

Name:

Score _____ / 100

1. (10 points) The following code is part of a linked list class and is, technically, a perfectly legal use of overloaded operators as far as the compiler is concerned. Why shouldn't it be used in practice?

```
void LinkedList::operator[] (int num) {  
    // remove the first num entries from the list  
    for(int count = 0; count < num && head != 0; count++) {  
        Node * curr = head;  
        head = head->next;  
        delete curr;  
    }  
}
```

2. (15 points) A developer is creating a C++ library for reading and writing XML files. There are a number of errors that can crop up in the course of reading an XML file (files that fail to open, improperly formatted files, etc) and the library needs to be able to deal with those errors. Assuming programs using the library should be able to recover from errors in the library if they wish, should the developer use thrown exceptions or assertions for error handling in the library? Why?
3. (10 points) A C++ program under development relies heavily on dynamically allocated memory. During testing, it is revealed that the program is suffering from frequent segmentation faults. Trace debugging does not reveal anything useful about why the seg faults are occurring. What should be done next to debug the program? (No, “ask the developer's C++ instructor” is not an acceptable answer. Sorry.)

4. (15 points) What output would be written to “output.txt” by the following code segment, assuming the file “input.txt” contains “3 foo bar baz”? (For convenience, the code below uses a “_” to indicate spaces in strings. Do the same in your output.)

```
#include <string>
#include <fstream>
#include <iomanip>

using namespace std;

int main() {
    int count;
    ifstream in ("input.txt");
    ofstream out ("output.txt");
    in >> count;
    string str[count];
    for(int i = 0; i < count; i++) {
        in >> str[i];
    }
    out << "|";
    for (int i = 0; i < count; i++) {
        out << setw(str[i].length()) << i << "_|";
    }
    out << endl;
    out << "|_";
    for (int i = 0; i < count; i++) {
        out << setw(str[i].length()) << str[i] << "_|_";
    }
    out << endl;
    return (0);
}
```

5. (15 points) Describe what the following function does, being sure to mention details related to the types of the arguments. Also identify any special requirements placed on the types of the arguments.

```
template <typename T, typename U>  
T GetMin (T a, U b) {  
    return (a<b?a:b);  
}
```

6. (10 points) What general purpose is served by an STL iterator?

7. (5 *bonus* points) What is the air speed velocity of an unladen swallow?

8. (15 points) Below is a simple C++ program. Describe what would need to be done to convert it to a C program. You DO NOT need to actually write the C version. (Hint: there are three major types of changes to be made.)

```
#include <iostream>
using namespace std;

int main() {
    cout << "how many values? ";
    int num;
    cin >> num;
    int * arr = new int[num];
    for(i = 0; i < num; i++) {
        cin >> arr[i];
    }
    delete arr;
}
```

9. (10 points) Briefly describe the differences between statically and dynamically linked libraries.