

# General Coding Techniques

## 1 Template

- The use of `<>` in C++

- 声明类模版

```
template <typename T>
MyClass{
    ...
}
```

- 使用: `MyClass<int>;`

- 可以在模版初始化中传递参数

```
template <typename T, int m, int n> MyClass{
    enum {rows = m, cols = n};
}
```

- 使用: `Myclass<int, 1, 3>;`

## 2 Input and Output variable

### 1. 输入输出对象

- `std::cerr` 的使用方法
  - `std::cerr` 标准错误输出流; `std::cout` 标准输出流; `std::cerr` 与 `std::cout` 的最大不同是 `cerr` 是不带输出缓冲的, 直接就可以输出到显示器上, 而 `cout` 是带输出缓冲的, 需要刷新缓冲区才能输出。

- `std::cout << std::setw(5) << "n" << std::setw(15) << "kron" << std::endl;`
  - `setw(10)` 指明其后的输出项占用的字符宽度为10, 即括号中的值紧跟其后的输出项占用的字符位置个数, 并且给定宽度大于实际数据位数时默认向右对齐, 左侧填充空格, 填满给定的宽度。

### 2. assert

- (a)

```
// Use assert() to make sure A is a square matrix of
assert(A.cols() == A.rows() && "Matrix mush be square.");
```

## 3 数据类型、运算符和表达式