Closed notes, book and neighbor. **If you have any questions ask them.**

**Notes:**

**Segment of code** – necessary C++ statements to perform the action described – not a complete program

**Program** – a complete C++ program – what you have been writing in lab.

Write clearly and make sure the case of a letter is clear (where applicable) since C++ is case sensitive. **Unless otherwise noted, assume a single space between all words.**

For this test the two-character sequence \n is to be taken to mean the newline character.

There are no INTENTIONAL syntax errors. Assume that all code in this exam will compile. There may be logic errors in some of the code.

---

**Multiple Choice (Questions 1 – 13) 26 Points**

Select all correct answers (multiple correct answers are possible)

1) A(n) __________________ is a function that returns a function value to its caller and is invoked in an expression.

A) Value-returning function  
B) Main Function  
C) Subprogram  
D) Void Function  
E) None of these

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

   A) 0  
   B) 1  
   C) As many as necessary  
   D) 2  
   E) 3  
   F) None of these

3) Circle all of the following that are examples of **event-controlled loops**:

   A) Count-Controlled  
   B) Sentinel-Controlled  
   C) Flag-Controlled  
   D) Previous-Value  
   E) End-Of-File Controlled  
   F) All of these

4) An individual pass through, or repetition of, the body of a loop is called a(n) ____.

   A) Loop test  
   B) Iteration  
   C) Termination condition  
   D) Priming read  
   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) None of these

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) None of these

7) 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

8) 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) Event-Controlled
D) Count-Controlled  E) None of These

9) A(n) ____________________ is a variable used in a function call.

A) Function Call  B) Reference  C) Parameter  D) Argument  E) None of These

10) 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

11) 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

12) ____________________ 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
13) Which of the following can be used as a switch expression? (Circle all correct answers):
   a) bool variable  b) string constant  c) char variable
d) integer variable  e) floating point variable  f) None of Them

14) (6 pts) The void function named ReadData has two parameters:
   a pass-by-value parameter named num of type int
   a pass-by-reference parameter named inFile of type ifstream
   write a valid function prototype and function heading for the function ReadData.

Function Prototype: _________________________________________________________

Function Heading: ___________________________________________________________

15) (6 pts) Assume that the Boolean variables X and Y have the Boolean value false and Z has the Boolean value true. What is the Boolean value of the following expressions?
   a) !(X || !Y) && (X && (Z || !Y)) ___________________

b) Y || Z && (X || !X) ______________________

16) (6 pts) What is the output when the following code is executed?
   int num = 2, loop = 0;
   bool done = false;
   while (!done)
   {
       loop++;
       num = num*2;
       if(num > 31)
           done = true;
       cout << “loop: “ << loop << endl;
   }
17) (14 pts) For the questions A through G, consider the following program. Note just the lines of importance have been displayed here. There is at least one answer for each question, and some may have more than one answer. Provide ALL the answers to a question. As an example, if the question is what line contains a return statement? The answer is: lines 12, 14, 16 and 18.

```cpp
#include <iostream>        // Line 1
using namespace std;       // Line 2

int function_1(int, float&, int);       // Line 3
void function_2(float, int&, float);    // Line 4
void function_3(void);       // Line 5

int main()         // Line 6
{
    int sum1, num, enter;      // Line 7
    float temp, avg;       // Line 8
    function_3();        // Line 9
    function_2(avg,num,temp);      // Line 10
    function_2(avg,function_1(sum1,avg,num),temp); // Line 11
    return 0;         // Line 12
}

int function_1(int sum, float& number, int enter)  // Line 13
{
    return enter;        // Line 14
}

void function_2(float avg, int& count, float number) // Line 15
{
    return;         // Line 16
}

void function_3(void)       // Line 17
{
    return;         // Line 18
}
```

A) Which of the lines indicated are **function prototypes**?

B) Which of the lines indicated include **function calls**?

C) Which lines are **function headings**?

D) What are the **arguments** used in the function call statement for function_1?

E) What are the **parameters** for function_1?

F) Which parameter(s) of function_2 are **value parameters**?

G) Which parameter(s) of function_2 are **reference parameters**?
18) (8 pts) Finish the code segment below that reads (using cin) one line of data containing an unknown number of character strings (words) that are separated by spaces. The final string value on the input line is the sentinel character string “EndOfLine”. The segment is to print out how many character strings were present on the input line – excluding the sentinel character string.

Example of a typical input line: Hello World Today is Thursday EndOfLine
Example of output for the line: 5 words are on the line

```cpp
string word;  // holds the word read from the input line
int count = 0; // count of number of words on the line
```

22) (6 pts) Write a segment using an if-then-else-if statement to print out the following information based on the value of the integer variable height:

“Tall” if height has a value greater than or equal to 72,
“Average” if height has a value between 48 and 72, and
“Short” if height is less than or equal to 48
19) (6 pts) What is the output for the following segment of code?
const int ONE = 1, TWO = 2, THREE = 3, FOUR = 4;
for (int x = 0; x < 6; x++)
{
    switch(x)
    {
        case ONE:   cout << "one-";
        break;
        case THREE: cout << "three-";
                    cout << "Odd";
                    break;
        case TWO:   cout << "two-";
        case FOUR:  cout << "four-";
                    cout << "Even";
                    break;
        default:   cout << "-none-";
    }
    cout << endl;
}
20) (6 pts) What is the output of the following program:
For the following code segment, write out what is printed to the screen.
Place a **single character in each box**, skip a box to indicate a space, and skip a row to indicate a
blank line.

```
#include <iostream>
using namespace std;
void Test();
int main()
{
    Test();  // First Call
    Test();  // Second Call
    Test();  // Third Call
    Test();  // Fourth Call
    return 0;
}
void Test()
{
    int i = 0;
    static int j = 5;
    i++;
    j++;
    cout << i << "—" << j << endl;
}
```
21) (8 pts) **Rewrite** the following segment of code as a **switch statement**. Output is to be the same for both segments of code.

```cpp
int value;
cin >> value;
if (value == 1)
   cout << "Compress\n";
else if (value == 2)
   cout << "Decompress\n";
else if (value == 3)
   cout << "Exit Selected\n";
else
   cout << "Invalid Choice\n";
```
23) (12 pts) True/False questions. Select T for true and F for false.

T F a) An if statement can occur inside the block associated with another if statement.
T F b) The body of a do-while loop executes one or more times.
T F c) Global identifiers have name precedence over global identifiers.
T F d) The use of the statement: return; is valid in a void function.
T F e) A logical expression can consist of a single Boolean variable?
T F f) Local variables in a function maintain their value from function call to function call.
T F g) A switch statement can have more than 1 default label.
T F h) The C++ statement for(;;); creates an infinite loop.
T F i) Reference parameters receive a copy of the arguments value.
T F j) Global variables are to be used to simplify writing function interfaces (headings)?
T F k) In sentinel-controlled loops, the sentinel is a value not expected as normal input?
T F l) A switch statement MUST have a default switch label.

24) (8 pts) For the following code segment, write out what is printed to the screen. Place a single character in each box, skip a box to indicate a space, and skip a row to indicate a blank line.

```cpp
int loop_b;
int loop_a = 3;

while (loop_a > 0)
{
    for (loop_b = 0; loop_b <= loop_a; loop_b++)
    {
        cout << loop_b << “-”;
        loop_a--;
        cout << loop_a << endl;
    }
}
```
25) (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. There will be a total of 4 lines written to the screen from this program. Analysis of this program requires some thought.

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

void function_A(int&);  // function definition
void function_B(int&);  // function definition
int number = 4;          // global variable
int main()
{
    int number = 3;      // local variable in main
    function_A(number);  // call to function A
    cout << "number in main is: " << number << endl;
    return 0;
}
void function_A(int& num)
{
    int number = 2;      // local variable in function A
    function_B(num);     // call to function B
    cout << "number in function A is: " << number << endl;
    num = num +2;        // modify global variable
}
void function_B(int& sum)
{
    sum = sum + 1;       // modify local variable in function B
    cout << "sum in function B is: " << sum << endl;
    cout << "number in function B is: " << number << endl;
}
```

The output for this program is as indicated 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). In the blank at the end of the line, put in the output value.

___ number in main is: ___
___ number in function A is: ___
___ sum in function B is: ___
___ number in function B is: ___
26) (10 pts) In the following code segment, all variables are integers.

```cpp
if (x < 20) 
    cout << "< 20\n";
if (x > 20) 
    cout << "> 20\n";
else 
    cout << "< 20\n";
    cout << "End\n";
```

(Place a single character in each box, skip a box to indicate a space, and skip a row to indicate a blank line.)

a) What is the output of the above segment of code when it is executed and \( x = 25 \)

b) What is the output of the above segment of code when it is executed and \( x = 10 \)
27) (8 pts) Finish the segment of code below so that it counts the number of empty lines in an input file. An empty line when read is equal to a null string which is represented as "". Use the getline function to read each line of the file. Output to the terminal the number of empty lines contained in the file.

```c++
int count;
string line;
ifstream inFile;
inFile.open("In.txt");
// put rest of code below this line.
// do not declare any more variables
```
28) (12 pts) **Write a void function** that opens an output file in the following manner:
1) Prompts the user for a file name, reads the name entered and echo prints the file name,
2) Opens the file name provided and associates it with an output file stream
3) If the file did not open:
   a) Print out an appropriate error message
   b) Clear the output file stream (which will reset it)
   c) Repeat 1, 2 and 3 until a file name is successfully entered

**This function requires a single parameter – an output file stream**
The above steps outline what is necessary for a while loop. However, a do-while loop is more compact if written properly.
Bonus #1 (+5 pts)
The following program is executed. The user enters the integer “4” when prompted for a number.
What is the output to the screen? Just fill in the blanks indicated in the output line shown

Be careful on this problem. Think about what is being performed with which variables

```cpp
#include <iostream>
using namespace std;
void Summation( int num, int& result );
int main ()
{
    int number = -1;
    int result = -1;
    cout << “Enter an integer between 1 and 10: “;
    cin >> number;

    Summation(number, result); // first function call
    Summation(number, result); // second function call

    // show the output for this cout statement only
    cout << “Summation for “ << number<< “ is: “<result <<endl;
    return 0;
}
void Summation( int num, int& result)
{
    static int loop = 0;

    for (loop = num; num >=1; num--)
    {
        result = result + loop;
        num = num -1;
    }
    return;
}
```

Answer: Summation for _________ is __________

Bonus #2 (+5 pts) What is the output of the following code segment if Input.txt contains the following values: 1 20 40 2 10

```cpp
ifstream inFile;
int value;
inFile.open(“Input.txt”);
while(inFile)
{
    inFile >> value;
    if (value < 5)
        continue;
    else
        cout << “value: “ << value << endl;
}
```