

Acquis d'apprentissage visés, partie 1

Ce document présente ce dont seront capables les étudiants à la suite des 3 premières séances de cours, 2 premières séances de TP et du travail personnel associé.

Ces acquis seront en particulier tous évalués lors du premier partiel au mois de décembre.

Des exercices types avec grille d'évaluation sont disponibles dans le dossier **Entrainement**.
AAV 1 (Calcul de complexité).

Les étudiants pourront calculer de façon correcte la complexité d'un algorithme itératif simple donné en pseudo-code qu'ils n'auront jamais vu auparavant.

- "simple" signifie que la complexité peut se déduire directement d'une exécution pas à pas de l'algorithme.
- La réponse sera donnée en utilisant de façon correcte et précise la notation O .

AAV 2 (Conception d'algorithme).

Les étudiants pourront concevoir un algorithme de quelques lignes pour répondre à un problème donné qu'ils n'auront jamais vu auparavant selon une complexité fixée. La stratégie de résolution à mettre en place aura été vue lors des séances d'apprentissage (cours ou TP).

- Niveau 1 : application directe des stratégies vue en cours.
- Niveau 2 : la stratégie à appliquer n'est pas évidente, il faut parfois combiner plusieurs stratégies.

Le niveau 1 fait partie des acquis de base et doit être atteint par tous les étudiants en partiel. Le niveau 2 doit être atteint par les étudiants lors des séances de TP tutorées (TP à terminer chez soi sur de longues périodes). En partiel, le niveau 2 est considéré comme non obligatoire : il est nécessaire pour obtenir "20/20" mais pas pour valider le partiel.

AAV 3 (Structures de données et complexité associées).

Sur les trois structures de données suivantes : **tableaux, listes chaînées, ensembles**. Les étudiants seront capables de :

- **Donner la complexité des opérations de base.**
- **Identifier la structure appropriée en fonction des complexités voulues.**
- **Identifier l'opération appropriée (ajout / suppression d'un élément en début / fin) sur une structure donnée pour une complexité donnée.**

AAV 4 (Structures chaînées).

Les étudiants pourront implanter les algorithmes de base (recherche, ajout, suppression) sur une structure de type *liste chaînée*.

AAV 5 (Analyser un algorithme récursif).

Face à un algorithme récursif donné de quelques lignes en pseudo code, les étudiants seront capables de :

- **Calculer le résultat obtenu ou la non terminaison pour des valeurs d'entrée données.**
- **Exprimer à l'aide d'une formule mathématique ou d'une phrase le résultat ou la non terminaison pour une valeur quelconque.**
- **Calculer le nombre d'appels récursifs pour des valeurs d'entrée données.**

- Exprimer le nombre d'appels récursifs par une formule récursive pour une valeur quelconque.
- Calculer la complexité de l'algorithme.

AAV 6 (Concevoir une fonction récursive).

Les étudiants pourront concevoir un algorithme récursif simple pour répondre à un problème donné.

- Niveau 1 : fonction mathématique de base avec une récurrence linéaire simple (exemple : somme des entiers de 1 à n). Ou alors, fonction plus complexe mais la récurrence est donnée aux étudiants.
- Niveau 2 : récurrence plus complexe et non donnée aux étudiants et / ou génération récursive.

Le niveau 1 fait partie des acquis de base et doit être atteint par tous les étudiants en partiel. Le niveau 2 doit être atteint par les étudiants lors des séances de TP tutorées (TP à terminer chez soi sur de longues périodes). En partiel, le niveau 2 est considéré comme non obligatoire : il est nécessaire pour obtenir "20/20" mais pas pour valider le partiel.