

# PROG L3-MI

## DS2

2023-11-14

*Consignes. La durée de ce devoir surveillé est de trente minutes. Aucun document n'est autorisé. Dans tous les exercices, il est possible d'admettre le résultat d'une question pour répondre aux questions suivantes. Toute réponse doit être justifiée.*

### Exercice 1 (environ 4 points) :

La fonction `void *calloc(size_t count, size_t size)` de la bibliothèque C standard a les spécification suivantes : elle alloue sur le tas une zone mémoire contiguë permettant de stocker `count` éléments ayant chacun une taille de `size` octets, et renvoie un pointeur vers cette zone mémoire. De plus, cette zone mémoire est initialisée à zéro (c'est à dire que chaque octet de cette zone mémoire doit valoir `(uint8_t)0`).

La fonction `void *memset(void *b, int c, size_t len)` de la bibliothèque C standard a les spécification suivantes : elle écrit `len` octets de valeur `c` (convertie en un `uint8_t`) dans la zone mémoire pointée par `b`.

#### Q.1 :

1. Écrivez une implémentation de `memset`.
2. Écrivez une implémentation de `calloc` qui utilise `malloc` pour effectuer l'allocation mémoire, et `memset` pour effectuer l'initialisation des données.

## Exercice 2 (environ 3 points)

Q.1 :

1. Écrivez une fonction `swap_bytes` de signature :

```
uint16_t swap_bytes(uint16_t a)
```

qui renvoie la valeur obtenue en échangeant les positions des deux octets de `a`. C'est à dire que si on écrit `a` comme la concaténation HL des deux octets H et L, `swap_bytes(a)` doit renvoyer LH.

2. Donnez en quelques phrases un scénario où cette fonction peut être utile.