Questions Suggested For Review

The questions on the examination will be of similar format.
If you find a duplicate question, please tell your instructor.

[1] Which of the following can be said to be examples of computer hardware? Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)
   (a) Keyboard
   (b) Linux
   (c) C++ compiler
   (d) Monitor
   (e) Gaddis
   (f) Memory
   (g) Hard disk
   (h) Top-down design
   (i) Structured program design
   (j) Coding
   (k) Software design
   (l) CD reader

[2] Which of the following can be said to be examples of computer’s logical units? Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)
   (a) Keyboard
   (b) Linux
   (c) C++ compiler
   (d) Memory
   (e) Input
   (f) Monitor
   (g) Hard disk
   (h) Top-down design
   (i) Structured program design
   (j) CPU
   (k) Output
   (l) Secondary Storage

[3] At least five of the following can be said to be true of the general area of the software development process. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)
   (a) The ultimate user always directly interacts with the program coder.
   (b) The systems analyst precisely defines the user's needs.
   (c) The systems designer precisely defines the user's needs.
   (d) The programmer produces the logic design for final implementation.
   (e) The systems designer identifies major subsystems and how they interact
   (f) Pseudo-code is sometimes produced by the systems designer.
   (g) A structure-chart is sometimes produced by the systems designer.
   (h) The user is expected to supply needs expressed precisely enough so the needs can be expressed through programming
   (i) The systems analyst produces functional specifications that include hardware layout details.
   (j) The systems designer may develop interface conventions between program modules.
   (k) The systems analyst produces a solution design.
At least five of the following can be said to be true of the general area of **good quality programs**. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) “Verifiable” programs test to see if the input data is correct.
(b) Imprecise program specifications are refined through the process of step-wise refinement.
(c) More time is often spent on program maintenance than is spent on program development.
(d) Testing a program by running it parallel with an old program sometimes is useful.
(e) Global variables help in documentation.
(f) Structure charts provide an effective way of documenting programs as they change.
(g) It is usually better to develop and test modules separately.
(h) Program clarity is enhanced by using shorter variable names.
(i) Many application programs run over a period of years and on different machines.
(j) Small program modules help with program understandability.
(k) Passing data through parameter lists is to be avoided.

Which of the following can be said to be true about **program design**? Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) The implementation cost can be minimized by keeping the parts small
(b) It is best to code the entire program as a single, large piece of code.
(c) Maintenance costs are minimized when the parts are separately correctable.
(d) It is easier to understand a program made of large parts as opposed to small parts.
(e) Unrelated parts of the program should be separated.
(f) Structured design begins when the coder first looks at the problem.
(g) The designer identifies the system components and the relationship between them.
(h) Top-down design starts with the details of what is to be done and finishes with a general statement of the program’s purpose.
(i) Structured program design involves methodically laying out the program structure before coding.
(j) Coding is the most important part of the program development process.
(k) The user should be expected to help in writing the program code.

At least five of the following can be said to be true of the general area of the **code segmentation**. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) The process of breaking up programming code into pieces or hunks is called modularization.
(b) When the hunks are parts of the same code stream, they are often called segments.
(c) Modules are code hunks that can be separated from the rest of the program.
(d) Modules can NOT be functions.
(e) Properly segmented code can be only entered through one path.
(f) In C++, modules can be implemented by functions.
(g) A function is not the same as a code object.
(h) It is desirable to have one module to do several different tasks.
(i) An invoked C++ function has to either receive or emit data.
(j) Good programming style discourages passing data through function parameter lists.
(k) Functions cannot be developed separately from each other.
[7] At least five of the following can be said to be true of the general area of the software development process. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) The ultimate user always directly interacts with the program coder.
(b) The systems analyst precisely defines the user's needs.
(c) The systems designer precisely defines the user's needs.
(d) The programmer produces the logic design for final implementation.
(e) The systems designer identifies major subsystems and how they interact.
(f) Pseudo-code is sometimes produced by the systems designer.
(g) A structure-chart is sometimes produced by the systems designer.
(h) The user is expected to supply needs expressed precisely enough so the needs can be expressed through programming.
(i) The systems analyst produces functional specifications that include hardware layout details.
(j) The systems designer may develop interface conventions between program modules.

[8] At least five of the following can be said to be true of the general area of good quality programs. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) “Verifiable” programs test to see if the input data is correct.
(b) Imprecise program specifications are refined through the process of step-wise refinement.
(c) More time is often spent on program maintenance than is spent on program development.
(d) Testing a program by running it parallel with an old program sometimes is useful.
(e) Global variables help in documentation.
(f) Structure charts provide an effective way of documenting programs as they change.
(g) It is usually better to develop and test modules separately.
(h) Program clarity is enhanced by using shorter variable names.
(i) Many application programs run over a period of years and on different machines.
(j) Small program modules help with program understandability.
(k) Passing data through parameter lists is to be avoided.

[9] Which of the following can be said to be true about data abstraction? Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) An object consists of private memory and a set of operations that may be applied within the object.
(b) A object orientation is a model of communications within a problem implementation space.
(c) An advantage is that it maximizes dependence between system components.
(d) An object orientation can only be applied to the elements of a program (e.g., data, functions, etc.).
(e) Procedural abstraction decides what the program and sub-programs are to do.
(f) Sub-programs need to be developed in close cooperation.
(g) Modularity aid in managing program complexity..
(h) There is a focus on how a data object will be implemented and stored.
(i) There is a focus on how operations will be implemented.
(j) The specification of how something will be done is the logical view of the problem.
(k) An example of abstraction is the specification of how an array will be implemented. 
(l) Information hiding hides low-level details from higher-level modules.
At least five of the following can be said to be true of the general area of the inheritance. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) Inheritance is a relationship between classes.  
(b) Inheritance describes the history of changes in programming code.  
(c) Some properties in derived class are the same as an ancestor class; some are not.  
(d) Functions cannot be inherited.  
(e) Derived class can inherit from multiple other classes.  
(f) Inheritance can be found in nearly all programming languages.  
(g) Inheritance ties together hierarchies of classes.  
(h) Inheritance requires the maintenance of multiple copies of identical member variables.  
(i) Subclasses inherit from superclasses  
(j) Single inheritance means that a single value is passed through a parameter list.  
(k) Multiple inheritance means that both values and functions are inherited.

At least five of the following can be said to be true of the general area of the dynamic memory. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) Dynamic memory is created before the program executes.  
(b) An example of a dynamic memory definition: int temperature [5];  
(c) Before execution, name of space to be allocated is specified.  
(d) Space only allocated when requested by executed statement.  
(e) Dynamic memory best used when know exactly how much data to be stored.  
(f) Dynamic memory should not be used when data is related in complex way.  
(g) Sorts can be done by rearranging dynamic data.  
(h) Dynamic data can be used to form “tag” lists.  
(i) Dynamic data values must change during program execution.  
(j) Dynamic data uses direct reference to find stored data.  
(k) Pointers are used to indirectly refer to stored data.

Computers are said to make logical comparisons. Provide an example of a logical comparison. You do not have to write a correct programming statement.

Computers are said to be a combination of hardware and software. Provide two examples of some software.

According to your instructor, what is the single best measure of the "goodness" of a computer program? (Answer in 25 words or less.)

What is underflow (Answer in 25 words or less.)

What is overflow (Answer in 25 words or less.)

In the process of developing a solution to a problem, both pseudo-code and hierarchy charts may be used. True or False: In general, both pseudo-code and hierarchy charts are used at the same time in the design of a program. Explain. (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

True or False: Pseudo-code should not look like the programming language that will be used to implement the pseudo-code. Explain. (Both parts of your answer must be correct to receive any credit to this question.)
[19] When solving a problem using a structured design approach, top-down analysis and structure charts are used. **True** or **False**: Each and every *fully refined* box on a structure chart should represent as much that is to be done as is possible. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[20] Name the physical *logical units* of a computer. Simply list them. You do not have to provide an explanation.

[21] All computers can do at least four general things. These things are called the “four necessary” characteristics of a computer. List them. You do not have to provide an explanation.

[22] What is used to store peripheral information in a computer? This can be answered with a single word. However, if you wish, you can use up to 25 words.

[23] **True** or **False**: *It is best to have multiple paths into and out of a program segment.* **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[24] The process of program maintenance is to change programs once they have been fully tested and delivered to the customer. **True** or **False**: In general, in a properly developed program, less programmer time is spent on maintenance than on original program development. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[25] Given the following program fragment, what does the statement “cout << endl;” accomplish? Answer in 25 words or less.

```cpp
#include <iostream.h>
int main ( )
{
    cout << endl;
    cout << "My name is Chester";
    return;
}
```

[26] In the process of developing a solution to a problem, both *pseudo-code* and *hierarchy charts* may be used. **True** or **False**: In general, both pseudo-code and hierarchy charts are *used at the same time in the design of a program.* **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[27] **True** or **False**: In minimizing a computer project's cost, the most important thing to minimize is the size of the computer. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[28] **True** or **False**: When a C++ function ends, data in the function that was “passed by value” into the function is no longer easily available to the C++ program invoking the function. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[29] In a structure chart, the following symbols represent information flow between modules. **True** or **False**: The only difference between the type of information represented by the symbols is the direction that the
information is going. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[30] **True** or **False**: In top-down analysis of a problem, the upper levels of the hierarchy chart are concerned as much as possible with defining functional requirements while the lower levels of the hierarchy chart are concerned as much as possible with establishing control. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[31] Structured program development is a particular way of designing and implementing a programming project. **True** or **False**: Structured program development builds a program as a builder builds a house. That is, the details are first discovered, then the details are combined into larger pieces, and the larger pieces are combined into still larger pieces until the project is completed. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[32] **True** or **False**: Structure charts show all of the following:
- partitioning of the system into modules
- hierarchy and organization of the system
- communication between modules
**Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[33] **True** or **False**: Structure charts show all of the following:
- internal module mechanics
- data internal to the modules
**Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[34] **True** or **False**: Pseudo-code should **not** look like the programming **language** that will be used to implement the pseudo-code. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[35] When solving a problem using a structured design approach, top-down analysis and structure charts are used. **True** or **False**: Each and every fully refined box on a structure chart should represent as much that is to be done as is possible. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[36] **True** or **False**: When a C++ function ends, the data in the function that was “passed by reference” is no longer easily available to the C++ program invoking the function. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

[37] **True** or **False**: The only classes that can be used in a C++ program are those defined explicitly in the program by the programmer. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

[38] **True** or **False**: The **Turing Test** is used to determine how modular a program is. **Explain.** (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)
[39] True or False: Data Mining is used to find dynamically stored data. Explain. (Both parts of your answer must be correct to receive any credit to this question.)

[40] In the following diagram, supports person, four legs, has back, hard are all characteristics that the specific types of chairs might inherit from chair. Which of these characteristics are inherited by all of the specific types of chairs (i.e., upholstered chair, kitchen chair, doll house chair)?

[41] True or False: In the following program, 
```c
setw (3)
```
changes the value of 
count Explain. (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)

```c
#include <iostream.h> // library for stream I/O
#include <iomanip.h>  // library for manipulation output
int main ()
{
    int count;
    cout << "Count" << endl; // title line
    cout << "-----" << endl; // title line
    count = 1;
    // output for count will be 3 columns wide
    // output will be right hand aligned
    cout << setw (3) << count << endl;
}
```

[42] Which of the following can be said to be true about a structured walk through? Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) Management organizes them
(b) Management tracks the results
(c) They are a form of peer review
(d) They consider all of the aspects of a project at the same time
(e) Programs that have been diligently walked through have about ten times fewer bugs
(f) Programs that have been diligently walked through have about half as many bugs
(g) During a walk through, experienced people are expected to instruct junior people
(h) They increase standards conformability
(i) They are primarily done when the program is completed
(j) They are primarily done to clarify specifications
(k) They do NOT help with debugging
Which of the following can be said to be true about the general area of coupling and cohesion. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) A module has only one data path in and one data path out.
(b) Coupling is a measure of how functionally independent one model is from another.
(c) The most effectively modular system is one where the sum of the coupling is maximized.
(d) An incremental module cannot be interrupted prior to completion.
(e) Parallel modules imply a multi-processor environment.
(f) The lowest degree of coupling is achieved when there is no direct connection between modules.
(g) High cohesion and high coupling generally occur simultaneously.
(h) Cohesion is the measure of the relative functional strength of a module.
(i) Temporal cohesion is more cohesive than logical cohesion.
(j) Coincidental cohesion is more cohesive than logical cohesion.
(k) Coupling keeps track of how many times a module is used.

At least five of the following can be said to be true of the general area of top-down problem analysis. Circle the letter of your answer. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) The problem is successively subdivided until the divided parts can be easily handled.
(b) The method starts by identifying the most detailed parts and then successively grouping them together until the problem can be stated as a single whole.
(c) Step-wise refinement is part of the top-down analysis process.
(d) Top levels are mostly concerned with functionality.
(e) Top levels are mostly concerned with control.
(f) Control and function are separated.
(g) Software structure can be shown by structure charts.
(h) Top-down design leads to segmentation, but not modularization.
(i) Productivity is decreased due to segmentation.
(j) Testing problems are increased because of the need to handle numerous modules.
(k) Program clarity is reduced because the program becomes hard to follow.

True or False: In general, the more sophisticated, elegant, and complex a correctly running program is, the better it is. Explain. (Both parts of your answer must be correct to receive any credit to this question.) (Answer in 25 words or less.)
Given the following program, which of the following is correct?. (Scoring: +1 for every correct answer, up to a maximum of +5; -1 for every incorrect answer.)

(a) Container is a derived class
(b) Box is a derived class
(c) Volume is a member function
(d) RailRoadCar is a base class
(e) Age is not a function
(f) BoxCar is a base class
(g) freight can be specified by an assignment statement
(h) freight width can be specified by an assignment statement
(i) Age is inherited
(j) Container inherits from Box
(k) RailRoadCar inherits from Container

```cpp
// --------------------------------------------------------------
class Container
{  
    public: int percentLoaded; 
};
// --------------------------------------------------------------
class Box:
    public Container
    {  
        public: double height, width, length;
        double Volume (); 
    };
// --------------------------------------------------------------
double Box::Volume ()
    {  
        double cubes;
        cubes = height * width * length;
        return cubes; 
    };
// --------------------------------------------------------------
class RailRoadCar
{  
    public: int yearBuilt;
    int Age (); 
};
// --------------------------------------------------------------
int RailRoadCar::Age ()
{  
    int currentYear;
    int years;
    currentYear = 2002;
    years = currentYear - yearBuilt;
    cout << years << " " << currentYear;
    return years; }
// --------------------------------------------------------------
class BoxCar:
    public Box,
    public RailRoadCar
{  
};
// --------------------------------------------------------------
int main() {
    BoxCar freight;
    freight.height=10.0;
    freight.width=9.0;
    freight.length=40.0;
    cout << freight.Volume () << endl;
    freight.yearBuilt = 1941;
    cout << endl << freight.Age ();
}
```
[47] Given the following program:

```cpp
#include <iostream.h>
// -----------------------------
class Node {
    public: int item;
    Node* next;
};
// -----------------------------
class Cell {
    public: int value;
    int count;
};
// -----------------------------
int main () {
    Node* pointer;
    pointer = new Node;
    Cell exam;
    pointer -> item = 7;
    exam.value = 6;
    cout << exam.value << " " << pointer -> item;
}
```

Chose one of the following and then explain your choice. Both parts of your answer must be correct to receive credit.
(a) Memory space for “item” is allocated before memory space for “value”
(b) Memory space for “value” is allocated before memory space for “item”
(c) Memory spaces for “item” and “value” are allocated at the same time

**Explain:**

[48] Given the following program, what is the first multiplication statement executed? Circle your answer. Graded right/wrong.

```cpp
#include <iostream.h>
int CarVolume (int high, int wide, int reach) {
    int volume;
    volume = high * wide * reach;
    return volume; } // value of variable returned
// -----------------------------
void main ( ) {
    int height; // height of a box car
    int width; // width of a box car
    int length; // length of a box car
    int capacity;
    cout << endl << "enter height" << endl;
    cin >> height;
    cout << endl << "enter width" << endl;
    cin >> width;
    cout << endl << "enter length" << endl;
    cin >> length;
    capacity = height * width * length;
    cout << CarVolume(height,width,length);
}
```
[49] Given the following program, what is declared but NOT defined? Circle as your answer everything that is declared but NOT defined. (This question is graded right/wrong.)

```c
#include <iostream.h>
int CarVolume (int high, int wide, int reach) {
    int volume;
    volume = high * wide * reach;
    return volume; } // value of variable returned
// -------------------------------
void main ( ) {
    int height; // height of a box car
    int width; // width of a box car
    int length; // length of a box car
    int capacity;
    cout << endl << "enter height" << endl;
    cin >> height;
    cout << endl << "enter width" << endl;
    cin >> width;
    cout << endl << "enter length" << endl;
    cin >> length;
    capacity = height * width * length;
    cout << CarVolume(height,width,length);
}
```

[50] Given the following program, what lines could be removed without affecting the way the program would execute? There are at least five such lines. Place an “x” in front of these statements. You will be given credit for no more than five correct answers. Each correct answer will receive a plus one. Each incorrect answer will receive a minus one.

```c
// Purpose: This is an example of how to:
// (a) read in two values,
// (b) add them together
// (c) compute their total
// (d) display their total
/
#include <iostream.h>
int main ( )
{
    int value1; // this is the first example value to be input
    int value2; // this is the second example value to be input
    int totalValues; // the total of both input values
    cin >> value1; // read in the first value
    cin >> value2; // read in the second value
    total = value1 + value2; // total input values
    cout << totalValues; // display total of two values
} // end example I/O program
```
[51] **True or False:** In the following program,

```cpp
    setw (3)
```
changes the value of

**count**

**Explain.** (Both parts of your answer must be correct to receive any credit to this question.)

(Answer in 25 words or less.)

```cpp
#include <iostream.h> // library for stream I/O
#include <iomanip.h>  // library for manipulation output
int main ()
{
    int count;
    cout << "Count" << endl; // title line
    cout << "-----" << endl; // title line
    count = 1;
    // output for count will be 3 columns wide
    // output will be right hand aligned
    cout << setw (3) << count << endl;
}
```

**In the programs:**

• **If input data is shown, the program may or may not use the data. Or, it may only use part of the data.**

• **If there are not enough #include statements, assume the existence of any missing ones.**

• **Different C++ compilers allow different ways of specifying the "main" function. Some prefer it one way, some the other. For the purposes of class examples and tests, any way is acceptable. On a test, do not say that a program does not compile because of the main/return specification. The versions are:**

  **First line:**

  ```cpp
  void main ( ) OR:
  <data type> main ( ) note: Usually, used: int main ( )
  ```

**Close of main function has to be consistent with first line; possibilities are:**

  ```cpp
  return: IF first line was void main ( )
  return 0; IF first line was int main ( )
  ; (semicolon only) For many compilers
  ```
[52] What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
// description: exam question
// ---------------------------
// define display part:
void Display (int dog) {
    int day;
    day = 95;
    cout << day << endl;
    return;
}
// ---------------------------
// Define main part of program:
void main ( ) {
    int day;
    day = 15;
    Display (day);
    cout << day;
}
```

[53] What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
// description: exam question
// -------------------------------------------
// define display part:
void Display (int dog) {
    int day;
    day = 27;
    dog = 95;
    cout << day << endl;
    return;
}
// -------------------------------------------
// Define main part of program:
int main ( ) {
    int day;
    day = 15;
    Display (day);[5]
    cout << day;
}
```
What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
void DisplayDayMonth (int month, int day) {
    cout << endl << "The day is: "
    << day << endl
    << "The month is: "
    << month << endl;
return; }
// -------------------------------------------------
int main ( ) {
    int day;
    int month;
    day = 12;
    month = 5;
    DisplayDayMonth (day,month);
}
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
void DisplayDayMonth (int month, int & day) {
    cout << day << " " << month << endl;
return; }
// -------------------------------------------------
int main ( ) {
    int day;
    int month;
    day = 12;
    month = 5;
    DisplayDayMonth (day,month);
    cout << day << " " << month << endl;
}
```
[56] If there is a single line of data and the carriage return is only hit at the end of it

1 12 31 5 5 6 8 19 0 -1 2 3

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c++
#include <iostream.h>
int main () {
    int sentinelValue = -1;
    int inputValue;
    int evenNumbers;
    int oddNumbers;
    evenNumbers = 0;
    oddNumbers = 0;
    cin >> inputValue;
    while ( inputValue > sentinelValue ) {
        if ( ( inputValue % 2 ) == 0 ) // even
            evenNumbers = evenNumbers + 1;
        if ( ( inputValue % 2 ) == 1 ) // odd
            oddNumbers = oddNumbers + 1;
        cin >> inputValue;
    }
    cout << endl << evenNumbers;
    cout << endl << oddNumbers;
}
```

[57] If there is a single line of data and the carriage return is only hit at the end of it

1 12 31 5 5 6 8 19 0 -1 2 3

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c++
#include <iostream.h>
int main () {
    int firstValue;
    int secondValue;
    int firstValuePlus;
    int plusSecondValue;
    firstValue = 1;
    secondValue = 1;
    firstValuePlus = firstValue ++;
    plusSecondValue = ++ secondValue;
    cout << endl << firstValue << " " << secondValue;
    cout << endl << firstValuePlus << " " << plusSecondValue;
}
```
What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
void CarVolume (int high, int wide, int reach, int& volume) {
    cout << volume << endl;
    volume = high * wide * reach;
    return;
}
```

```
// Define main part of program:
int main ( ) {
    int height;
    int width;
    int length;
    int volume;
    volume = 0;
    height = 7;
    width = 3;
    length = 4;
    cout << endl << height << " " << width << " " << length << endl;
    CarVolume (height, length, width, volume);
    cout << "volume is: " << volume;
}
```

If there is a single line of data:

```
4 3 6 2 1 3
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
int main ( ) {
    int number;
    cin >> number;
    cout << endl;
    cout << number;
    while (number != 0) {
        number = number - 1;
        cout << " " << number;
    }
}
```
If there is a single line of data:

```
4 3 6 2 1 3
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c
#include <iostream.h>
void CarVolume (int high, int wide, int reach, int& volume) {
    cout << volume << endl;
    volume = high * wide * reach;
    return;
}

int main ( ) {
    int height;
    int width;
    int length;
    int volume;
    volume = 100;
    height = 5;
    width = 3;
    length = 4;
    cout << endl << height << " " << width << " " << length << endl;
    CarVolume (height, length, width, volume);
    cout << "volume is: " << volume;
}
```

---

If there is a single line of data:

```
100 200 300
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c
#include <iostream.h>
int main () {
    const int secondsMinute = 60;
    int seconds;
    int minutes;
    int secondsLeft;
    cin >> seconds;
    minutes = seconds / secondsMinute;
    secondsLeft = seconds % secondsMinute;
    cout << endl << seconds << " " << minutes << " " << secondsLeft;
    cout << endl << seconds << " " << secondsLeft;
    return 0;
}
```
If there is a single line of data and the carriage return is only hit at the end of it

3 4 3 6 2 1 3

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
int main ( ) {
    int number;
    cin >> number;
    cout << endl;
    cout << number;
    while (number != 0) {
        number = number - 1;
        cout << " " << number;
    }
}
```

If there is a single line of data:

9 12 24 -1 0 -5

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
int main ()
{
    const int sentinelValue = -1;
    int valueRead;
    int countRead;
    countRead = 0;
    cin >> valueRead;
    while ( valueRead > sentinelValue ) {
        countRead = countRead + 1;
        cin >> valueRead;
        cout << valueRead;
    }
    cout << countRead << " values read in";
}
```
If there is a single line of data and the carriage return is only hit at the end of it

W B A C W b t F A A a W B $ A B b t

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c
#include <iostream.h>
int main () {
    const char sentinelValue = '$';
    char gradeRead;
    int aCount, bCount, cCount, dCount, fCount, wCount;
    aCount = 0; bCount = 0; cCount = 0;
    dCount = 0; fCount = 0; wCount = 0;
    cin >> gradeRead;
    while ( gradeRead != sentinelValue ) {
        switch ( gradeRead ) {
            case 'A': aCount = aCount + 1;
                break;
            case 'B': bCount = bCount + 1;
                break;
            case 'C': cCount = cCount + 1;
                break;
            case 'D': dCount = dCount + 1;
                break;
            case 'F': fCount = fCount + 1;
                break;
            case 'W': wCount = wCount + 1;
                break;
            case ' ': break;
            default: break;
        }
        cin >> gradeRead;
    }
    cout << endl << aCount << " " << bCount << " " << cCount
         << " " << dCount << " " << fCount << " " << wCount; }
```
[65] If there is a single line of data and the carriage return is only hit at the end of it:

89 92 90 86 78 72 64 -1 98 86 -2

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c++
#include <iostream.h>
int main ()
{
    const int sentinelValue = -1;
    int numericGrade;
    int aGrade;
    int bGrade;
    int cGrade;
    int totalGrades;
    aGrade = 0;
    bGrade = 0;
    cGrade = 0;
    totalGrades = 0;
    cin >> numericGrade;
    while ( numericGrade > sentinelValue ) {
        totalGrades = totalGrades + 1;
        if (numericGrade >= 90 )
            aGrade = aGrade + 1;
        if ((numericGrade >= 80 ) and (numericGrade < 90) )
            bGrade = bGrade + 1;
        cin >> numericGrade;
    }
    cout << totalGrades << " " << aGrade << " "
    << bGrade << " " << cGrade;
}
```

[66] If there is a single line of data and the carriage return is only hit at the end of it:

9 12 24 -1 0 -5

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c++
#include <iostream.h>
int main ()
{
    const int sentinelValue = -1;
    int valueRead;
    int countRead;
    countRead = 0;
    cin >> valueRead;
    while ( valueRead > sentinelValue ) {
        countRead = countRead + 1;
        cin >> valueRead;
        cout << valueRead << " ";
    }
    cout << endl << countRead;
}
```
If there is a single line of data:

```
1 12 31 5 5 6 8 19 0 -1 2 3
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
class CubicDimensions {
    public: int height, width, length;
    int CubicVolume ();
};

int CubicDimensions::CubicVolume ( ) {
    int volume;
    volume = height * width * length;
    return volume;
}

int main ( ) {
    CubicDimensions boxCar;
    boxCar.height = 3;
    boxCar.width = 4;
    boxCar.length = 2;
    cout << boxCar.CubicVolume( ) << endl;
    CubicDimensions house;
    house.height = 4;
    house.width = 5;
    house.length = 3;
    cout << house.CubicVolume( );
}
```
If there is a single line of data and the carriage return is only hit at the end of it

```
1 12 31 5 5 6 8 19 0 -1 2 3
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
int Factorial (int start) {
  if (start == 0)
    return 1; // final value returned
  else {
    cout << endl << start;
    return start * Factorial(start-1); }
}
int main () {
  int factorialValue;
  factorialValue = Factorial(4);
  cout << " " << factorialValue;
}
```

If there is a single line of data and the carriage return is only hit at the end of it

```
3 4 7 6 -1 8
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
// test data: 3 4 7 6 -1 8
// --------------------------------------------------------------
#include <iostream.h>
// --------------------------------------------------------------
int Factorial (int start) {
  if (start == 0)
    return 1;
  else {
    cout << " ";
    return start * Factorial (start -1); }
}
// --------------------------------------------------------------
int main () {
  int start;
  cin >> start;
  cout << Factorial (start) << " " << start;
}
```
[70] If there is a single line of data and the carriage return is only hit at the end of it

1 12 31 5 5 6 8 19 0 -1 2 3

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
class TankCar {
    public: int radius, length;
    TankCar () {radius=3.0; length=40.0;}
    int TankCarVolume (); }

int TankCar::TankCarVolume () {
    const double pi = 3.0;  // test approximation
double volume;
    volume = pi* radius * radius * length;
    return volume; }

int main ( ) {
    TankCar oil;
    TankCar milk;
    cout << oil.TankCarVolume () << endl;
    oil.radius = 4.0;
    oil.length = 30;
    cout << oil.TankCarVolume () << endl;
    cout << milk.TankCarVolume ();
}
```

[71] If there is a single line of data and the carriage return is only hit at the end of it

t17b2b24t5-1

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
class TrainCar {
    public:
        char type;
        int age;
};

int main () {
    TrainCar batchCars[100];
    int carCount;
    int outputCounter;
    for (carCount = 1; carCount <= 2; ++carCount)
    {cin >> batchCars[carCount].type;
     cin >> batchCars[carCount].age;}
    for (outputCounter = 1; outputCounter <= 2; ++outputCounter)
    {cout << batchCars[outputCounter].type << " " << batchCars[outputCounter].age << endl;}
}
[72] If there is a single line of data and the carriage return is only hit at the end of it

9 12 24 -1 0 -5

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c++
#include <iostream.h>
void Bump (int& inside, int& outside) {
    int temporaryHold;
    temporaryHold = inside;
    inside = outside;
    outside = temporaryHold;
    return;
}

int main ( ) {
    int inner;
    int outer;
    cin >> inner; cin >> outer;
    cout << inner << ' ' << outer <<endl;
    Bump (inner, outer);
    cout << inner << ' ' << outer <<endl;
}
```

[73] What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```c++
#include <iostream.h>
// description: exam question
// define display part:
void Display (int dog) {
    int day;
    day = 27;
    dog = 95;
    cout << day << endl;
    return;
}

// Define main part of program:
int main ( ) {
    int day;
    day = 15;
    Display (day);
    cout << day;
}
```
What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```
#include <iostream.h>
void DisplayDayMonth (int month, int & day) {
    cout << day << " " << month << endl;
    return; }

int main ( ) {
    int day;
    int month;
    day = 12;
    month = 5;
    DisplayDayMonth (day,month);
    cout << day << " " << month << endl;
}
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```
#include <iostream.h>
void Bump (int& inside,int& outside) {
    int temporaryHold;
    temporaryHold = inside;
    inside = outside;
    outside = temporaryHold;
    return; }

int main ( ) {
    int inner;
    int outer;
    cin >> inner; cin >> outer;
    cout << inner << ' ' << outer <<endl;
    Bump (inner,outer);
    cout << inner << ' ' << outer <<endl;
}
```
[76] What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
void DisplayDayMonth (int month, int & day) {
    cout << day << " " << month << endl;
    day = 12;
    month = 5;
    return; }
// --------------------------------------------
int main ( ) {
    int day;
    int month;
    day = 12;
    month = 5;
    DisplayDayMonth (day,month);
    cout << day << " " << month << endl;
}
```

[77] If there is a single line of data and the carriage return is only hit at the end of it

```
3 4 7 6 -1 8
```
What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
//-------------------------------------
void Read (int valueRead, int & itemCount) {
    if (valueRead > -1) {
        cout << valueRead << " ";
        itemCount = itemCount +1;
        cin >> valueRead;
        Read (valueRead,itemCount); }
}
// ------------------------------------
int main () {
    int start;
    int count;
    count = 0;
    cin >> start;
    Read (start,count);
    cout << start << " " << count;
}
If there is a single line of data and the carriage return is only hit at the end of it:

```
89 92 90 86 78 72 64 -1 98 86 -2
```

What is the output produced by this program? If you think that it does not compile, answer: "Does not compile"; if you think that the results cannot be determined, answer: "Cannot tell for sure"; otherwise, supply the output. (This question is graded right/wrong)

```cpp
#include <iostream.h>
çlass Container
{  
  public: int percentLoaded;
};  
// --------------------------------------------------

class Box:
  
  public Container
  {  
    public: double height, width, length;
    double Volume ();
  };
// --------------------------------------------------

double Box::Volume ()
{  
  double cubes;
  cubes = height * width * length;
  return cubes;
}
// --------------------------------------------------
class RailRoadCar
{  
  public: int yearBuilt;
    int Age ();
};  
// --------------------------------------------------

int RailRoadCar::Age ()
{  
  int currentYear;
  int years;
  currentYear = 2002;
  years = currentYear - yearBuilt;
  cout << years << " " << currentYear;
  return years;
}
// --------------------------------------------------
class BoxCar:
  
  public Box,
  public RailRoadCar
{  
};  
// --------------------------------------------------

int main()
{  
  BoxCar freight;  
  freight.height=10.0;
  freight.width=9.0;
  freight.length=40.0;
  cout << freight.Volume () << endl;
  freight.yearBuilt = 1941;
  cout << endl << freight.Age ();
}
```