Quiz 2 CSCI 333 Fall 2001
If at first you don't succeed
Try, try again

Name: _____

You have your grade on the original Quiz 2. You, *working alone*, can complete this Quiz 2 as a take-home exam and turn it in by Wednesday, 15 November, 2001.

Your recorded score on Quiz 2 will be the following
ROUND(MAX(2**OriginalQuiz*2+100, 2**TakehomeQuiz2*+*OriginalQuiz2*)/3)

Problem 1 (15 points)

Show the comparisons and exchanges performed in a single quicksort partition when 30 is chosen as the pivot with the following array values:

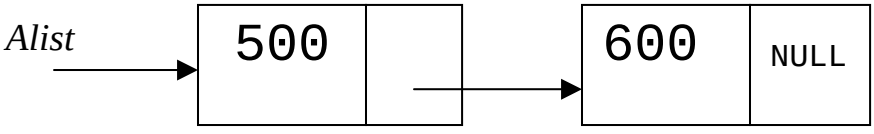
27, 35, 37, 21, 25, 39, 31, 23, 33, 29

Problem 2 (20 points)

This two-part problem uses the singly linked list node implementation shown below and taken from Figure 4.5 (page 96) of the textbook:

```
// Singly-linked list node
template <class Elem> class Link {
public:
  Elem element;      // Value for this node
  Link *next;        // Pointer to next node in list
  Link(const Elem& elemval, Link* nextval =NULL)
    { element = elemval;  next = nextval; }
  Link(Link* nextval =NULL) { next = nextval; }
};
```

Assuming that `Elem`, the type of the value field is `int`; write the C++ code needed to create the following two-element list and assign it to a pointer variable of type `Link<int>` named `Alist`.



Now write a C++ function `MaxOfTwo` to find the larger of the first two elements of a `Link<int>` list. If the list contains no elements, return -1. If it contains only one element, return the value of that one element.

Problem 3 (20 points)

Repeat both parts of problem 2, but this time use `LList<int>`, the linked list class defined in Figure 4.7 (page 97) of the textbook. Leave the "fence" at the beginning of the list at the end of both operations.

Problem 4 (15 points)

Build a binary search tree by starting with an empty tree and then adding, in order, the nine values: 400, 700, 500, 200, 300, 600, 900, 100, 800.

Problem 5 (15 points)

How many comparisons would insertion sort *and* selection sort make when sorting the following list of fifteen elements (where only three elements are out-of-order)?

1, 2, 3, 4, 5, **8**, **7**, **6**, 9, 10, 11, 12, 13, 14, 15

Try to give an exact answer. Show your work if you wish to have a chance of receiving partial credit. You should be able to do this without following every step of the two algorithms.

Problem 6 (15 points)

Show the comparisons and exchanges made by heapsort in its initial phase of building the max-heap when it is sorting the nine elements 400, 700, 500, 200, 300, 600, 900, 100, and 800. For full credit, you must also show how these elements are stored in an array.