True or False – (12 Points)

1. (12 pts) Circle T for true and F for false:

T  F  a) A struct declaration ends with a semicolon.

T  F  b) Local variables retain their value from function call to function call.

T  F  c) Local variables can be accessed only within the block in which they are declared.

T  F  d) Arguments corresponding to value parameters must be variables.

T  F  e) Local identifiers have name precedence over global identifiers.

T  F  f) void functions can use the statement return;

T  F  g) Member names within a structure must be unique.

T  F  h) Members of a structure can be different DataTypes.

T  F  i) A union variable contains only information for one of its members at any given time

T  F  j) The lifetime of a local variable is for the duration of the program

T  F  j) Static variables in a function are deleted when the function finishes execution

T  F  j) A default statement must be used in a switch statement.

Multiple Choice (12 points) – Questions 2 – 13

For these problems circle all correct answers.
For example if answers A, C and E are all valid then circle A, C and E.

2. How many function values does a value-returning function have?

A) 0  B) 2  C) As many as necessary

D) 1  E) 3  F) None of these
3. Which operations below ARE ALLOWABLE aggregate operations on structures?

   A) Input/Output  B) Assignment  C) Comparison
   D) Return as a functions return value E) Arithmetic

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

   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

5. Value parameters (passing by value) are used if a parameters 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
   E) B and C
   F) A and C

6. Reference parameters (passing by reference) are used if a parameters 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
   E) B and C
   F) A and C

7. Which parameters in the following function heading are value parameters?

   void DoSomething(string& date, int& num, float average, float sum, string& name)

   A) date   B) num   C) average D) sum   E) name

8. Which parameters in the following function heading are reference parameters?

   void DoSomething(string  month, int day, float& cost, float& pay, string year)

   A) month   B) day   C) cost   D) pay   E) year
9. A function that **returns a function value** is known as what kind of function?

A) Expression less B) Void C) Empty
D) Reference Parameter E) Value returning F) None of these

10. The **void** function named *GetNums* has two parameters:
A pass-by-reference parameter named *x* of type **float**
A pass-by-value parameter named *num* of type **int**.

Which of the following choices is the most appropriate **function prototype** for the function *GetNums*? (There is only one answer on this problem)

A) `void GetNums( float , int& );`
B) `void GetNums( float& , int );`
C) `void GetNums( float , int );`
D) `void GetNums( float& , int& );`

11. 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**.

Which of the following choices is a valid **function prototype** for the function *GetNums*? (more than one answer is possible)

A) `void GetNums( float&, int& );`
B) `void GetNums (float&, int);`
C) `void GetNums (float x , int& num );`
D) `void GetNums (float& x , int num );`

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

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

13. In loops, a variable that is incremented each time a particular event occurs is called __???__.

A) A loop control counter B) An iteration counter C) A sentinel value D) An event counter E) None of the above
Short Answer (76 points) – Questions 14 – 23

14. (8 pts) There are four functions shown in the code segment below. Assume all variables and function prototypes have been correctly declared before this segment of code.

Compute(inFile, MyCalc());
avg = Average(SumUp(num), num);

A) Which function(s) is(are) most likely value-returning function(s)?

B) Which function(s) is(are) most likely void function(s)?

C) What are the arguments that are used in the function calls?

15. (6 pts) If the numbers entered are 1 1 2 2 5 6 7 8 9, what is the output for the following segment of code? Place one character per box, and skip a box to indicate a space.

```cpp
int sum = 0;
int number;
do
{
    cin >> number;  // read in a number from standard input
    sum = sum + number;
    cout << sum << "-";  
}while (sum < 13);
```
16. (8 pts) When the program shown below is executed, what is the output to the screen? This problem deals with the scope of a variable in a program, and the order of execution of statements.

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

void function_A(int&);
void function_B(int);
int number = 4; // global variable declaration of number
int main()
{
    int number = 3;
    function_A(number);
    cout << "number in main is: " << number << endl;
    return 0;
}
void function_A(int& number)
{
    number = number + 2;
    function_B(number);
    cout << "number in function A is: " << number << endl;
}
void function_B(int sum)
{
    sum = sum - 1;
    cout << "sum in function B is: " << sum << endl;
    cout << "number in function B is: " << number << endl;
}
```

The identifying phrases written by the cout statements in this program are shown below. In the blank to the left of the lines, place 1,2,3 or 4 to indicate the order the statements are printed (1 for first, 4 for last). The value output is placed in the blank at the end of the line.

___ number in main is: ___
___ number in function A is: ___
___ number in function B is: ___
___ sum in function B is: ___
17. (5 pts) Consider the following structure declarations when answering the questions below.

```cpp
struct Date {
    int day;
    int month;
    int year;
};
struct Person {
    string name;
    Date birthday;
};
```

a) Write a statement that declares the identifier `bday` as a variable of DataType `Date`.

b) Write a `cout` statement that will output the value of `month` of the variable `bday`.

c) Write a statement that declares the identifier `student` as a variable of DataType `Person`.

d) Write a statement that assigns a value of “John” to the `name` member of `student`.

e) Write a `cout` statement that will output the value of `year` of the `birthday` member of `student`.

18. (5 pts) Write the type declaration for a struct DataType named `LogType` containing the following members:
   - an integer variable representing the log entry number
   - a string variable representing the name of the person making the log entry
   - a floating-point variable indicating the cost of the entry
   - A `Date` variable indicating the date of the entry where `Date` is a structure already defined
19. (10 pts) Finish the program below by adding the void function specified below. **Add only a function prototype, function call statement and function definition** to the following program. No other information is to be added (i.e. variables)

The name of the void function is **InitStruct**.
The function has one parameter of the struct DataType **Time**.
The function is to initialize each the structure members with a time of 12:00:00 am (form is HH:MM:SS) 
The information stored in the structure **must be available in main()** after the function call.

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

struct Time
{
    int hour;
    int minute;
    int second;
    string designator;  // holds am or pm
};

// Place the function prototype below this line

int main()
{
    Time time;

    // Place the function call statement below this line

    return 0;
}

// Place the function definition below this line
```
20. (8 pts) Rewrite the value returning function definition below as a void function definition such that the caller of the function still has access to the result (contained in the function variable sum) that is being returned by the value returning function.

- Use three parameters (one reference and two value) with the void function.

```c++
float FindSum(float average, int number)
{
    float sum;
    sum = average*number;
    return sum;
}
```

21. (8 pts) What is the output for the following segment of code? Place one character per box and skip a box to indicate a space.

```c++
for (int i = 0 ; i < 5; i++)
{
    cout << i;
    for (int j = 0; j < 5; j++)
    {
        if ( j >= 2)
            continue;
        cout << "-" << j;
    }
    cout << "-" << j << endl;
    if ( i >= 3)
        break;
}
```
22. (8 pts) For the following code segment, write out what is printed to the screen. Show the displayed output precisely by using the following rules:
- Write one character per box.
- Skip a box to indicate the presence of a blank space in the output.
- Skip a row to indicate the presence of a blank line in the output.

```cpp
#include <iostream>
using namespace std;
void Test(int);
int main()
{
    Test(1);
    Test(2);
    Test(3);
    Test(4);
    return 0;
}
void Test(int add)
{
    static int i = 5;
    static int j = 0;

    i--;
    j = j + add;

    cout << i << "-" << j << endl;
}
```
23. (10 pts) Write a **void function definition** that is described below:

- The name of the void function is **OpenInput**.
- The function has one parameter – an input file stream.
- The function is to:
  a) Prompt the user for the name of an input file and read it.
  b) Open the file.
  c) If the file did not open successfully, print out a message stating as such, reset the input stream variable (code to do this is inFile.clear();)
  d) Repeat steps a,b,c until a file is successfully opened or the user enters ctrl-c.
Extra Credit #1 (2 pts) Given the for loop below, write an equivalent do-while loop.

```cpp
for(int loop = 0; loop <= 10; loop++)
    cout<< " loop is: " << loop << endl;
```

Extra Credit #2 (4 pts) Write a value returning function definition that counts the number of non-blank lines in an input file. The value returned by the function is the number of non-blank lines contained in the input file. This function requires a single parameter – the input file stream.