

## C++作业：按块组织数据

要求：

- 1、实现大量数据存储，数据存储按块(Block)进行；初始化时，可以设定块的长度，默认长度为 100 个元素。当执行 `push_back()`时，当长度达到设定空间时，自动按设定长度增加块，新数据放在新块之中，原有块的数据不拷贝、不删除；为管理相应的块，可增加 `blockTable`，用于保存每块信息；
- 2、函数 `push_front()`用于在所有数据之前增加元素，须考虑本题目设计精神，即容量可以前空白，也可以尾空白；用于高效前增和后增(此处指增加记录)；
- 3、当删除某个元素时，通过查询 `blockTable`，找到相应块及其在块中位置，通过块内移动方式进行数据位置调整。删除时余下空间，任然在块中，但要确保已存储数据连续存储。当一个块的数据全部删除时，删除该块；删除函数包括：`pop_back()`【后删除】、`pop_front()`【前删除】、`delete(N)`【指定位置删除】；
- 4、当插入数据时，如果块已满，则新增块，块的长度为原长度的百分之 110%，相当于每次增长 10%，其中 10%可通过函数设置。  
注意：`blockList` 的设计，需要保留块的地址、容量、已用数量等；
- 5、函数 `compress()`函数用于压缩，将所有块的长度与容量保持一致；
- 6、需重载实现[]运算符，能通过[]运算符取值或赋值；
- 7、需实现所有类型构造函数和析构函数；
- 8、需实现赋值运算符重载；
- 9、实现迭代器，能使用课堂 `countIF()`和 `forEach()`；
- 10、将迭代器升级到随机迭代器；