push\_front：

Description

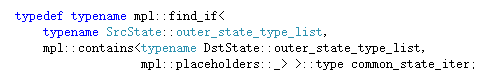
push\_front performs an insertion at the beginning of the sequence. The algorithm returns a new sequence which contains type T as its first element. The result sequence preserves all the functional and performance characteristics of the original Sequence, except its size and identity.

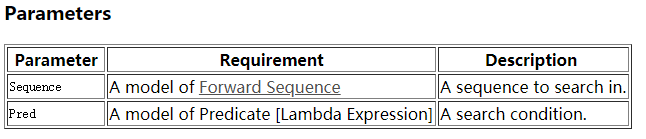
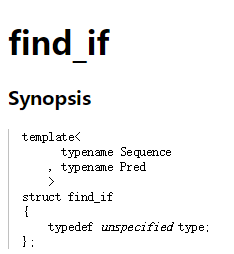
typedef push\_front<Sequence,T>::type s;

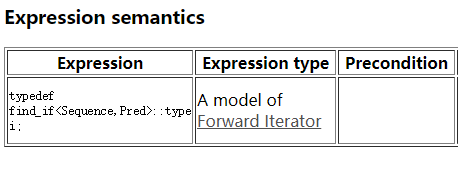
说明：往队列Sequence首个元素前面插入T元素，生成队列s。（这里的Sequence类型应该是 mpl::list<>。）

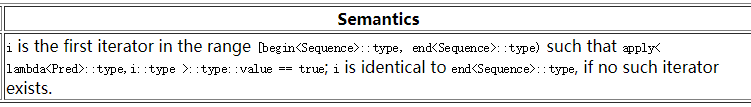
find\_if

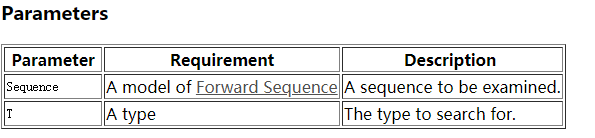
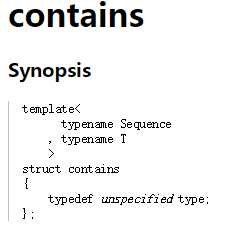
说明：find\_if前面应该放入源状态的外部状态列表，后面放入的是lambda表达式，这个lambda表达式由contains来组成。contains前一个参数放入目标状态的外部状态列表，后面一个参数放入未命名占位符。如果源的某个外部状态放入lambda表达式，返回类型是true，说明匹配，然后就返回它的迭代器。

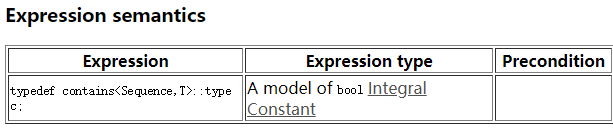


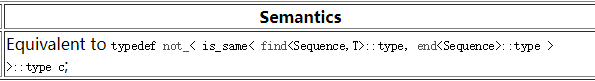


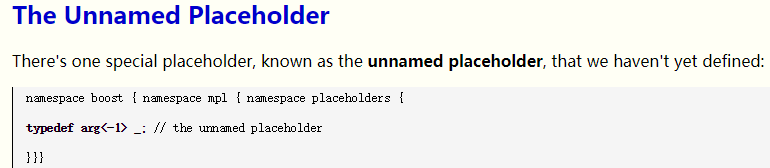






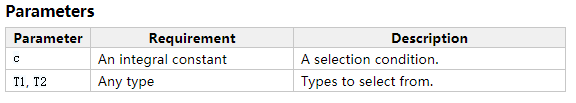
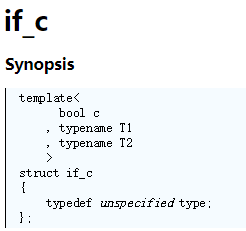






The details of its implementation aren't important; all you really need to know about the unnamed placeholder is that it gets special treatment. When a lambda expression is being transformed into a metafunction class by mpl::lambda,

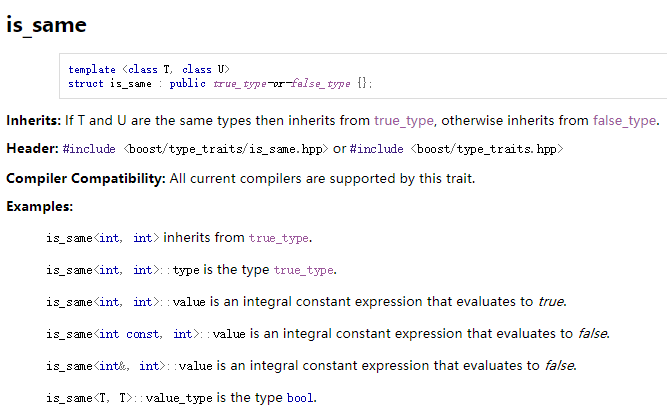
the nth appearance of the unnamed placeholder in a given template specialization is replaced with \_n.



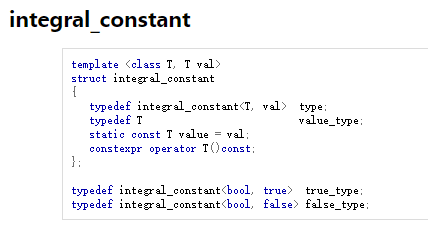
Description

Returns one of its two arguments, T1 or T2, depending on the value of integral constant c. if\_c<c,t1,t2>::type is a shorcut notation for if\_< bool\_<c>,t1,t2 >::type.

<https://www.boost.org/doc/libs/1_59_0/libs/mpl/doc/refmanual/if-c.html>



如果传入类型相同则继承true\_type，否则继承false\_type。（这两种结构体都继承自integral\_constant）



**Deref**

