

```

1  #include <cstdlib>
2
3  using namespace std;
4
5  #ifndef SLNODE_H
6  #define SLNODE_H
7
8  template<typename T>
9  class SLNode
10 {
11     public:
12
13     /***** constructor/destructor definitions *****/
14
15     SLNode()
16     : nextNode( NULL ), contents( NULL )
17     {
18         /* empty */
19     }
20
21     SLNode( T contents )
22     : contents( contents ), nextNode( NULL )
23     {
24         /* empty */
25     }
26
27     ~SLNode()
28     {
29         nextNode = NULL;
30     }
31
32     /***** accessor/mutator function definitions *****/
33
34     void setContents( T &contents )
35     {
36         this->contents = contents;
37     }
38
39     T getContents() const
40     {
41         return contents;
42     }
43
44     void setNextNode( SLNode<T>* nextNode )
45     {
46         this->nextNode = nextNode;
47     }
48
49     SLNode<T>* getNextNode() const
50     {

```

```
51         return this->nextNode;
52     }
53
54     SLNode<T>*& getNextNode()
55     {
56         return this->nextNode;
57     }
58
59     private:
60
61     /***** private variable declarations *****/
62
63     SLNode<T>* nextNode;
64     T contents;
65 };
66
67 #endif
68
```