CS 216 Exam 1 – Fall 2004

Name:_______________________  Lab Section:_______________
Email Address:_______________ Student ID # ______________

This exam is closed note, closed book. You will have an hour and fifty minutes total to complete the exam. You may NOT use calculators.

Good Luck!!

<table>
<thead>
<tr>
<th></th>
<th>MAX</th>
<th>SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL</td>
<td></td>
<td>104</td>
</tr>
</tbody>
</table>

Write and sign pledge after taking the exam:
Define each term and give an example that explains it:

1. (2 points) excess B

2. (2 points) height of an AVL tree

3. (2 points) FIFO data structure

4. (2 points) abstract data type
5. (2 points) Order the following rates of growth from slowest rate of growth to fastest rate of growth: \(n^8, \ n \log n, \ n, \ 2^n, \ 300, \ \log n\)

6. (3 points) Name 3 factors that are ignored by big-Oh notation.

7. (4 points) Fill in the blanks in the definition of big-Oh notation:

\[ T(N) = O(f(N)) \text{ if:} \]

there are positive constants \(c\) and \(n_0\) such that:

\[ \text{______________________________} \]

when:

\[ \text{______________________________} \]
8. (10 points total) Describe the running time of the following pseudocode in Big-Oh notation in terms of the variable \( n \). Assume all variables used have been declared. 

\textit{Show your work for partial credit.}

\begin{verbatim}
int vote_count(int k) {
    int sum;
    for (int i = 0; i < 500; ++i)
        sum = sum + (i * k);
    return sum;
}

a) answ = vote_count(n);

b) int sum;
   if (n < 100)
       cout << "whoa!";
   else
       for (int i = 0; i < n; ++i)
           sum += vote_count(n);

c) for (int j = 4; j < n; ++j) {
    cin >> val;
    for (int i = 0; i < j; ++i) {
        b = b * val;
        for (int k = 0; k < n; ++k)
            c = b + c;
    }
}

d) for (int i = 0; i < n * n; ++i) {
    sum = sum/n;
    for (int j = 0; j < i; ++j)
        j >> cout;
}

e) for (int i = 0; i < n; ++i) {
    for (int j = 0; j < i * n; ++j)
        sum = sum + i;
    for (int k = 0; k < n + n; ++k)
        a[k] = a[k] + sum;
}
\end{verbatim}
9. (6 points total) What is the representation of each of the following in the indicated radix? Be sure to show your work.

a) $127_{10}$ in decimal

b) $1132_8$ in hex

c) $2E_{15}$ in radix 10

10. (6 points total) Consider the positive binary integer represented in two’s complement: $0110010110000111_2$.

a) Express this binary number in octal

b. Express this binary number in hexadecimal

c. Negate the number (i.e. give the two’s complement representation of a negative version of the same number) Use the same number of bits.
11. (3 points) Draw the binary search tree created by inserting these values in this order:

        4  1  9  6  7  2  5  0  8

12. (2 points) Give a pre-order traversal of your tree shown above:

13. (2 points) Give a post-order traversal of your tree shown above:

14. (3 points) Delete the root of the tree shown above using one of the methods described in class. Draw the new tree here:
15. For each operation below give: 1) How you would most efficiently implement the operation, 2) Describe the worst case scenario (e.g. “The worst case occurs when the value you are looking for is not in the list”) and 3) What is the worst case Big-Oh running time of this scenario. **State any assumptions you make.**

a) (3 points) Find the maximum value stored in an AVL tree.

b) (3 points) Pop a value from a stack implemented as an array.

c) (3 points) Find the total number of values stored in a queue implemented as a doubly linked list.
16. (3 points) Given the following tree:

Are it an AVL tree? If not, circle the node(s) where the AVL property is violated. Why or why not (must answer for any credit)?

17. (3 points) Given the following tree:

Are it an AVL tree? If not, circle the node(s) where the AVL property is violated. Why or why not (must answer for any credit)?
18. (7 points) Assume we are using the 32-bit IEEE single precision floating point format as described in class and used in lab. The mantissa has 24 bits including the hidden bit. There is one sign bit and there are eight exponent bits. The exponent is stored in excess 127.

What decimal floating point number is represented by the following 32 bits? SHOW YOUR WORK!

1000 1010 1111 0000 0000 0000 0000 0000

a) Is this a positive or negative number?

b) What is the exponent (in base 10)?

c) What is the value of the mantissa (in base 10)

d) What is the total value?

Note: you may leave your answer in the form: \( value_{10} \times base^{exponent} \)

Where you specify value, base and exponent.
19. (20 points) This question tests your understanding of stacks and pointer manipulation. You must implement a stack ADT in C++. The underlying representation of the stack should be the Node class as described below used in a singly linked list. Your stack should store integers and should handle errors (printing an error message is fine).

You will be graded mostly on the correctness of the ideas of your solution rather than exact C++ syntax, but your solution should be clear. Correct C++ code is the best way to ensure we understand your solution. You may NOT use the STL in any way for this question. You should use the header file provided below. You do not need to implement copy constructors, destructors, or operator= for these classes, but otherwise the routines you implement should handle memory management appropriately.

You should implement all the functions with \( \rightarrow \) in front of them.

class Node {
  public:
    Node(int value): val(value), next(NULL) {};
    int val;
  Node *next;
};

class Stack {
  public:
    \( \rightarrow \) Stack();  // constructor
    \( \rightarrow \) void push(int value);  // pushes value onto the stack.
    \( \rightarrow \) int pop();    // returns and removes the value on // the top of the stack.
    \( \rightarrow \) int top();    // returns the value on the top of // the stack without removing it.
    \( \rightarrow \) bool isEmpty(); // returns true if the stack contains // no elements.
  private:
    // Add data members here.
};
20. (8 points) Implement a non-member function:

\[
\text{void print_stack_inorder}(\text{Stack my_stack})
\]

that takes a Stack as input, and prints out all the values in my_stack in ascending numerical order, e.g. from smallest value to largest value. To answer this question the only data structure you may use is a Stack (or temporary stacks). You may not use lists, arrays, vectors, queues. For the purposes of this question you can assume that a copy constructor that implements a deep copy exists for the Stack class. Assume you have a working version of the stack from the previous question, and a definition of the interface.
21. (2 points) What is the worst case big-Oh running time of your \texttt{pop} method and why?

22. (3 points) What is the worst case big-Oh running time of your \texttt{print_stack_inorder} function and why?