ∈ *E* *P*(*x*) » est « ∀*x* ∈ *E* non *P*(*x*) »

**Exemples :**

La négation de « ∃*n* ∈ *IN* (*n* - 1 = 0) » est

« ∀*n* ∈ IN (*n* - 1 **** 0) ».

* La négation de « ∀*x* ∈ *IR* (*x* + 1 ∈ ***Z***) » est « ∃*x* ∈ *IR* (*x* + 1 **** ***Z***) ».
* La negation de ∀*x* ∈ *IR* ∃*y* **** (*x* + *y* > 5) est« ∃*x* ∈ *IR* ∀ *y* **** (*x* + *y* ****5) »

**Remarques**

L’ordre des quantificateurs est très important. Par exemple les deux phrases logiques :

∀*x* ∈ *IR* ∃*y* ∈ *IR* (*x* + *y* > 0) et

∃*y* ∈ *IR* ∀*x* ∈ *IR* (*x* + *y* > 0). sont différentes.

La première est vraie, la seconde est fausse. En effet une phrase logique se lit de gauche à droite, ainsi la première phrase affirme « Pour tout réel *x*, il existe un réel *y* (qui peut donc dépendre de *x*) tel que *x* + *y* > 0. » (par exemple on peut prendre *y* = |*x*| + 1). C’est donc une phrase vraie. Par contre la deuxième se lit : « Il existe un réel *y*, tel que pour tout réel *x*, *x* + *y* > 0. » Cette phrase est fausse, cela ne peut pas être le même *y* qui convient pour tous les *x* ! On retrouve la même différence dans les phrases suivantes. Voici une phrase vraie « Pour tout étudiant, il existe un numéro de matriculassions », bien sûr le numéro dépend de la personne. Par contre cette phrase est fausse : « Il existe un numéro de matriculassions, pour tous les étudiants ». Ce serait le même numéro pour tous les étudiants!

**Remarques importantes.**

* Quand on écrit « ∃*x* *∈* *IR* ( *f* (*x*) = 0) » cela signifie juste qu’il existe un réel pour lequel *f* s’annule on peut lire la phrase ainsi : « il existe au moins un réel *x* tel que *f* (*x*) = 0 ». pour préciser que *f* s’annule en une unique valeur, on écrit: ∃! *x* ∈ *IR* (*f* (*x*) = 0).
* Pour la négation d’une phrase logique, il n’est pas nécessaire de savoir si la phrase est fausse ou vraie. Le procédé est algorithmique : on change le « pour tout » en « il existe » et inversement, puis on prend la négation de la proposition *P*.
* Pour la négation d’une proposition, il faut être précis : la négation de l’inégalité stricte « < » est l’inégalité large « **** », et inversement.

Exercices:

1. Écrire la table de vérité du « ou exclusif ». (C’est le ou dans la phrase « fromage ou dessert », l’un ou l’autre mais pas les deux.)

2. Écrire la table de vérité de « non (*P* et *Q*) ». Que remarquez vous ?

3. Écrire la négation de « *P* ⇒ *Q* ».

4. Démontrer les assertions restantes de la proposition ?.

5. Écrire la négation de « *P* et (*Q* ou *R*) ».

6. Écrire à l’aide des quantificateurs la phrase suivante :

« Pour tout nombre réel, son carré est positif ». Puis écrire la négation.

7. Mêmes questions avec les phrases : « Pour chaque réel, je peux trouver un entier relative tel que leur produit soit strictement plus grand que 1 ». Puis

« Pour tout entier *n*, il existe un unique réel *x* tel que **** ».

**2. Raisonnements**

Des méthodes classiques de raisonnements.

2.1. **méthode direct:**

Si on veut montrer que laproposition « *P* ⇒ *Q* » est vraie. On suppose que *P* est vraie et on montre qu’alors *Q* est vraie.

Exemple 1.

Montrer que si *a*, *b* ∈ *Q* alors *a* + *b* ∈ *Q*.

**Démonstration.**

Soient *a* ∈ *Q*, *b* ∈ *Q*. donc ils existent *p* ∈ **Z** et *q* ∈ ****et ****∈ **Z** et ****∈ ****

telques *a* = **** et *b* = **** d'ou

*a* + *b* = ****+****= **** avec le numérateur est bien un élément de **Z**; le dénominateur  est un élément de ****. Donc *a* + *b* s’écrit bien de la forme *a* + *b* = avec ∈ Z, ∈ ****. Ainsi *a* + *b* ∈ *Q*

.**Méthode : Cas par cas**

La méthode du cas par cas la s'apelle aussi méthode de disjunction.

Si l’on souhaite vérifier une proposition  *P*(*x*) pour tous les *x* dans un ensemble *E*, on montre proposition pour les *x* dans une partie *A* de *E*, puis pour les *x* n’appartenant pas à *A*.

. **Exemple 2.**

Montrer que pour tout *x* ∈ *IR*, 

**Démonstration.**

Soit *x* ∈ *IR*. Nous distinguons deux cas. Premier cas :

*x* = 1.Alors on obtient

Deuxième cas *x* > 1. Alors . Calculons alors 

 donc .

Troisième cas: x < 1. Alors . Nous obtenons 

Conclusion. Dans tous les cas 

**2.3. Contraposée**

Le raisonnement par contraposition est basé sur l’équivalence suivante (voir la proposition ?) : La propostion « *P* ⇒ *Q* » est équivalente à « non(*Q*) ⇒ non(*P*) ». Donc si l’on souhaite montrer l’assertion « *P* ⇒ *Q* », on montre en fait que si non(*Q*) est vraie alors non(*P*) est vraie.

Exemple 3. Soit *n* ∈ *IN*. Montrer que si *n* 2 est pair alors *n* est pair.

Démonstration. Nous supposons que *n* n’est pas pair. Nous voulons montrer qu’alors *n* 2 n’est pas pair. Comme *n* n’est pas pair, il est impair et donc il existe *k* ∈ *IN* tel que *n* = 2*k*+1. Alors *n* 2 = (2*k*+1) 2 = 4*k* 2+4*k*+1 = 2*l*+1 avec *l* = 2*k* 2 + 2*k* ∈ *IN*. Et donc *n* 2 est impair. Conclusion : nous avons montré que si *n* est impair alors *n* 2 est impair. Par contraposition ceci est équivalent à : si *n* 2 est pair alors n est pair.

**2.4. Absurde**

Le raisonnement par l’absurde pour montrer « *P* ⇒ *Q* » repose sur le principe suivant : on suppose à la fois que *P* est vraie et que *Q* est fausse et on cherche une contradiction. Ainsi si *P* est vraie alors *Q* doit être vraie et donc « *P* ⇒ *Q* » est vraie

Exemple 4. Soient *a*, *b* > 0. Montrer que si .

**Démonstration.**

Nous raisonnons par l’absurde en supposant que et .

Comme alors *a*(1+ *a*) = *b*(1+ *b*) donc *a* + *a* 2 = *b* + *b* 2 d’où *a* 2 − *b* 2 = *b* − *a*. Cela conduit à (*a* − *b*)(*a* + *b*) = −(*a* − *b*). Comme alors et donc en divisant par *a* − *b* on obtient *a* + *b* = −1. La somme des deux nombres positifs *a* et *b* ne peut être négative. Nous obtenons une contradiction.

Conclusion : si alors a = b.

Dans la pratique, on peut choisir indifféremment entre un raisonnement par contraposition ou par l’absurde. Attention cependant de bien préciser quel type de raisonnement vous choisissez et surtout de ne pas changer en cours de rédaction !

2.5 **Contre-exemple**

Si l’on veut montrer qu’une proposition du type « ∀*x* ∈ *E* *P*(*x*) » est vraie alors pour chaque *x* de *E* il faut montrer que *P*(*x*) est vraie. Par contre pour montrer que cette proposition est fausse alors il suffit de trouver *x* ∈ *E* tel que *P*(*x*) soit fausse. (Rappelez-vous la négation de « ∀*x* ∈ *E* *P*(x) » est « ∃*x* ∈ *E* non *P*(*x*) ».)

Trouver un tel *x* c’est trouver un contre-exemple à la proposition

« ∀*x* ∈ *E* *P*(*x*) ».

Exemple 5. Montrer que la proposition suivante est fausse

« Tout entier positif est somme de trois carrés ». (Les carrés sont les 02 , 12 , 22 , 32 ,... Par exemple 6 = 2 2 + 1 2 + 1 2 .)

**Démonstration.**

Un contre-exemple est 7 : les carrés inférieurs à 7 sont 0, 1, 4 mais avec trois de ces nombres on ne peut faire 7.

**2.6 Récurrence**

Le principe de récurrence permet de montrer qu’une assertion *P*(*n*), dépendant de *n*, est vraie pour tout *n* ∈IN. La démonstration par récurrence se déroule en trois étapes : lors de l’initialisation on prouve *P*(0). Pour l’étape d’hérédité, on suppose *n* > 0 donné avec *P*(*n*) vraie, et on démontre alors que l’assertion *P*(*n* + 1) au rang suivant est vraie. Enfin dans la conclusion, on rappelle que par le principe de récurrence *P*(*n*) est vraie pour tout *n* ∈*IN*.

**Exemple 6**. Montrer que pour tout *n* ∈*IN*, 2*n* > *n*.

**Démonstration**.

Pour *n* > 0, notons P(*n*) l’assertion suivante : 2 *n* > *n*. Nous allons démontrer par récurrence que *P*(*n*) est vraie pour tout *n* > 0.

Initialisation. Pour *n* = 0 nous avons 20 = 1 > 0. Donc *P*(0) est vraie. Hérédité. Fixons *n* > 0. Supposons que *P*(*n*) soit vraie. Nous allons montrer que *P*(*n* + 1) est vraie. 2 *n*+1 = 2*n* + 2*n* > n + 2*n* car par *P*(*n*) nous savons 2*n* > n, et on a 2 *n*+1 > *n* + 1 car 2*n* > 1. Donc *P*(*n* + 1) est vraie.

Conclusion. Par le principe de récurrence *P*(*n*) est vraie pour tout

*n* > 0, c’est-à-dire 2*n* > n pour tout *n* > 0.

**Remarques** :

• La rédaction d’une récurrence est assez rigide. Respectez scrupuleusement la rédaction proposée : donnez un nom à l’assertion que vous souhaitez montrer (ici *P*(*n*)), respectez les trois étapes (même si souvent l’étape d’initialisation est très facile). En particulier méditez et conservez la première ligne de l’hérédité

« Fixons *n* > 0. Supposons que *P*(*n*) soit vraie.

Nous allons montrer que *P*(*n* + 1) est vraie. »

• Si on doit démontrer qu’une propriété est vraie pour tout *n* > *n*0 , alors on commence l’initialisation au rang *n*0 .

• Le principe de récurrence est basé sur la construction de l’ensemble IN. En effet un des axiomes pour définir IN est le suivant : « Soit *A* une partie de IN qui contient 0 et telle que si *n* ∈ *A* alors *n* + 1 ∈ *A*. Alors *A* = IN ».

**Quantificateurs et connecteurs**

Dans des formules faisant une utilisation simultanée de quantificateurs et des connecteurs de conjonction et de disjonction il faut faire attention à la signification des formules.

Ainsi les formules : etsont équivalentes.

Par contre les formuleset ne sont pas équivalentes, la seconde implique la première.

De même etsont équivalentes,

alors que implique 

**Exemple:** on considère les propriétés suivantes dans l'ensemble 

et

La formule affirme que tout élément de  est positif ou négatif.

La formule affirme que tous les nombres de 

sont soit tous positifs soit tous négatifs.

**Clôture d’un prédicat**

**Définition** Une variable qui apparait à la suite d’un quantificateur est dite liée. Une variable qui n’est pas liée est dite libre.

**Exemple** . Ici la variable *x* est liée et la variable  est libre.

**Définition** Une formule qui ne comporte aucune variable libre est dit close. Une

formule close est une proposition.

**Exemple** La formule est une formule close (toutes les variables sont liées). C’est une proposition vraie.

**Définition** 1.16 La clôture universelle (resp. clôture existentielle) d’une formule est la

formule obtenue en adjoignant au début de cette formule les quantificateurs ∀ (resp. ∃) à toutes les variables libres de la formule.

**Exemple** 1.21 Soit le prédicat .

La clôture universelle de ce prédicat est donnée par:

− 1.

La clôture existentielle de ce prédicat est donnée par:

− 1.

**Définition** 1.17 Une formule est dite valide si sa clôture universelle est valide (vraie

dans toutes les situations).

Une formule est dite inconsistante si sa négation est valide.

# Exercices (devoir à rendre le 1er cours)

**Exercice 1**

Trouver des propositions P et Q telles que

1. P⟹Q est vrai et Q⟹P est vrai.
2. P⟹Q est faux et Q⟹P est vrai.
3. P⟹Q est faux et Q⟹P est faux.

**Exercice 2**

Soit A, B et C trois propositions. Démontrer que les propositions A ET (B OU C)   et  (A et B) OU (A ET C) sont équivalentes.

**Exercice 3**

Soit P et Q deux propositions. Montrer que les propositions NON(P⟹Q) et P ET NON Q sont équivalentes.

**Exercice 4**

Écrire sous forme normale conjonctive et sous forme normale disjonctive les propositions ci-dessous :

1-

2-

3-

4-

**Exercice 5**

"S'il pleut, Abel prend un parapluie. Béatrice ne prend jamais de parapluie s'il ne pleut pas et en prend toujours un quand il pleut". Que peut-on déduire de ces affirmations dans les différentes situations ci-dessous? Justifier soigneusement vos réponses en introduisant 3 propositions logiques pp, qq et rr.

**Exercice 6**On rappelle qu'un entier p divise *n*, et on note *p*|*n*, s'il existe un entier relatif *k* tel que *n* = *k*×*p*.

1. Est-ce que 6|*n* est une condition nécessaire à ce que *n* soit pair?
2. Est-ce que 6|*n* est une condition suffisante à ce que *n* soit pair?

**Exercice 7**

Trouver des conditions nécessaires (pas forcément suffisantes) à chacune des propositions suivantes :

1. Avoir son bac.
2. Le point A appartient au segment [BC]
3. Le quadrilatère ABCD est un rectangle.

**Exercice 8**

Trouver des conditions suffisantes (pas forcément nécessaires) à chacune des propositions suivantes :

1. Avoir son bac.
2. Le point A appartient au segment [BC]
3. Le quadrilatère ABCD est un rectangle.

**Exercice 9**

Soit la proposition P : "Le quadrilatère ABCD est un rectangle" et les propositions

1. Q1 : "Les diagonales de ABCD ont même longueur"
2. Q2 : "ABCD est un carré"
3. Q3 : "ABCD est un parallélogramme ayant un angle droit"
4. Q4 : "Les diagonales de ABCD sont médiatrices l'une de l'autre"
5. Q5 : "Les diagonales de ABCD ont même milieu".

Dire si chacune des propositions Q1, Q2, Q3, Q4, Q5 est pour P une condition nécessaire non suffisante, une condition suffisante non nécessaire, une condition nécessaire et suffisante, ou ni l'un ni l'autre.

**Exercice 10**

Parmi toutes les propositions suivantes, regrouper par paquets celles qui sont équivalentes :

1. Tu auras ton examen si tu travailles régulièrement.
2. Pour avoir son examen, il faut travailler régulièrement.
3. Si tu ne travailles pas régulièrement, tu n'auras pas ton examen.
4. Il est nécessaire de travailler régulièrement pour avoir son examen.
5. Pour avoir son examen, il suffit de travailler régulièrement.
6. Ne pas travailler régulièrement entraîne un échec à l'examen.
7. Si tu n'as pas ton examen, c'est que tu n'as pas travaillé régulièrement.
8. Travail régulier implique réussite à l'examen.
9. On ne peut avoir son examen qu'en travaillant régulièrement

**Exercice 11**

Soit*A*, *B* et *C* trois propositions. Si on admet que (*A*⟹*B*)⟹*C*(*A*⟹*B*)⟹*C* est vrai, qui est, avec certitude, nécessaire à qui? Qui est suffisant à qui?

**Exercice 12**

Soit *f*:*IR*→ *IR* une fonction. Donner la négation des propositions suivantes :

1. ∀*x*∈ *IR*, *f* (*x*) ≠0
2. ∀*M*>0, ∃*A* >0, ∀*x* ≥A, *f* (*x*)>*M*
3. ∀*x*∈ *IR*, *f* (*x*) >0⟹x ≤0
4. ∀*ε*>0, ∃*η* >0,∀(*x*,*y*)∈I2, (|*x−y*|≤*η*⟹| *f* (*x*)−*f* (*y*)| ≤ε).

**Exercice 13**

Soit *n* un entier naturel non nul.

Démontrer si les propositions suivantes sont vraies ou fausses.

* 1. ∀*n*∈, ∀*x*∈*IR*, (1+*x*)*n* ≥1+*nx*
  2. ∀*n*∈, ∀*x*∈ *IR* +, (1+*x*)*n* ≥1+*nx*
  3. ∃*n*∈, ∀*x*∈ *IR*, (1+*x*)*n* ≥1+*nx*
  4. ∀*n*∈, ∃*x*∈ *IR*, (1+*x*)*n* ≥1+*nx*
  5. ∃*n*∈, ∀*x*∈, (1+*x*)*n* ≥1+*nx*

**Exercice 14**Soit *f*: *IR* → *IR* une fonction. Exprimer à l'aide de quantificateurs les propositions suivantes :

1. *f* est constante;
2. *f*  n'est pas constante;
3. *f*  s'annule;
4. *f*  est périodique.

**Exercice 15**

Soit  *f* :*IR*→ *IR* une fonction. Énoncer en langage courant les propositions suivantes écrites à l'aide de quantificateurs. Peut-on trouver une fonction qui satisfait cette proposition? Qui ne la satisfait pas?

1. ∀*x*∈ *IR*, ∃*y*∈ *IR*,  *f* (*x*)< *f* (y)
2. ∀*x*∈ *IR*, ∃*T*∈ *IR*,   ( *f* (*x*)=f (*x+T*)
3. ∀*x*∈*IR*, ∃T∈ *IR*∗,  *f* (x)=*f* (x+T)
4. ∃*x*∈ *IR*, ∀*y*∈ *IR*, y=*f* (*x*).