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.

```cpp
#include <iostream.h>
using namespace std;

int main() {
    int n = -1;
    int fact = 1;
    cout << "This program computes n factorial."
         << endl;
    do {
        cout << "Enter a positive integer n: ";
        cin >> n;
    } while (n < 0);
    if (n == 0)
        cout << "Factorial of 0 is 1" << endl;
    else
        for (int i = 1; i <= n; i++)
            fact *= i;
        cout << "Factorial of " << n << " is " << fact << endl;
    return 0;
}
```

+1 pt each correct line number, +2 points each correct fix

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Correction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#include &lt;iostream&gt; (no .h)</td>
</tr>
<tr>
<td>5</td>
<td>int fact = 1;</td>
</tr>
<tr>
<td>9</td>
<td>cin &gt;&gt; n;</td>
</tr>
<tr>
<td>11</td>
<td>if (n == 0)</td>
</tr>
<tr>
<td>14</td>
<td>for (int i = 1; i &lt;= n; i++)</td>
</tr>
<tr>
<td>16</td>
<td>cout &lt;&lt; &quot;Factorial of &quot; &lt;&lt; n &lt;&lt; &quot; is &quot; &lt;&lt; fact &lt;&lt; endl;</td>
</tr>
</tbody>
</table>
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 $avg$.

$$avg = \frac{x + y}{2.0}; \quad +2 \text{ points}$$

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:

```cpp
int umbcTime = 11;
int alienTime;
alienTime = umbcTime % 6; \quad +2 \text{ points}
```

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

```cpp
if (numStudents >= 1 && numStudents <= 500) \quad +2 \text{ points}
    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`:

```cpp
switch( selection ) {
    case 's':
        SaveData();
        break;
    case 'h':
        DisplayHelp();
        break; \quad +2 \text{ points}
    default:
        cerr << "Invalid selection" << endl;
}
```
3. (8 points) Explain why the following program will not do what the programmer intended:

```cpp
#include <iostream>
using namespace std;
void AbsValue(double x);

int main() {
    double x = -7.251;
    // Replace x with its absolute value
    AbsValue(x);
}

void AbsValue(double x) {
    if ( x < 0.0 )
        x = -x;
}
```

The argument x is passed by value, so although the parameter x is changed within AbsValue(), there is no change to the value in main().

+4 points "passed by value", +2 if on the right track
+4 points "changes in function but not in main", +2 if on the right track

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

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

a. 5           c. 8
b. 6           d. 10

+ 4 points for correct answer (b); no partial credit
5. (8 points) What will the following program print to the screen? Complete the boxes below.

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

int main() {
    int i = 3, j = 4;
    {
        int j = 5;
        cout << i << " " << j << endl;
        i += j;
    }
    cout << i << " " << j << endl;
}
```

Output:

+2 points for each correct answer

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
</tr>
</tbody>
</table>

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

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

int Sum(int x, int y, int z);

int main() {
    int a = 1, b = 2, c
    cout << Sum(a, b, c) << endl;
}

int Sum(int x, int y, int z) {
    return x + y + z;
}
```

+1 point each correct answer

Arguments: a, b, c
Parameters: x, y, z
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:

```c
int Trace(int array[][MAX_SIZE], int size);
double Trace(double array[][MAX_SIZE], int size);
```

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

```c
double Trace(double array[][MAX_SIZE], int size) {
  double tr = 0.0;  
  if (size <= MAX_SIZE) {
    for (int i = 0; i < size; ++i) {
      tr += array[i][i];
    }
  }
  return tr;
}
```
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.

/*
* Trace() - compute the trace of a double matrix +3 function name
*
* Preconditions
*   The size-by-size sub-array of array contains +3 precon. #1
*     valid data.
*   size is >= 1 and <= MAX_SIZE +3 precon. #2
* Postconditions
*   returns the trace of the size-by-size matrix +3 postcon.
*/

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.

   do-while   +2 points

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

   for   +2 points

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.

   while   +2 points

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

   for   +2 points
10. (8 points) What output is produced by the following code?

```cpp
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;
```

+2 points for each correct answer
<table>
<thead>
<tr>
<th>Page</th>
<th>Points</th>
<th>Earned</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>