1. Record your answers on SCANTRON form 882-E (It is green!)
2. Print your name on your scantron in the space labeled NAME.
3. Print CMSC 1613 in the space labeled SUBJECT.
4. Print the date, 8-2-2011, in the space labeled DATE.
5. Print your CRN, 32359, in the space labeled PERIOD.
6. Print the test number and version, T3/V1, in the space labeled TEST NO.
7. You may not consult your neighbors, colleagues, or fellow students to answer the questions on this test.
8. Mark all the selections that satisfy the question. If selections b and c are valid answers to the question, then mark selections b and c.
9. Darken your selections completely. Make a heavy black mark that completely fills your selection.
10. Answer all 50 questions.
11. Submit your completed scantron at 2:00 p.m. on Tuesday, August 2, 2011 in MCS 115.
1. Finding the sum of the previous two values generates the next value in a Fibonacci sequence. Mathematically, \( F_n = F_{n-1} + F_{n-2} \) where \( F_0 = F_1 = 1 \) For example, \( F_2 = F_1 + F_0 = 1 + 1 = 2 \). Mark expressions that are true.

   a. \( F_4 = 8 \)
   b. \( F_7 = 34 \)
   c. \( F_{10} = 89 \)
   d. \( F_{13} = 377 \)

2. Evaluate the sum shown where \( r = 2 \) and \( n = 17 \).

   \[
   \sum_{i=0}^{n} r^i
   \]

   a. 262,144
   b. 13,072
   c. 13,071
   d. 262,143

3. In what order are the operations executed in the expression \(-b\%c+++a\)?

   a. post increment, unary minus, remainder, addition
   b. unary minus, post increment, addition, remainder
   c. addition, post increment, remainder, unary minus
   d. remainder, post increment, unary minus, addition
4. Identify the flowchart in figure 4 that best describes the flow of control in a C++ for-
statement.

   a. Figure 4.3
   b. Figure 4.4
   c. Figure 4.1
   d. Figure 4.2
5. Declarations in figure 5 describe a deck of cards. Variable D is an instance of a deck of cards. Mark the correct references to member pips in the thirteenth card.

   a.  \textit{D[12].Pips}
   b.  \textit{D.pips[12]}
   c.  \textit{D[12].pips}
   d.  \textit{D.Pips[12]}

   \begin{verbatim}
   enum Suit {clubs,diamonds,hearts,spades};
   enum Pips {deuce,trey,four,five,six,seven,eight,nine,ten,jack,queen,king,ace};
   struct Deck {
       Suit suit[52];
       Pips pips[52];
   };
   Deck D;
   \end{verbatim}

   Figure 5.

6. Identify parameters in the program in figure 6.

   a.  c and d
   b.  b and d
   c.  d and c
   d.  a and b

   \begin{verbatim}
   #include <iostream>
   using namespace std;
   void P(char c){c+=4;}
   void Q(char& d){d+=3;}
   int main() {
       char a('a'),b('b');
       P(a);Q(a);P(b);Q(b);
       cout << endl << a << b << endl;
       return 0;
   }
   \end{verbatim}

   Figure 6.
7. What lines have compilation errors in figure 7? Please note that line numbers given on the left are not part of the program.

   a. 6
   b. 4, 6, 9
   c. none of the other combinations
   d. 6, 9

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#include &lt;iostream&gt;</td>
</tr>
<tr>
<td>2</td>
<td>#include &lt;string&gt;</td>
</tr>
<tr>
<td>3</td>
<td>using namespace std;</td>
</tr>
<tr>
<td>4</td>
<td>struct Pie{</td>
</tr>
<tr>
<td>5</td>
<td>string Flavor;</td>
</tr>
<tr>
<td>6</td>
<td>Pie():Flavor(){}</td>
</tr>
<tr>
<td>7</td>
<td>}</td>
</tr>
<tr>
<td>8</td>
<td>int main()</td>
</tr>
<tr>
<td>9</td>
<td>{ Pie Lemon(&quot;lemon&quot;);</td>
</tr>
<tr>
<td>10</td>
<td>return 0;</td>
</tr>
<tr>
<td>11</td>
<td>}</td>
</tr>
</tbody>
</table>

Figure 7. Program for question 7.

8. Which set of operators include only those that are bitwise operators?

   a. && || !
   b. ~ | &
   c. <<= == != >>=
   d. + - * /

9. Variable s is a C-style string. Which of the following expressions concatenate the letter a to the string s?

   a. s+’a’
   b. strcat(s,’a’)
   c. s[’a’]
   d. strcat(s,”a”)
10. Identify the flowchart in figure 10 that best describes the flow of control in a C++ *do-statement*.

a. Figure 10.3  
b. Figure 10.1  
c. Figure 10.2  
d. Figure 10.4

11. Identify incorrect statements describing the attributes of parameters.

a. Parameters appear between enclosing parentheses where the function is called.  
b. Parameters are defined between enclosing parentheses where the function is defined.  
c. Parameters specify how arguments are passed.  
d. Parameters include a type specification.
12. Select the figure that contains a *makefile* that creates executable file `p07` from file `p07.cpp`, class `Amortization`, and class `Date`?

a. Figure 12.2  
b. Figure 12.4  
c. Figure 12.3  
d. Figure 12.1

---

```plaintext
// Version 1
// Version 1
obj = p07.o Amortization07.o Date07.o

p07: ${obj}
g++ -o p07 ${obj} -lm

p07.o: p07.cpp Amortization.h
g++ -c -g p07.cpp

Amortization07.o: Amortization07.cpp Amortization07.h Date07.h
g++ -c -g Amortization07.h

Date07.o: Date07.cpp Date07.h
g++ -c -g Date07.cpp

Figure 12.1 makefile, version 1, to create executable file `p07` from file `p07.cpp`, class `Amortization`, and class `Date`.
```

```plaintext
# Version 2
# Version 2
obj = p07.o Amortization07.o Date07.o

p07: ${obj}
g++ -o p07 ${obj} -lm

p07.o: p07.cpp Amortization07.h
g++ -c -g p07.cpp

Amortization07.o: Amortization07.cpp Amortization07.h Date07.h
g++ -c -g Amortization07.h

Date07.o: Date07.cpp Date07.h
g++ -c -g Date07.cpp

Figure 12.2 makefile, version 2, to create executable file `p07` from file `p07.cpp`, class `Amortization`, and class `Date`.
```
Figure 12.3 makefile, version 3, to create executable file p07 from file p07.cpp, class Amortization, and class Date.

Figure 12.4 makefile, version 4, to create executable file p07 from file p07.cpp, class Amortization, and class Date.

13. Mark correct statements.

a. The statement of a while-statement is executed one or more times and the statement of a do-statement is executed zero or more times.
b. The statement of a while-statement is executed one or more times and the statement of a do-statement is executed one or more times.
c. The statement of a while-statement is executed zero or more times and the statement of a do-statement is executed zero or more times.
d. The statement of a while-statement is executed zero or more times and the statement of a do-statement is executed one or more times.
14. Evaluate $e^x$ where $x = a \ln b$, $a = 4, b = 2.5$

   a. 32.0000  
   b. 39.0625  
   c. 4.0000   
   d. 25.0000

15. Function $Mgr$ accepts arguments of type $ifstream$ and $ofstream$. Select the best function prototype for function $Mgr$.

   a. `void Mgr(istream i, ostream o);`
   b. `void Mgr(istream& i, ostream& o);`
   c. `void Mgr(ifstream& i, ofstream& o);`
   d. `void Mgr(istream* i, ostream* o);`

16. Identify correct grammatical expressions for the syntax of a `for`-statement.

   a. `for (condition ; statement ; increment-decrement) initialization`
   b. `for (initialization ; condition ; increment-decrement) statement`
   c. `for (increment-decrement ; condition ; initialization) statement`
   d. `for (statement ; condition ; increment-decrement) expression`

17. Identify arguments in the program in figure 17.

   a. a and b
   b. b and c
   c. c and d
   d. a and d

```cpp
#include <iostream>
using namespace std;
void P(char c){c+=4;}
void Q(char& d){d+=3;}
int main()
{
    char a('a'), b('b');
    P(a);Q(a);P(b);Q(b);
    cout << endl << a << b << endl;
    return 0;
}
```

Figure 17.

18. Which of the following is an invalid single-character constant?

   a. `'\?'`
   b. `'\b'`
   c. `'\^'`
   d. `'\n'`
19. Variables $n$ and $d$ have type integer. What values of $n$ and $d$ make the expression $n/d*d==n$ true?

   a. $n=7$, $d=7$
   b. $n=13$, $d=3$
   c. $n=51$, $d=3$
   d. $n=21$, $d=4$

20. Which variable declarations contains no errors or warnings?

   a. unsigned double $ud$;  
   b. long double char $dc$;  
   c. signed long int $si$;  
   d. unsigned real $ur$;

21. Which function does not compute the value of $\sum_{i=1}^{10}(2i+3)$?

   a. int $a$(void){return 2*10*11/2+10*3;}
   b. int $b$(void){return 140;}
   c. int $c$(void){int $s$(0),$k$(0);for (int $i$=1;$i<10;$i++){$s$+=i;$k$+=3;}return 2*$s$+$k$;}
   d. int $d$(void){int $s$=0;for(int $i$=1;$i<10;$i++)$s$+=2*$i$+3{return $s$;}

22. Variables $a$ and $b$ have type bool. What values of $a$ and $b$ make the expression $(a>b)$ true?

   a. $a=false$, $b=true$
   b. $a=true$, $b=true$
   c. $a=false$, $b=false$
   d. $a=true$, $b=false$

23. Identify correct grammatical expressions for the syntax of a while-statement.

   a. while (expression) { statement-list }
   b. do expression while (statement)
   c. do statement while (expression)
   d. while (expression) statement

24. Variable $s$ is a C++-style string. Which of the following expressions determine the length of the string?

   a. $s$.length
   b. $s$.length()
   c. length($s$)
   d. length($s$)
25. A is an array of elements having type `char`. The array has two (2) rows and five (5) columns. Mark the C++ expression that references the element in the first row and the second column.

   a. `A[1][0]`
   b. `A[2][1]`
   c. `*((char*)A+1)`
   d. `*((char*)A+7)`

26. Which of the following sets is not a subset of the set of rational numbers?

   a. the set of real numbers
   b. the set rational numbers
   c. the set whole numbers.
   d. the set integers

27. Select the correct method for accessing a member of in a dynamically allocated structure.

   a. Members of a structure by coding an arrow and a period (->.) between the structure name and the member name.
   b. Members of a structure can be accessed by prefixing the structure name and a commercial at sign (@) to the member name.
   c. Members of a structure can be accessed by prefixing the structure name and a circumflex (^) to the member name.
   d. Members of a structure can be accessed by prefixing the structure name and an arrow (->) to the member name.

28. Mark the incorrect statement describing a C++ structure.

   a. Every member of a structure can be accessed by any function in which an instance of that structure is visible.
   b. Every capability of a class can be implemented in a structure.
   c. Private members of a structure cannot be accessed by functions that are not members of the structure.
   d. A structure is an aggregate where every member must have the same type.

29. Mark the correct parameter declaration that accepts an array of values having type double with six (6) rows and four (4) columns.

   a. `double A[4][6];`
   b. `double A[][4];`
   c. `double* A[6];`
   d. `double* A[4];`
30. Declarations in figure 30 describe a deck of cards. Variable \( D \) is an instance of a deck of cards. Mark the correct reference to member pips in the thirteenth card.

a. \( D[12].pips \)  
b. \( D.Pips[12] \)  
c. \( D.pips[12] \)  
d. \( D[12].Pips \)

```c
enum Suit {clubs, diamonds, hearts, spades};
enum Pips {deuce, trey, four, five, six, seven, eight, nine, ten, jack, queen, king, ace};
struct Card { Suit suit; Pips pips; };
Card D[52];
```

**Figure 30.** Declarations for question 30.

31. Identify the flowchart in figure 31 that best describes the flow of control in a C++ `while` statement.

a. Figure 31.1  
b. Figure 31.2  
c. Figure 31.3  
d. Figure 31.4

![Flowchart](Image)
32. Function main appears directly after the declarations in figure 32. Which implementation of function main is valid?

   a. `int main() {Rectangle R(5,6); cout << R.Area() << endl; return 0;}`
   b. `int main() {Rectangle R(5,6); cout << R.L << endl; return 0;}`
   c. `int main() {cout << Rectangle(5,6) << endl; return 0;}`
   d. `int main() {Rectangle R(5,6); cout << R.Perimeter(5,6) << endl; return 0;}`

#include <iostream>
using namespace std;

class Rectangle {
    double L, W;
public:
    Rectangle(double l=0.0, double w=0.0): L(l), W(w){}
    double Area(void){return L * W;}
    double Perimeter(void){return 2*(L+W);}
};

Figure 32. Declarations for question 32.

33. When is the expression `c<=d==c>=d` true given that `c` and `d` both have type `char`?

   a. for all `c<=d`
   b. for all `c!=d`
   c. for all `c>=d`
   d. for all `c==d`

34. Which figure contains a correct implementation of function `IsVowel`. Function `IsVowel` returns true if its parameter is a vowel. Only letters that are always accorded the status of vowels are considered to be vowels.

   a. Figure 34.1
   b. Figure 34.2
   c. Figure 34.3
   d. Figure 34.4

```cpp
bool IsVowel(char c) {
    static char V[]="aeiou";
    for (int a=0; V[a]; a++) if (tolower(c)==V[a]) return true;
    return false;
}
```

Figure 34.1 function `IsVowel`. 
```c
bool IsVowel(char c) {
    static string V="AaEeIiOoUu";
    for (int a=1;a<=V.length();a++) if (c==V[a]) return true;
    return false;
}
```

**Figure 34.2 function IsVowel.**

```c
bool IsVowel(char c){return c=='a'||c=='e'||c=='i'||c=='o'||c=='u'?true:false;}
```

**Figure 34.3 function IsVowel.**

```c
bool IsVowel(char c)
{
    c=tolower(c);
    if (c=='a') return true;
    if (c=='e') return true;
    if (c=='i') return true;
    if (c=='o') return true;
    if (c=='u') return true;
    return false;
}
```

**Figure 34.4 function IsVowel.**

35. Which prototype of function main is suitable to process command line arguments?

a. `int main(char* argc, int argv[]);`

b. `int main(int argc[], char* argv );`

c. `int main(int argc, char* argv[]);`

d. `int main(char* argc[], int argv);`

36. Mark the selection whose expression evaluates to the corresponding value in table 36.

<table>
<thead>
<tr>
<th>Selection</th>
<th>Variables</th>
<th>Expression</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>int a=4,b=3,c=2;</td>
<td>-a+++b/c</td>
<td>-2</td>
</tr>
<tr>
<td>2</td>
<td>int a=4,b=3,c=2;</td>
<td>- - a+b/c++</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>int a=4,b=3,c=2;</td>
<td>a*=b/c</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>int a=4,b=3,c=2;</td>
<td>a1=b&amp;c</td>
<td>1</td>
</tr>
</tbody>
</table>

**Table 36.**
37. *Makefile* comments

   a. are enclosed between // and a newline character
   b. begin with a # in column 1
   c. are enclosed between /* and */
   d. are enclosed between -- and a newline character

38. Which statement reclaims storage allocated in figure 38?

   a. `A[] delete;`
   b. `delete A[];`
   c. `A delete[];`
   d. `delete[] A;`

   ```
   double* A=new double[100];
   ```

   **Figure 38.** Statement for question 38.

39. Mark the incorrect statement.

   a. A *goto-statement* may be used to transfer control to any statement within a function.
   b. Execution of a *break-statement* terminates the execution of the function in which it appears.
   c. A *continue-statement* terminates the execution of the body of the smallest enclosing *while-statement, do-statement, or for-statement.*
   d. A *return-statement* is used to terminate the current function, perhaps returning a value.
40. $310.25 is deposited every month into an interest-bearing account for ten and one-half (10.5) years. Interest is compounded monthly. The APR on the account is 6.25. Select the value closest to the amount in the account at the end of the term. (Please note that the last deposit is made on the same day as the entire sum is withdrawn and the first payment is made on the first day of the term.)

a. $ 51,539.32  
b. $ 50,802.60  
c. $ 55,653.94  
d. $ 54,272.99

41. Mark syntactically incorrect C++ declarations.

b. char C="January";  
d. int A[][]={0,1,1,2,3,5,8};

42. Identify correct grammatical expressions for the syntax of a do-statement.

a. do expression while (statement)  
b. while (expression) do statement  
c. do statement while (expression)  
d. while (statement) do expression
43. What is displayed by the program in figure 43?

   a. One for the money.
      Three to get ready
   b. One for the money. Two for the show.
      Three to get ready and four to go.
   c. One for the money.
      Two for the show.
      Three to get ready
      and four to go.
   d. Two for the show. and four to go.

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

int main()
{
    for (int a=0; a<4; a++)
    {
        switch (a+1)
        {
            case 1: cout << "One for the money. ";
            case 2: cout << "Two for the show."; break;
            case 3: cout << "Three to get ready ";
            case 4: cout << "and four to go."; break;
        }
        cout << endl;
    }
    return 0;
}
```

**Figure 43.** Program for question 43.

44. What is the value of `floor(-1.56)`?

   a. -1
   b. -2.00
   c. -2
   d. -1.00

45. Select the statement that does not describe an argument.

   a. An argument can be passed by reference if it is a variable.
   b. An argument can be passed by-reference if it is an expression.
   c. An argument can be passed by value if it is a variable.
   d. An argument can be passed by-value if it is an expression.
46. How many times is the expression v>1 on line 5 of figure 46 executed?

a. 5  
b. 4  
c. 17  
d. 16

<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>#include &lt;iostream&gt;</td>
</tr>
<tr>
<td>2</td>
<td>using namespace std;</td>
</tr>
<tr>
<td>3</td>
<td>int main()</td>
</tr>
<tr>
<td>4</td>
<td>{ int v=17;</td>
</tr>
<tr>
<td>5</td>
<td>while (v&gt;1) v/=2;</td>
</tr>
<tr>
<td>6</td>
<td>return 0;</td>
</tr>
<tr>
<td>7</td>
<td>}</td>
</tr>
</tbody>
</table>

Figure 46. Program for question 46.

47. Mark the correct C++ declaration.

a. enum color {red,green,blue};  
b. enum color (red,green,blue);  
c. enum color <red,green,blue>;  
d. enum color [red,green,blue];

48. Evaluate function tu given that parameter c has the value 'c'.

char tu(char c){return 'A'+c-'a';}

a. 2  
b. 3  
c. c  
d. C

49. Mark the incorrect statement regarding a constructor for a C++ class.

a. Two constructors in the same class can have identical parameter lists. 
b. Two constructors in the same class can have the same name. 
c. Constructors are not required to have at least one parameter to distinguish them from other member functions. 
d. All constructors have no return type.
50. Select the statement that correctly describes parameters and arguments.

a. A parameter becomes an alias when the corresponding argument is passed by-value.
b. A parameter is the address of the corresponding argument when it is passed by-value.
c. A parameter becomes an alias when the corresponding argument is passed by-reference.
d. A parameter is a copy of the corresponding argument when it is passed by-reference.