The programming approach in which all problems are broken down using only three simple control structures, each of which has only one starting point and one end point, is called ____________________________.

The names of the three structures are:

1. ____________________________________________
2. ____________________________________________
3. ____________________________________________

**Repetition** logical control structures (also known as ________ ) involve a flow of steps in which a decision is made about whether to branch ____________ rather than ______________ to another step.

The three major considerations when designing loops are:

1. ____________________________________________
2. _____________________________________________________________________________
3. ____________________________________________

**Loop Flowchart Terminology:**

The flowchart below shows one structure used for implementing a repetitious flow control. The *italicized notations* within the flowchart's "comment shapes" normally are not written on the diagram, but instead illustrate terminology used by programmers to discuss positions in the structure.

The steps shown on the right are not always written in the order shown. The order depends on the objectives of the process and the decisions made by the analyst about how to best accomplish them.

One complete cycle from the Loopback point back to that point is referred to as a ______________ or ____________________________.

The "Body Steps" are the program's steps that need to be repeated. The "Increment Step" implies any step that relates to the next pass around the loop. These might be to increase or decrease a counter value or to acquire a next value from outside of the program (thru user input or the reading of a data file).
Loop Structures:

Under the structured programming approach, Loops can have only two structures, "leading decision" (a.k.a. "_______-test") and "trailing decision" (a.k.a. "_______-test"). Under structured programming, the testing step in a loop must never be positioned (where)______________________________.

Leading vs. Trailing Structures:

Draw a flowchart below to repeat a process (represented by the shape[ ] ) using the Leading Decision Structure.

Draw a flowchart below to repeat a process (represented by the shape[ ] ) using the Trailing Decision Structure.

Loop Control Methods:

Loop "tests" (branching decisions) are typically based on a relational test of a value in a variable called the __________ variable.

If the contents of that variable was set by the programmer, then the control method is referred to as __________ control (also known as __________ control).

If the contents of that variable was acquired from outside of the program, the control method is referred to as __________ control (also known as __________ control).

If the loop test involves a more complex condition involving both programmer defined and externally obtained values, then the control method is referred to as __________ control.
Counting Controlled Loops:

Draw two structured flowcharts below (one leading decision, one trailing decision) describing a process in which the numbers 0 - 8 are displayed in a column on the screen, skipping a value of 2 each time. A sample of the desired output is shown to the right of this paragraph. The \J symbol represents a "new line" control character (coded with "\n" or "endl" in C++).

The flowcharting symbol for video output looks like \[\text{\textbullet\textbullet}\]. Use a variable named N inside it and remember to draw a box around that identifier to distinguish it as a not being literal text.
The four basic parts of all counting controlled loops are:

1. _______________________________________________
2. _______________________________________________
3. _______________________________________________
4. _______________________________________________

Externally Controlled Loops:

Externally Controlled Loops compare data (such as user input) acquired from outside the program to a known value that is intended to trigger an exit from the loop. This value is referred to as a ________________ value.

In some loops, the test condition is based on a range (e.g. "less than 0") rather than equivalence to a single value. Care must be taken to ensure that values which cause an exit from the loop will not be processed in its body.

Draw two structured flowcharts below (leading decision and trailing decision) describing a process in which a series of weights (each stored in W) are repeatedly requested and added to a sub-total (T). When the user enters a zero, the program should stop asking for weights and then display and identify the total. A sample of the desired output is shown to the right.

Another name for the sub-totaling process is ______________________.

Remember to set the initial value of the variable T to zero before the loop starts. This action is known as ____________________________.
Coding Loops in C++:

Leading Decision Loops required an initialization statement ahead of the other loop statements to set a value first for the control variable. The next statement defining the "entry" to (and exit from) the loop should start with the C++ keyword ___________ followed by the ____________ (in parentheses) and then the loop's body enclosed within __________ (if the body involves more than one statement). If the loop is a counting controlled loop, the body should also include a(n) ____________________________ step.

Fill-in the C++ source code skeleton below with the code for the Leading Decision Counting Loop on Page 3. The boldfaced text below would be required. The C++ comments (enclosed inside of /* */ pairs) act as ________________ (notes to help programmers relate positions in the source code to positions with its related algorithm/flowchart).

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

int main()
{
    /* Declare and initialize the counter */

    /* Test the condition and perform the body and increment steps */
    _____________ ( _____________ )
    {
        /* Body: Output the counter value on the console with a new line */

        /* Increment the counter by 2 */

    }
    /* Terminate the program */
    return 0;
}
```

Enter the source code that you wrote above into a text editor and use a C++ compiler to produce an executable program. Then for practice, try altering some of the program's data by trying different values for the: initial counter value to be displayed, the final counter value to be displayed, and the increment value. Draw the revised flowchart, and then write and test the program code. Finally, try coding the other flowcharts in these pages and alter them to see what happens.