

Exercice 1 : C++

Partie 1.1

- main
- classe simple
- compilation / librairie partagée
- LD_LIBRARY_PATH
- exécution

```
> cd exercices/ex1/part1
> make
g++ -Wall -fPIC -c hello.cpp
g++ -shared -Wl,-soname,libhello.so -o libhello.so hello.o
g++ main.cpp -o main -L. -lhello
$ ./main
./main: error while loading shared libraries: libhello.so: cannot open shared object file: No such
file or directory
> LD_LIBRARY_PATH=$PWD ./main
Hello 1
Hello 2
```

Question 1 : Ajouter un getter/setter sur l'attribut privé mNumber de la classe Hello. Tester dans le main.

Partie 1.2

- analyser / corriger
- héritage
- classe abstraite
- const
- valgrind

```
> cd exercices/ex1/part2
> make
g++ -std=c++11 -Wall -fPIC -c hello.cpp helloworld.cpp
g++ -shared -Wl,-soname,libhello.so -o libhello.so hello.o helloworld.o
g++ -std=c++11 main.cpp -o main -L. -lhello
> LD_LIBRARY_PATH=$PWD ./main
....
```

Question 1 : Debugger le code!

Question 2 : Utiliser valgrind pour vérifier qu'il n'y a pas de fuite mémoire. Si c'est le cas, corriger.

```
> LD_LIBRARY_PATH=$PWD valgrind ./main
```

Question 3 : Créer une classe HelloMonde qui affiche "Hello Monde!" lors de l'appel à la méthode *print*. Tester.

Partie 1.3

- surcharge
- méthode statique
- enumerate
- namespace
- switch/case
- valeur par défaut
- inline

```
> cd exercices/ex1/part3  
> make
```

Question 1 : Modifier le main pour utiliser la méthode statique *print* de la classe *Printer*. Essayer les différentes possibilités (enumerate, string, utiliser/ne pas utiliser le mot clé *using*).

Question 2 : Rajouter une valeur par défaut à la méthode *print(Lang lang)*. Tester