The University of Alabama in Huntsville
Electrical and Computer Engineering
CPE 112 01
Test #3
October 30, 2002

Name: _____

True or False (2 points each)

1. _____ In the C++ program fragment

```
count = 1;
while (count < 10)
    count++;
cout << "Hello";
```

the output statement that prints "Hello" is not part of the body of the loop.

2. _____ In C++, an infinite loop results from using the assignment operator in the following way:

```
while (gamma = 2)
{
 ...
}
```

3. _____ It is possible to supply different argument names every time a function is called.

4. _____ If a module consists of only a single line, it is usually best to code it directly into the program rather than turn it into a function.

5. _____ If there are several items in a parameter list, the compiler matches the formal and arguments by their relative positions in the formal and argument lists.

6. _____ Any parameter that can be classified as both incoming and outgoing must be coded as a reference parameter.

7. _____ A switch expression can be of any simple data type.

8. _____ The body of a Do-While loop will always execute at least once, whereas the body of a For loop may never execute.

Multiple Choice (2 points each)

9. _____ What is the value of `loopCount` after control exits the following loop?

```
loopCount = 1;
while (loopCount <= 145)
{
    alpha = alpha + 7;
    loopCount++;
}
```

a. 1
b. 144
c. 145
d. 146

10. _____Parameter passage by value is used if a parameter's data flow is

a. one-way, into the function.
b. one-way, out of the function.
c. two-way, into and out of the function.
d. a and b above
e. b and c above

11. ____ Indicate where (if at all) the following loop needs a priming read.

```
sum = 0;                        // Line 1
while (inFile)                  // Line 2
{                               // Line 3
  sum = sum + number;     // Line 4
 inFile >> number;              // Line 5
}                               // Line 6
```

a. between lines 1 and 2
b. between lines 2 and 3
c. between lines 3 and 4
d. between lines 4 and 5
e. No priming read is necessary.

12. _____Given the input data

```
25 10 6 -1
```

what is the output of the following code fragment? (All variables are of type `int`.)

```
sum = 0;
cin >> number;
while (number != -1)
{
 cin >> number;
 sum = sum + number;
}
cout << sum << endl;
```

a. 15
b. 41
c. 40
d. 16
e. no output--this is an infinite loop

13. ____ Given the function prototype

```
void Fix( int&, float );
```

which of the following is an appropriate function call? (`someInt` is of type `int`, and `someFloat` is of type `float`.)

a. `Fix(24, 6.85);`
b. `someFloat = 0.3 * Fix(someInt, 6.85);`
c. `Fix(someInt + 5, someFloat);`
d. a and c above
e. none of the above

14. ____ Consider the function definition

```
void DoThis( int& alpha,
             int  beta  )
{
    int temp;

    alpha = alpha + 100;
    temp = beta;
    beta = 999;
}
```

Suppose that the caller has integer variables `gamma` and `delta` whose values are 10 and 20, respectively. What are the values of `gamma` and `delta` after return from the following function call?

```
DoThis(gamma,delta);
```

a. `gamma` = 10 and `delta` = 20
b. `gamma` = 110 and `delta` = 20
c. `gamma` = 10 and `delta` = 999
d. `gamma` = 110 and `delta` = 999
e. none of the above

15. ____ What is the output of the following code fragment? (All variables are of type `int`.)

```
limit = 8;
cout << 'H';
loopCount = 10;
do
{
    cout << 'E';
    loopCount++;
} while (loopCount <= limit);
cout << "LP";
```

a. HLP
b. HELP
c. HEELP
d. HEEELP
e. none of the above

16. ____ Which of the loops below produces the same number of loop iterations as the following loop? (`count` s of type `int`.)

```
for (count = 1; count <= 10; count++)
    DoSomething();
```

a.
```
for (count = 10; count >= 1; count--)
    DoSomething();
```
b.
```
for (count = 0; count < 10; count++)
    DoSomething();
```
c.
```
for (count = 10; count >= 0; count--)
    DoSomething();
```
d. a and b above
e. a, b, and c above

17. ____ Which For loop is equivalent to the following While loop? (All variables are of type `int`.)

```
count = -5;
while (count <= 15)
{
    sum = sum + count;
    count++;
}
```

a.
```
for (count = -5; count <= 15; count++)
    sum = sum + count;
```

b.
```
for (count = -5; count <= 15; count++)
{
    sum = sum + count;
    count++;
}
```

c.
```
for (count = -5; count <= 15; count++)
{
    count++;
    sum = sum + count;
}
```

d.
```
for (count = 1; count <= 21; count++)
    sum = sum + count;
```

18. ____ Which of the following is *not* a C++ looping control structure?
    a. While
    b. For
    c. Do-While
    d. Switch

Fill in the Blank (2 points each)

19. A(n) _____ is an individual pass through, or repetition of, the body of a loop.

20. The operators &&, `||`, and `!` are known as _____ operators.

21. A(n) _____-controlled loop is one in which the number of repetitions is known in advance.

22. When a function uses a(n) _____ parameter, the parameter receives a copy of the argument's value.

23. The pass by _____ mechanism is used for the parameter `alpha` below:

```
void DoSomething( char& alpha,
                  char  beta  );
```

24. A(n) _____ is a variable or expression listed in a call to a function.

25. A Switch statement, like an If statement, is a(n) _____ control structure.

26. (12 points) What is the output of the following program segment? (All variables are of type `int`.)

```
for (i = 1; i <= 5; i++)
{
   sum = 0;
   for (j = 1; j <= i; j++)
      sum = sum + j;
   cout << sum << Ô Ô;
}
```

27. (14 points) Write a `void` function that reads in a specified number of `float` values and returns their average. A call to this might look like

```
GetMeanOf (5, mean);
```

where the first argument specifies the number of data values to be read, and the second argument contains the result. Document the data flow of each parameter with `/* in */`, `/* out */`, or `/* inout */`.

28. (12 points) Write a program segment that uses a for loop to sum the odd integers from 16 through 36, inclusive.

29. (7 points) the program below has a function named `Change`. Fill in the values of all variables before and after the function is called. Then fill in the values of all variables after the return to the main function. (If any value is undefined, write U instead of a number.)

```
#include <iostream>
using namespace std;
void Change (int, int&);
int main()
{
   int a;
   int b;

   a = 10;
   b = 7;
   Change (a, b);
   cout << a << Ô Ô<< b << endl;
   return 0;
}
void Change (int  x,
             int& y)
{
   int b;

   b = x;
   y = y + b;
   x = y;
}
```

Just before `Change` is called:

a _____

b _____

In `Change` when control enters the function:

x _____

y _____

b _____

After return from `Change`:

a _____

b _____

30. (5 points) Number the marked statements in the following program to show the order in which they are executed.

```
      #include <iostream>
      using namespace std;
      void DoThis (int&, int&);
      int main()
      {
         int number1;
         int number2;
_____    cout << ÒExercise Ó;
_____    DoThis (number1, number2);
_____    cout << number1 << Ô Ô << number2 << endl;
         return 0;
      }
      void DoThis (int& value1,
                   int& value2)
      {
         int value3;

_____    cin >> value3 >> value1;
_____    value2 = value1 + 10;
   }
```