

Séance 1 — Mathématiques pour l'informatique

- Thème : Logique & conditions (Bac+2)
- Objectif du jour : traduire des énoncés en tests booléens corrects, lire des tables de vérité, appliquer 2–3 équivalences clés.
- Plan : accueil & formulaire, connecteurs, tables & équivalences, atelier, démo Python, wrap■up.

Accueil & connexion

- Formulaire d'introduction (5–7 min) — objectifs, préférences d'apprentissage, autoévaluation
- Tour rapide : chacun partage 1 objectif + 1 thème qui motive

Connecteurs logiques

- Propositions : True/False ; variables p , q
- Connecteurs : \neg (non), \wedge (et), \vee (ou), \rightarrow (implique), \leftrightarrow (équivalence)
- Priorité & parenthèses : toujours expliciter en code

Tables de vérité

- Construire la table pour (p, q)
- Classer : tautologie / contradiction / contingente
- Exemple : $(p \rightarrow q) \leftrightarrow (\neg p \vee q)$ — objectif : montrer que c'est une tautologie

Équivalences utiles (à connaître)

- Implication : $p \rightarrow q \equiv \neg p \vee q$
- De Morgan : $\neg(p \wedge q) \equiv \neg p \vee \neg q$; $\neg(p \vee q) \equiv \neg p \wedge \neg q$
- Biconditionnel : $p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$

Traduire des conditions (côté code)

- Valider un mot de passe : ≥ 8 caractères ET au moins un chiffre
- Admission : moyenne ≥ 60 OU (moyenne ≥ 50 ET rattrapage validé)
- Toujours vérifier la logique avec des cas limites (True/False)

Atelier binômes (15–20 min)

- A. Construire la table de $(p \rightarrow q) \leftrightarrow (\neg p \vee q)$ et conclure
- B. Traduire “entrée valide \Rightarrow programme continue, sinon message d’erreur ou arrêt”
- C. Corriger une condition `if` fausse (exemples fournis)

Démo Python (5 min)

- Générateur de tables, vérification d'équivalences
- Idea : tester ``if`/`while`` avec cas de vérité pour éviter les bugs

Ensembles (aperçu)

- Notations : \cup , \cap , \setminus , Δ ; cardinalité
- Application : dédoublonnage avec `set` en Python
- Lien : base pour la prochaine séance (S3)

Ticket de sortie

- Écrire 1 équivalence vue aujourd'hui et expliquer en une phrase pourquoi elle est utile en code
- Question libre pour S2

Devoir / Ressources

- Mini■quiz S2 annoncé
- Notebook S1 (tables, équivalences), Formulaire d'intro
- Bibliographie — MCS (libre) + Rosen (manuel)

Merci & à la prochaine

- Contact enseignant, canaux de questions
- Rappel : supports disponibles en PDF/HTML/Notebook