UMBC

Student's Name _____

Date _____ Subject _____

Professor's name: _____

Comments: _____

_____

_____

_____

Integrity:
A Value That Endures

"By enrolling in this course, each student assumes the responsibilities of an active participant in UMBC's scholarly community in which everyone's academic work and behavior are held to the highest standards of honesty. Cheating, fabrication, plagiarism, and helping others to commit these acts are all forms of academic dishonesty, and they are wrong. Academic misconduct could result in disciplinary action that may include, but is not limited to, suspension or dismissal. To read the full Student Academic Conduct Policy, consult the UMBC Student Handbook, the Faculty Handbook, or the UMBC Policies section of the UMBC Directory."

UMBC Faculty Senate
February 13, 2001

1. (18 points) There are six logic or syntax errors in the following program. Circle each error and write the line number and correction in the space provided below.

```
1      #include <iostream.h>
2      using namespace std;
3      int main() {
4        int n = -1;
5        int fact;
6        cout << "This program computes n factorial."
               << endl;
7        do {
8           cout << "Enter a positive integer n: ";
9           cin << n;
10       } while (n < 0);
11       if (n = 0)
12          cout << "Factorial of 0 is 1" << endl;
13       else
14          for (int i = 1; i < n; ++i)
15             fact *= i;
16       cout << "Factorial of " n << " is " << fact
               << endl;
17       return 0;
18    }
```

Line Number	Correction

2. (8 points) Complete the code:

   a. I want to compute the *average* of the two integer variables *x* and *y* and save it to the double variable *avg*.

```
avg = ( x + y ) / [          ] ;
```

   b. A race of aliens from a planet with a six hour day wants to convert 11 am local earth time to the time on their home planet:

```
int umbcTime = 11;

int alientTime;

alientTime = umbcTime [          ] 6;
```

   c. The program should only call the function *ReturnGrades()* if the variable *numStudents* has a value between 1 and 500, inclusive:

```
if ( [                                              ] )
        ReturnGrades(grades, numStudents);
```

   d. The user of a data analysis program can enter 's' to save their data or 'h' to display a help message. The users selection is stored in the variable *selection*:

```
switch( selection ) {
  case 's':
    SaveData();
    break;
  case 'h':
    DisplayHelp();
    [          ] ;
  default:
    cerr << "Invalid selection" << endl;
}
```

3. (8 points) Explain why the following program will not do what the programmer intended:

```
1      #include <iostream>
2      using namespace std;
3      void AbsValue(double x);
4      int main() {
5         double x = -7.251;
6         // Replace x with its absolute value
7         AbsValue(x);
8      }
9      void AbsValue(double x) {
10        if ( x < 0.0 )
11           x = -x;
12     }
```

4. (4 points) What is the value of *x* in the following code sample? Circle the correct answer.

```
1      int a = 2, b = 3, c = 5, d = 8;
2      int x = a + b * d / c;
```

a. 5   c. 8

b. 6   d. 10

5. (8 points) What will the following program print to the screen? Complete the boxes below.

```
1      #include <iostream>
2      using namespace std;
3      int main() {
4        int i = 3, j = 4;
5        {
6          int j = 5;
7          cout << i << " " << j << endl;
8          i += j;
9        }
10       cout << i << " " << j << endl;
11     }
```

Output:

6. (6 points) List the names of the *arguments* and the *parameters* in the following example code:

```
1      #include <iostream>
2      using namespace std;
4      int Sum(int x, int y, int z);
5      int main() {
6        int a = 1, b = 2, c
7        cout << Sum(a, b, c) << endl;
8      }
9      int Sum(int x, int y, int z) {
10         return x + y + z;
11     }
```

Arguments:

Parameters:

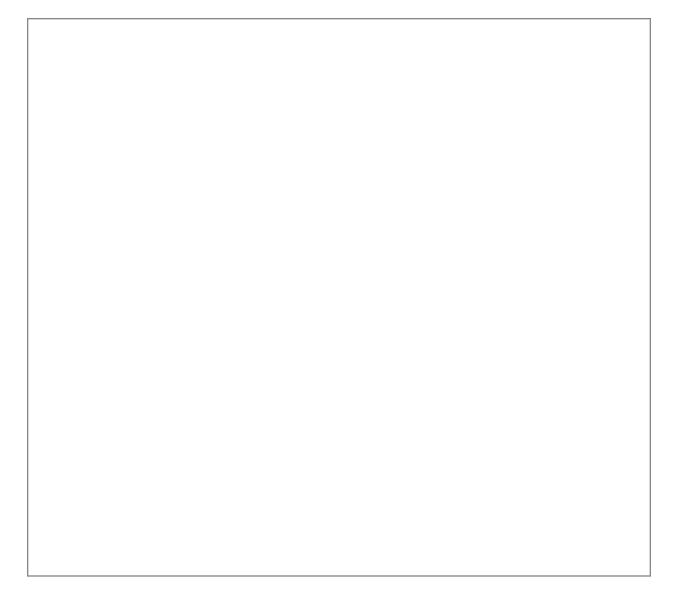| |
|---|

7. The *trace* of a matrix is the sum of its diagonal entries. I want to write overloaded functions to compute the trace of a double or integer matrix, stored as a two-dimensional array. The trace of an integer array is an integer, and the trace of a double array is a double. The array will be declared to have MAX_SIZE rows and columns, but the actual size will be passed as an integer argument to the function.

    a. (8 points) Complete the function prototypes for the two functions:

| | Trace(| , |); |

| | Trace(| , |); |

    b. (12 points) Write the double version of the *Trace()* function:

8. (12 points) Write a function header comment for your *Trace()* function from (7.b), including a description of the function, its pre-conditions, and post-conditions.

9. (8 points) We've learned about three different loop statements. In each of the following situations, which is the *most* appropriate?

    a. A user will be prompted to enter an integer in the range one to ten, inclusive; the prompt will be repeated until the user enters a value in the correct range.

    b. Sum the values in a fixed-length double array.

    c. If there is a data file in a particular directory, read the file and process the data; repeat so long as there are still data files in the directory.

    d. At the end of the semester, compute the lab average for a particular student.

10. (8 points) What output is produced by the following code?

```
1     int *p1, *p2;
2     int x = 3, y = 5;
3     int z[3] = {1, 2, 3};
4     p1 = &x;
5     p2 = &y;
6     cout << *p1 * *p2 << endl;
7     p2 = p1;
8     cout << *p1 + *p2 << endl;
9     p1 = z;
10    cout << *(p1+1) * *p2 << endl;
11    cout << *(p1+2) / *p2 << endl;
```

```



```

```



```

```



```

```



```

Page	Points	Earned
1	18	
2	8	
3	12	
4	14	
5	20	
6	20	
7	8	
Total	**100**	