

# Déboggage et mise en forme

## Exercice 1 (20%)

Écrivez un programme qui demande l'utilisateur le rayon d'une sphère et écrit sur la console sa surface et son volume.

## Exercice 2 (30%)

Écrivez un programme qui implémente deux fonctions:

- **swap**: qui prend en argument deux pointeurs sur des nombres entiers et échange les valeurs contenues.
- **print\_values**: qui prend en argument deux pointeurs sur des nombres entiers et écrit sur la console un message indiquant la valeur contenu dans le pointeur et la valeur du pointeur.

La console devrait écrire quelque chose dans cette veine:

**The value a is: 4 and is contained in address: 0x7fff5fbff7c8**

**The value b is: 3 and is contained in address: 0x7fff5fbff7c4**

**The value a is: 3 and is contained in address: 0x7fff5fbff7c8**

**The value b is: 4 and is contained in address: 0x7fff5fbff7c4**

## Exercice 3 (50%)

Écrivez le programme *Super Hero VERSUS Horde of Monsters*. Héritant d'une classe

**Character**, créez deux classes **Hero** et **Monster** qui sur-définisse la méthode **death()** en écrivant sur la console:

- Pour Hero: *The hero is dead, long live the hero!*
- Pour Monster: *Yet another dead monster!*

Les **Monster** sont stockés dans une `std::list` et le **Hero** se bat contre eux les uns après les autres. Tant que le **Hero** ou le **Monster** ne sont pas morts, le round continue. Chaque round constitue de deux phases:

- Le **Hero** attaque le **Monster** en utilisant sa méthode **fight**
- Le **Monster** attaque le **Hero** en utilisant sa méthode **fight**

Dès qu'un **Monster** est mort, le suivant vient se battre contre le **Hero**. Dès que le **Hero** meurt, c'est game over et le programme quitte.

La méthode **fight** utilise cette formule:  $damage = attack/defense \cdot strength$

La valeur de **damage** ne peut pas être plus petite que 0, comme elle ne peut pas être plus grande que **strength**.

Chaque **Character** a ces attributs:

- **health**: les points de vie restants
- **strength**: la puissance des dommages

- **attack**: la valeur de comparaison d'attaque
- **defense**: la valeur de comparaison de défense.