Exam II Instructions

General Instructions:

- Neatness counts!! If the machine cannot read your answer, you will receive no credit.
- This is a closed book/closed notes exam.
- No reference materials of any kind will be permitted.
- No calculators or other portable computing/data storage devices are permitted.
- Any C++ code segments that you write must be syntactically correct or you will lose points.

There are two parts to this exam:

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Bubble Sheet – True/False, Multiple Choice, Vocabulary, fill in the blank</td>
</tr>
<tr>
<td>II</td>
<td>Program writing</td>
</tr>
</tbody>
</table>

Bubble Sheet Instructions: (Part I)

1) Use a #2 pencil to complete the bubble sheet!
2) Print your name in the Name box as follows:  

   LASTNAME □ FIRSTNAME □ MIDDLEINITIAL

   Example:  Simpson □ Homer □ J

3) Use your #2 pencil to fill in the corresponding bubbles under each character of your name.
   (Bubble marking instructions are included on Side 2 of the bubble sheet.)
4) Record your answers to the questions in Part I on Side 1 of the bubble sheet. Neatness Counts!!

Warnings:

- For Part I, answers not recorded on the bubble sheet will receive no credit!!
- Be wary of skipping problems!!
- I recommend that you answer each question in this section in the order presented.
  ➢ Make sure that the answer you record on the bubble sheet for question X corresponds to your selection for question X.
  ➢ You may lose a large number of points if you do not follow this advice!!

Instructions for Part II (30 Points)

1) Answer the problems on the separate test pages handed out. Write your name on the top sheet and turn these sheets in with the bubble sheet.
Part I [120 points] – For each problem, select the best answer and record it on the bubble sheet!!

1) A do-while loop is executed one or more times?
   A) True    B) False

2) A struct declaration must end in a semicolon?
   A) True    B) False

3) While loops can be nested (one loop inside of another loop)?
   A) True    B) False

4) Local variables cannot be accessed outside of the block in which they are declared?
   A) True    B) False

5) In C++, a function definition cannot be nested within another function definition.
   A) True    B) False

6) A switch statement must have at least one case label?
   A) True    B) False

7) The default label in a switch statement is required?
   A) True    B) False

8) The following do-while loop will compile (it is valid syntax)?
   ```
   do-while(true)
   cout << "This do-while loop compiles?\n";
   ```
   A) True    B) False

9) A union variable contains only information for one of its members at any given time?
   A) True    B) False

10) If a function assigns a new value to a value parameter, the value of the corresponding argument is automatically overwritten with the new value.
    A) True    B) False

11) The scope of a function parameter is identical to the scope of a local variable declared in the outermost block of the function body.
    A) True    B) False

12) Function names have local scope in C++.
    A) True    B) False
13) An argument appears in the function heading, and a parameter appears in the function call.

   A) True       B) False

14) If a function has a pass-by-reference parameter, the corresponding argument may be a variable, a constant, or an arbitrary expression.

   A) True       B) False

15) A loop that executes a specified number of times is called a count-controlled loop.

   A) True       B) False

16) A ______________ loop is a loop that executes a specified number of times.

   A) While       B) Count-Controlled       C) Looping       D) Event-Controlled       E) None of These

17) A(n) ______________ loop is a loop that terminates when something happens inside the loop body to signal that the loop should be exited.

   A) Sequence       B) Selection       C) Looping       D) Sub-program       E) None of These

18) Which of the following is not an allowed aggregate operation on structure variables?

   A) Assignment       B) Comparison       C) Return as a functions return value       D) Argument Passed by Reference       E) None of These

19) ____________________ is an assertion that should be true after a module has executed.

   A) Precondition       B) Logical Expression       C) Relational Expression       D) Postcondition       E) None of These

20) A(n) ______________ counter is a variable that is incremented with each iteration of a loop.

   A) Iteration       B) Event       C) Static       D) Expression       E) None of These

21) A(n) ______________ parameter is a parameter that receives a copy of the value of the corresponding argument.

   A) Function       B) Value       C) Variable       D) Reference       E) None of THESE

22) A(n) ______________ parameter is a parameter that receives the location (memory address) of the caller’s argument.

   A) Function       B) Value       C) Variable       D) Reference       E) None of THESE
23) ________________ is the region of program code where it is legal to reference (use) an identifier.

A) Scope  B) Non-Local Identifier  C) Name Precedence  
D) Local Identifier  E) None of these

24) ________________ is the precedence that a local identifier in a function has over a global identifier with the same name.

A) Scope  B) Non-Local Identifier  C) Name Precedence  
D) Local Identifier  E) None of these

25) With respect to a given block, a ________________ is any identifier declared outside that block.

A) Scope  B) Non-Local Identifier  C) Name Precedence  
D) Local Identifier  E) None of these

26) How many times does the body of the following WHILE loop execute? (All variables are of type int)

```c
sum = 0;
count = 0;
while (count <= 10) {
    cin >> number;
    sum = sum + number;
    count = count + 1;
}
```

A) 9  B) 10  C) 11  D) 12  E) None of the above

27) How many times does the body of the following WHILE loop execute? (All variables are of type int)

```c
count = 0;
while ( !(count == 3) ) {
    cout << count << endl;
    count++;
}
```

A) 3  B) 4  C) 5  D) 6  E) None of the above

28) What is the output for the following segment of code? Assume all variables are declared as integers.

```c
sum =0;
for (j = 0; j < 5; j++){
    sum += j;
    cout << j;
}
```

A) 0  B) 5  C) 10  D) 15  E) Nothing, it is an infinite loop
29) How many times does the body of the following WHILE loop execute?
(All variables are of type int)

```c++
sum = 0;
count = 1;
while (count > 10)
{
    cin >> number;
    sum = sum + number;
    count = count + 1;
}
```

A) 9  B) 10  C) 11  D) 12  E) None of the above

30) What is the output for the following segment of code, if the numbers entered are 1 3 5 7 9?
Assume all variables are declared as integers.

```c++
sum = 0;
cin >> number;
do
{
    sum = sum + number;
    number++;
} while (number < 5);
cout << sum;
```

A) 4  B) 8  C) 9  D) 10  E) None of These

31) What is the output for the following segment of code, if the numbers entered are 1 3 5 7 9?
Assume all variables are declared as integers.

```c++
sum = 0;
cin >> number;
do
{
    cin >> number;
    sum = sum + number;
} while (number <= 5);
cout << sum;
```

A) 4  B) 8  C) 9  D) 10  E) None of These
32) What is the output for the following segment of code? Assume all variables are declared as integers.

```cpp
sum = 0;
for (j = 1 ; j <7; j++)
{
    if ( j%3 == 0 )
        continue;
    else
        sum = sum +j;
}
cout << sum;
```

A) 12  B) 21  C) 0  D) 3  E) None of These

33) What is the output for the following segment of code? Assume all variables are declared as integers.

```cpp
sum = 0;
for (j = 1 ; j <7; j++)
{
    if ( j%3 == 0 )
        break;
    else
        sum = sum +j;
}
cout << sum;
```

A) 12  B) 21  C) 0  D) 3  E) None of These

34) The `void` function named `GetNums` has two parameters:

- a pass-by-value parameter named `x` of type float
- a pass-by-reference parameter named `num` of type int.

Which of the following choices is a valid function heading for the function `GetNums`?

A) `void GetNums( float , int& )`
B) `void GetNums( float x , int num )`
C) `void GetNums( float x , int& num )`
D) both A and C
E) none of the above

35) The `void` function named `GetNums` has two parameters:

- a pass-by-value parameter named `x` of type float
- a pass-by-reference parameter named `num` of type int.

Which of the following choices is a valid function prototype for the function `GetNums`?

A) `void GetNums( float , int& )`;
B) `void GetNums( float x , int num )`;
C) `void GetNums( float x , int& num )`;
D) both A and C
E) none of the above
36) What is the output of the following code segment if num has a value of 3? Assume all variables are integers.

```cpp
switch(num)
{
    case 1: cout << "a";
    case 2: cout << "b";
        break;
    case 3: cout << "c";
    default: cout << "end";
}
```

A) abced B) abc C) ab D) c E) None of These

37) What is the output of the following code segment if num has a value of 2? Assume all variables are integers.

```cpp
switch(num)
{
    case 1: cout << "a";
    case 2: cout << "b";
    case 3: cout << "c";
    default: cout << "end";
}
```

A) bcend B) c C) abcend D) end E) None of These

38) The following syntax template correctly describes the syntax of a WHILE loop in C++.

```
WhileStatement
    while ( Expression );
    Statement
```

A) True B) False

39) An individual pass through, or repetition of, the body of a loop is called a(n) __???__.

A) Loop test B) Priming read C) Termination condition D) Iteration E) None of the above

40) A variable that is incremented each time a particular event occurs is called a(n) __???__.

A) Loop control expression B) Iteration counter C) Sentinel value D) Event counter E) None of the above
41) Which of the following, if any, are examples of count-controlled loops?

A) Sentinel-controlled loop  
B) End-of-file-controlled loop  
C) Flag-controlled loop  
D) B and C only  
E) None of the above

42) What type of WHILE loop is illustrated in the code segment below? (All variables are of type `int`)

```cpp
count = 0;  
cin >> someInt;  
while (cin)  
{  
  count = count + 1;  
  cout << someInt << endl;  
  cin >> someInt;  
}
```

A) Sentinel-controlled loop  
B) Flag-controlled loop  
C) Count-controlled loop  
D) End-of-file-controlled loop

43) How many integer values has the user input when the following while loop terminates? (All variables are of type `int`)

```cpp
sum = 0;  
count = 1;  
while (count <= 10)  
{  
  cin >> number;  
  sum = sum + number;  
  count = count + 1;  
}
```

A) 9  
B) 10  
C) 11  
D) 12  
E) None of the above

44) How many times does the body of the following WHILE loop execute? (All variables are of type `int`)

```cpp
count = 0;  
while ( !(count == 3) )  
{  
  cout << count << endl;  
  count++;  
}
```

A) 3  
B) 4  
C) 5  
D) 6  
E) None of the above

45) Pass by value should be used if a parameter’s data flow is

A) two-way, into and out of the function  
B) one-way, into the function  
C) one-way, out of the function  
D) A and B only  
E) B and C only
46) Pass by reference should be used if a parameter's data flow is

A) two-way, into and out of the function
B) one-way, into the function
C) one-way, out of the function
D) A and B only
E) B and C only

47) The void function named GetNums has two parameters:

- a pass-by-reference parameter named x of type float
- a pass-by-reference parameter named num of type int.

Select the most appropriate function heading from the choices listed below.

A) void GetNums( float& , int& )
B) void GetNums( float x , int num )
C) void GetNums( float x& , int& num )
D) both A and C
E) none of the above

48) A function SomeFunc has two parameters, alpha and beta, of type int. The data flow for beta is one-way, out of the function. The data flow for alpha is two-way, into and out of the function. What is the most appropriate function heading for SomeFunc?

A) void SomeFunc( int alpha, int beta )
B) void SomeFunc( int alpha, int beta )
C) void SomeFunc( int& alpha, int beta )
D) void SomeFunc( int alpha, int& beta )
E) None of the above

49) The region of program code where it is legal to reference (use) an identifier is referred to as the _???_ of the identifier.

A) Logical expression
B) Side effect
C) Scope
D) Lifetime
E) None of the above

50) A variable for which memory remains allocated throughout the execution of the entire program.

A) Automatic variable  B) Static variable  C) Regular variable  D) Variable declaration

51) A _???_ is a dummy function that assists in testing part of a program. It has the same name and interface as the actual function, but it is usually much simpler.

A) Driver
B) Void function
C) Stub
D) Value-returning function
E) None of the above
52) What is the output of the following code segment?

```cpp
int sum = 0; int counter = 0;
do{
    sum = sum + counter;
    sum++;
}while(counter < 5);
cout << sum << endl;
```

A) 5          B) 10          C) 15
D) Nothing – it is an infinite loop          E) A compile error occurs

53) What is the output of the following code segment? Assume that inFile is a valid, open file stream
for reading. The values read are: -1 1 -2 2 -3 3

```cpp
int value;
for(k=0 ; k < 5 ; k++)
{
    inFile >> value;
    if (value < 0 )
        continue;
    cout << value;
}
```

A) –1-2-3           B) –1-2           C) 12           D) –1-2-3           E) None of These

54) What is the output of the following code segment? Assume that inFile is a valid, open file stream
for reading. The values read are: -1 1 -2 2 -3 3

```cpp
int k;
int value;
for(k=0 ; k < 5 ; k++)
{
    inFile >> value;
    if (value > 0 )
        break;
    cout << value;
}
```

A) –1-2-3           B) –1-2           C) 12           D) –1-2-3           E) None of These

55) A C++ program can be written without using a `main ()` function?

A) True          B) False

56) Which of the following is not an advantage of using global named constants?

A) The value cannot be accidentally changed while the programming is running.
B) Simplifies program maintenance.
C) Provides for consistency between various program components.
D) A, B, and C are advantages of using global variables.
E) A, B, and C are all advantages of the use of global named constants.
For problems 57 - 60, consider the declarations shown:

```csharp
struct Date
{
    int month;
    int day;
    int year;
};

struct StudentRec
{
    string name;
    int grade;
    Date birthday
};

StudentRec stu1, stu2, stu3;  // variable declarations
```

57) Which of the following assigns a value of 10 to the month member of the birthday structure of stu1?

A) stu1.month = 10;   B) month.birthday.stu1 = 10;   C) stu1.birthday = 10;
D) stu1.birthday.month = 10;   E) None of these

58) The structure StudentRec is an example of a Hierarchical structure?

A) True   B) False

59) Using the above declarations, what is the output of the code below?

```csharp
stu2.name = "Joe";
stu1.name = "Smith";
stu1 = stu2;
cout << stu1.name;
```

A) Smith   B) 30   C) Joe   D) Nothing – A compile error occurs   E) None of these

60) Using the above declarations, what is the output of the code below?

```csharp
stu1.name = "";
stu2.name = "Joe";
stu3.name = "Smith";
stu2.grade = 40; stu3.grade = 30;
stu1 =stu2+stu3;
cout << stu1.name << stu1.grade;
```

A) JoeSmith70   B) Joe30   C) Smith40   D) JoeSmith30   E) None of these
Problems 1-4

Write a C++ value-returning function named `SumOfSquares` that will sum the squares of the integers from 1 through \( N \) inclusive and return the integer sum via the function return value (i.e. return value type `int`). This function has one parameter. Do not use the `pow` (power) function!!!

(Preconditions: \( N \) assigned a value and \( N \geq 1 \))

\( N \) will be an arbitrary value of type `int` supplied by the calling program via an argument.

Hints: 
\[
\sum_{i=1}^{N} i^2 = 1^2 + 2^2 + \ldots + N^2
\]

1. [3 points] Write a **complete** Function Prototype for `SumOfSquares` Here:

2. [3 points] Write the **complete** Function Heading for `SumOfSquares` Here:

3. [3 points] Write the **complete** Function Body for `SumOfSquares` Here:

4. [3 points] Write a `cout` statement which uses the function `SumOfSquares` to compute the sum of the squares of the integers from 1 through 3 and prints that value to standard output:
Problems 5 - 8

Write a C++ void function named \texttt{SumOfSquares} that will sum the squares of the integers from 1 through \texttt{N} inclusive and will use the parameter list to communicate the integer sum to the calling program. This function uses 2 parameters. Do not use the \texttt{pow} (power) function!!!
(Preconditions: \texttt{N} assigned a value and \texttt{N} \(\geq 1\))
\texttt{N} will be an arbitrary value of type \texttt{int} supplied by the calling program via an argument.

Hints:
\[ \text{sum} = \sum_{i=1}^{N} i^2 = 1^2 + 2^2 + \ldots + N^2 \]

5. [3 points] Write a \textbf{complete} Function Prototype for \texttt{SumOfSquares} Here:

6. [3 points] Write the \textbf{complete} Function Heading for \texttt{SumOfSquares} Here:

7. [3 points] Write the \textbf{complete} Function Body for \texttt{SumOfSquares} Here:

8. [3 points] Write the missing function call which uses the function \texttt{SumOfSquares} to compute the sum of the squares of the integers from 1 through 3 so that the value may be printed to the screen.
(Hints: One argument should be the literal integer constant 3)

\begin{verbatim}
int main()
{
    int sum;

    \text{-----------------------------}

    cout << “Sum = “ << sum << endl;
}
\end{verbatim}
Problems 9 - 10

```cpp
void Square( __9__ int& x)
{
    x = x * x;
    return x;
}
```

9. [3 points] For the function `Square` above, write a C++ comment which correctly documents the parameter data flow. Be sure that the comment you write does not introduce an error!

10. [3 points] Assuming an appropriate prototype has been included previously, will the function `Square` above compile correctly? If not, explain why.

    Circle:  Yes  No