

# M1102 - Amphi C++ - Complément 2

Alain Casali    Marc Laporte

Aix Marseille Univ



## 1 Tester ses sous-programmes

- Test interne
- Prérequis
- Prérequis

# Tester ses sous-programmes

La première qualité d'un programme c'est d'être juste (répondre au besoin du client).

Entrée  $\implies$  Programme  $\implies$  Sortie

Entrée : connue (ou à construire)

Programme : maîtrise du code source (ou pas)

Sortie : Est-elle conforme aux attentes ?

Normalement, ce n'est pas le programmeur qui fait les tests, mais le testeur.

Dans la pratique, ...

Exemple de méthodes de tests :

- Test boîte noire ;
- TDD (Test Driven Développement) ;
- Test rouge / vert.

En général, on écrit une fonction de test par fonction développée.

# Test interne

Quelque part dans le code :

```
[Type RetVal =] XXX ([const] Type1 & VarIdent1,  
                    [const] Type2 & VarIdent2, ...);  
TestXXX (const Type1 & VarIdent1, const Type2 & VarIdent2, ...,  
        const Type & RetVal);
```



La fonction de test :

- doit inclure un éventuel paramètre de retour (à la fin);
- doit avoir exactement les mêmes paramètres que la fonction à tester;
- on ne doit modifier aucune valeur ( $\Rightarrow$  `const` de partout);

# Exemple

## Exemple : Fonction à tester

```
void Init (vector<unsigned> & V, const unsigned & Size)
{
    V.resize (Size);
    for (unsigned i (0); i < Size; i = i + 1) V[i] = i + 1;
}
```

## Exemple : Fonction de test

```
void TestInit (const vector<unsigned> & V,
              const unsigned & Size)
{
    if (V.size () != Size)
    {
        cerr << "I don't know what to tell" << endl;
    }
    if (V[0] != 1)
    {
        cerr << "I don't know what to tell" << endl;
    }
}
```

# Utilisation d'assertions

## Exemple : Fonction de test

```
void TestInit (const vector<unsigned> & V,  
              const unsigned & Size)  
{  
    assert (V.size () != Size);  
    assert (V[0] != 1);  
}
```



- si une assertion est fausse, le programme s'arrête ;
- seules les erreurs sont affichées sur la console.

## Prérequis

```
#include <cassert>
```

# Test rouge / vert

## Exemple : Fonction de test

```
void TestInit (const vector<unsigned> & V,  
              const unsigned & Size)  
{  
    EASY_ASSERT (V.size () != Size);  
    EASY_ASSERT (V[0] != 1);  
}
```

- si une assertion est fausse, elle s'affiche en rouge dans la console ;
- si une assertion est juste, elle s'affiche en vert dans la console ;

## Prérequis

```
#include "EasyAssert.h"
```



Ne pas oublier d'inclure le fichier `EasyAssert.h` dans le projet !