<span class="hljs-function"><span class="hljs-keyword">decltype</span>(<span class="hljs-keyword">auto</span>) <span class="hljs-title">look\_up\_a\_string\_1</span><span class="hljs-params">()</span> </span>{ <span class="hljs-keyword">return</span> lookup1(); }

<span class="hljs-function"><span class="hljs-keyword">void</span> <span class="hljs-title">look\_up\_a\_string\_2</span><span class="hljs-params">()</span> </span>{ <span class="hljs-keyword">return</span> lookup2(); }

<span class="hljs-function"><span class="hljs-keyword">friend</span> <span class="hljs-keyword">void</span> <span class="hljs-title">A::showB</span><span class="hljs-params">(B x)</span> </span>{}

<span class="hljs-function"><span class="hljs-keyword">friend</span> <span class="hljs-keyword">void</span> <span class="hljs-title">showB</span><span class="hljs-params">(B x)</span> </span>{}

<span class="hljs-function"><span class="hljs-keyword">friend</span> <span class="hljs-keyword">void</span> <span class="hljs-title">showB</span><span class="hljs-params">(B::SomeType x)</span> </span>{}

<span class="hljs-function"><span class="hljs-keyword">inline</span> <span class="hljs-keyword">int</span> <span class="hljs-title">add</span><span class="hljs-params">(<span class="hljs-keyword">int</span> a, <span class="hljs-keyword">int</span> b)</span> </span>{}

<span class="hljs-function">int8t <span class="hljs-title">Get\_Tile\_Value</span><span class="hljs-params">()</span> </span>{}

<span class="hljs-function"><span class="hljs-keyword">int8\_t</span> <span class="hljs-title">Get\_Tile\_Value</span><span class="hljs-params">()</span> </span>{}

<span class="hljs-function">B::type <span class="hljs-title">test</span><span class="hljs-params">()</span> </span>{};

<span class="hljs-comment">// template</span>

<span class="hljs-function">boost::optional&lt;application&gt; <span class="hljs-title">handle\_key</span><span class="hljs-params">(application state, key\_code key, coord size)</span></span>;

test();