

```

                                node2. template
// FILE: node2. template
// IMPLEMENTS: The functions of the node template class and the
// linked list toolkit (see node2.h for documentation).
//
// NOTE:
//   Since node is a template class, this file is included in node2.h.
//   Therefore, we should not put any using directives in this file.
//
// INVARIANT for the node class:
//   The data of a node is stored in data_field, and the link in link_field.

#include <cassert>    // Provides assert
#include <cstdlib>    // Provides NULL and size_t

namespace main_savitch_6B
{
    template <class Item>
    std::size_t list_length(const node<Item>* head_ptr)
    // Library facilities used: cstdlib
    {
        const node<Item> *cursor;
        std::size_t answer;

        answer = 0;
        for (cursor = head_ptr; cursor != NULL; cursor = cursor->link( ))
            ++answer;

        return answer;
    }

    template <class Item>
    void list_head_insert(node<Item>*& head_ptr, const Item& entry)
    {
        head_ptr = new node<Item>(entry, head_ptr);
    }

    ...

    template <class NodePtr, class Item>
    NodePtr list_search(NodePtr head_ptr, const Item& target)
    // Library facilities used: cstdlib
    {
        NodePtr cursor;

        for (cursor = head_ptr; cursor != NULL; cursor = cursor->link( ))
            if (target == cursor->data( ))
                return cursor;
        return NULL;
    }

    ...
}

```