

# Compte rendu TP Allocateur Mémoire

## Introduction

Notre code est très commenté. Pour plus d'information, si référer.

### mem\_init

Nous avons décidé de faire pointer le next de la dernière zone libre sur null et pas sur la première afin de faciliter nos recherches tous au long du tp.

### mem\_show

On a défini deux structures fb. une qui parcourra que les zones libres (next) avec son next et l'autre qui parcourt toutes les zones (all). On reconnaît une zone libre si notre next == all. Si notre next dépasse le all alors c'est une zone occupée.

### mem\_alloc

Si on a une zone libre d'utilisable alors on traite deux cas : -Si les deux zones ont la même taille OU que la zone à allouer et son header est plus grand que la taille disponible ET inférieur à la taille disponible + son header. -Si la taille à allouer plus son header son inférieur à la taille disponible.

Dans le premier cas, on dit que la zone libre devient occupée (ça nous évite de perdre de la mémoire petit à petit de 7 octets max à chaque fois)

### mem\_free

Plusieurs cas ici : -Si une zone libre avant l'occupée puis on regarde si il en a une après. -Sinon si une zone libre après l'occupée. -Pas de zone libre autour.

La différence de ces cas est la fusion de la zone occupée avec les zones libres

### mem\_fit\_first

Pour plus de facilité, on garde toujours le premier header de la première zone libre. Il regarde donc pour la première zone si la taille voulue + son header rentre dans la taille disponible de la zone libre. Sinon, pour les autres zones, il regarde en fonction de la taille de la zone libre et de son header.

### mem\_get\_size

Renvoie juste la taille de la zone nouvellement allouée

### mem\_fit\_best/worst

Code similaire à mem\_fit\_first mais on regarde dans un cas la plus grande zone et la plus petite dans l'autre

## Tests

### **test\_alloc :**

Nous avons décidé de créer qu'un seul fichier de test afin de pouvoir profiter de test progressifs. Dans un premier temps, nous faisons plusieurs allocation de tailles différentes grace à une boucle avec affiche affichage des erreurs si ca marche pas. mais ca marche. Ensuite, on fait une succession d'affichage des zones libre et occupé en même temps que l'on free certaines zones. On ne les libère pas avec un boucle cette fois mais de manière stratégique afin de tester nos divers cas.