

# Cahier des charges -- Allocateur Mémoire

## Description

Ce document détaille le projet AISE que vous aurez à réaliser pour ce semestre 2020. Vous devrez développer un allocateur mémoire. Il devra pouvoir être lancé en remplacement de l'allocateur système pour n'importe quel type d'application ou de bibliothèque. L'évaluation se fera sur deux parties, une première portera sur les composants basiques étudiés en cours, la seconde sur des aspects avancés. **Le projet est à réaliser en binôme, qui nous sera communiqué à l'avance.** La date de rendu est fixée au :

**Vendredi 13 Mars 2018, 23:59:59**

Le barème de cette première partie sera très proche de la répartition suivante et vous permettra d'atteindre facilement la moyenne :

Code	Composant	Description	Barème
A1	<b>Support des fonctions d'allocation</b>	Interception des fonctions depuis une bibliothèque tierce (malloc/realloc/calloc/free)	2
A2	<b>Méthode de compilation</b>	Support d'une application compilée contre l'allocateur (comme une bibliothèque)	1
A3	<b>Mapping mémoire</b>	Allocation de pages via mmap()/munmap()	1
A4	<b>Système de build</b>	Le code se construit simplement avec un Makefile (ou CMake, Autotools...)	1
A5	<b>Recyclage de blocs</b>	Structure de données permettant de réutiliser les blocs précédemment libérés (éviter malloc <-> mmap() )	2
A6	<b>Rapport</b>	Rapport structuré détaillant les choix d'implémentation et mesures de performances) dans un format standard type PDF.	3
—	<b>TOTAL</b>	—	<b>10</b>

Pour aller plus loin, et tenter d'améliorer votre score, voici un certain nombre de fonctionnalités que vous êtes encouragés à explorer. Vous êtes notés sur **24 pour une note sur 20**, il devrait être facile d'avoir une très bonne note ! Voici la liste des fonctionnalités plus complexes (verso de page) :

Code	Composant	Description	Barème
B1	<b>Alignement</b>	Tenir compte des problèmes d'alignement mémoire en fonction de la taille d'un mot	1
B2	<b>Métadonnées</b>	Chaque bloc alloué devrait avoir des méta-données le concernant au lieu d'une gestion trop globale	2
B3	<b>Multithreading</b>	Support efficace de l'allocateur en contexte multithread	2,5
B4	<b>Préchargement de l'allocateur</b>	Fonctionnement de l'allocateur par PRELOAD ( <u>attention</u> : interposition de libc)	3
B5	<b>Politique de sélection</b>	Optimisation et leur justification des fonctions d'allocation (recherche de performances, algorithmes...)	2
B6	<b>Mesure de performances</b>	Comparaison et analyse de votre allocateur à l'existant (allocateur système, jemalloc, TCMalloc...)	2
B7	<b>Profileur mémoire</b>	Mise en place des métriques nécessaires à la création du profil de consommation mémoire d'une application (compteurs...)	1,5
—	<b>TOTAL</b>	—	<b>14</b>

## Rendu

Vous devrez rendre une archive compressée (lisible sous Linux) nommée selon votre binôme, qui devra contenir au minimum :

- Les **sources** du Shell, en langage C ;
- Un **README** au format texte (style « Markdown » conseillé) à la racine du projet, expliquant dans les grandes lignes votre projet et la manière de le compiler ;
- La **liste des fonctionnalités** que vous pensez avoir implémentées, en listant l'ensemble des codes des grilles ci-dessus.

Chaque archive devra nous parvenir avant la date indiquée en première page, de la manière qui vous convient (email, serveur de téléchargement type WeTransfer, Torrent...). Une évaluation des projets sera faite le dernier jour (après le devoir sur table), directement par une démo de votre allocateur. Vous pourrez utiliser la machine que vous souhaitez, tant que celle-ci est sous Linux. Vous pourrez avoir à nous expliquer chaque ligne de votre programme, afin que nous puissions vérifier que vous avez bien compris leur fonctionnement. Chaque archive sera sommée via `sha256sum` à réception pour vérifier qu'il n'y a pas d'altération entre la date de rendu et la démo. Nous utilisons des systèmes de détection du plagiat comme celui fourni par <http://theory.stanford.edu/~aiken/moss/>, il ne sert donc à rien de renommer les variables en copiant le code d'un autre... N'oubliez pas que vous travaillez pour vous et que la connaissance acquise au cours de ce semestre est un élément essentiel pour tout informaticien, surtout s'orientant vers le HPC.