**Repeating Sequences and Loops**

“Code REUSE …

Write a block of code ONE time, and Reuse it MANY times.

This is what good programmers do”

Quote Professor Schwarz

|  |  |  |  |
| --- | --- | --- | --- |
| A program can be a straight **Sequence** of commands. This very simple program would not be very useful when all it could do is just one thing.  (Case 1) | You need programs with flexibility. You need programs that have **Alternative Sequences**  (Case 2) | It was commonly found that some code has **Repeating Sequences** of the same code. The duplicate code is next to each other: **contiguous.** See below  (Case 3) | Eliminate duplicate code in case 3. Write the code **once** and reuse it as **many** time as needed. Use a **LOOP**  (Case 4) |
| C:\Users\Rainbow\Desktop\Sequence Flow chart.png | C:\Users\Rainbow\Desktop\alternative Sequence.pngAlso called *Decision*, *Selection*, *Branching or if.*  *We did this in the previous module 8.* | C:\Users\Rainbow\Desktop\Contiguious Repeating Sequence.png  This situation gives rise for the need to simplify the code by using loops.  **Many** | C:\Users\Rainbow\Desktop\Loop Repeating Sequence 2.png  **One**  Also called *Looping*, *Repetition*, *Iteration* |

**Bad programming practices Alert** (variations on case 3 above):

* If you write a “**block of code**”,
  + and only use it once… Just leave it be. You code is good.
  + or you need to use it a **fixed** number of time: say ten (10) times… Copy and Paste it… you can do it…it’s only 10 times…
  + or you need to use it a **fixed** number of time: thousand (1000) times.. COPY and PASTE a thousand time…. take your time…. not 999, not 1001, be careful ... spend a few hours… get it right…no problem…hah hah..
  + or you need to use it a **variable** number of time… each and every time you run the program… Open up the program, change the source code…CUT and PASTE so you have the correct number this run… Do this every time your program needs to run… Do you think your boss might question what you are doing ?
  + or you need to use it a **variable** number of times… an unknown number of times each time you run your program… OOPS.. not sure how to handle this… good thing I am taking this class ☺

**Let’s be SMART and save time... Let’s learn LOOPs to eliminate the above bad programming practices…..**

**LOOP**

**Loop, Repetition, Iteration**

With **looping**, you avoid having to type or copy a code block multiple times. You only have to write it once.

The Criteria/Conditions determine the number of times the single code block is repeated.

The Mechanics of programming loops. There are ***three*** (3) ‘***Loop Elements’*** important to coding a loop.

* + A *Variable* with start value
  + Criteria/Condition that checks the *variable* to determine when to end the loop or to execute code block again
  + *Variable* modification

**Definition: INFINITE LOOP** – Badly coded loop… Continues looping forever because loop condition/criteria not working correctly.

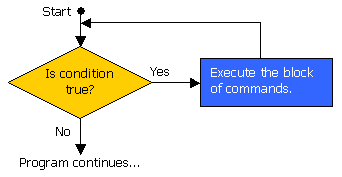
**TYPES OF LOOPS**

There are two types of loops: **Pretest** and **Posttest**.

A pretest simply means the criteria/condition is checked **before** the code block is executed.

A posttest simply means the criteria/condition is checked **after** the code block is executed at least once.

**\*\* FOR LOOP**

C++ Keyword: **for**

The ‘for’ loop is a pretest loop. The condition is at the top of the ‘for’ loop. Pretest means the condition/criteria is checked before the code block is executed. If the condition/criteria is initially FALSE, the block of code is NEVER run.

The ‘For’ loop is easy to use: All ***three*** elements in one place in header

**Basic Syntax:**

**for** ( declare variable = initial **value** ; criteria/condition checking **value** ; change **value** ) { // Condition at top..

body of loop.. Block of code to be repeated

}

Code Example 1 – **Try it**

#include <iostream>

using namespace std;

int main ( ) {

// Example - Print out value of counter during each loop

// declare and initialize loop counter

int count = 20;

// declare i and initialize to 1, check if value of I <= 20, Add 1 to i

for(int i = 1; i <= count ; i++){

// print out value of i each time loop executed..

cout << i << endl;

}

system ("pause"); // Only for PC

return 0;

}

Lets learn two new special symbols: **++** and **--**

i**++** //Means **add 1** to the value of i.

i**--** //Means **subtract 1** from i.

i = i + 1; // **Add 1** to the value of i

i = i – 1; // **Subtract 1** from i.

Code Example 2 – **Try it**

#include <iostream>

using namespace std;

int main ( ) {

// Example – Prompt user for loop count, and print out value at each loop.

int times;

cout << "Enter number of times to repeat (between 1 and 25): ";

cin >> times;

if ((times < 0) || (times > 25))

{

// if times value is out of range

cout << endl << "Next time enter a number between 1 and 25" << endl;

}

else

{

// If times value good, do for loop

// A SIMPLE for loop

for(int i = 1; i <= times; i++){

// print out value of i each time loop executed..

cout << i << endl;

}

}

system ("pause"); // Only for PC

return 0;

}

Code Example 3 – **Try it**

#include <iostream>

using namespace std;

int main ( ) {

int count = 0;

int maxNumber = 0;

int enteredNumber = 0;

cout << "This program determine which entered number is the largest" << endl;

cout << "Enter count of numbers to check: " ;

cin >> count;

// Print out message – input number – compare to see if new number is bigger.. Repeat

for ( int i = 0; i < count ; i++ ) {

cout << "Enter number " << i << " :";

cin >> enteredNumber;

if ( enteredNumber > maxNumber )

{

maxNumber = enteredNumber;

}

}

cout << endl << "Largest Number is " << maxNumber << endl;

system ("pause"); // Only for PC

return 0;

}

**Video Watch Me** – For Loop <http://www.youtube.com/watch?v=b-eYJEYYAsk>

For Loop <http://www.youtube.com/watch?v=4XbNgrGbq5M>

**Nested ‘for’ loop**

You can place any valid statement inside the body of a ‘for’ loop, even another ‘for’ loop.

Code Example 4 – **Try it**

#include <iostream>

using namespace std;

int main ( ) {

// Example – Fun Printing diamonds

int I = 0, j = 0, n = 1;

int max = 0;

int skip = 0;

cout << endl << " Enter the max number of diamonds to print: ";

cin >> max;

cout << endl << " Enter the skip value: ";

cin >> skip;

// Example – for loop nested in a for loop

for( i = 1; i <= max; ++i )

{

for( j = 1; j <= n; ++j )

{

cout << "\*";

}

if( i <= max / skip)

{ n += skip; }

else

{ n -= skip; }

cout << endl;

}

system ("pause"); // Only for PC.

return 0;

}

**Video Watch Me** – **Nested FOR Loop** <http://www.youtube.com/watch?v=iN2DAYlQHQE>

Note: Research online, the three conditions in the for loop. Which do you need, which can you do without ?

**Assignment**

1. Create separate Flow Charts for the above code Examples 1, 2, 3 and 4.
2. Write a program using a ***single*** ‘for’ loop and have a person walk down the stairs.

You only need to make the man go down 5 stairs. It should look like this:

\* o

\* /|\

\* / \

\*\*\*\*\*\*

\* o

\* /|\

\* / \

\*\*\*\*\*\*

\* o

\* /|\

\* / \

\*\*\*\*\*\*

\* o

\* /|\

\* / \

\*\*\*\*\*\*

\* o

\* /|\

\* / \

\*\*\*\*\*\*

Hint: Code the person block first.. Next, in time the for loops, how many spaces to print before person at each step…

1. Write program using two(2) nested ‘for’ loops to print out the values of a Multiplicaction Table 12 By 12. Output should look like Table: 12 columns and 12 rows

Hint: Multiply the first for loop value by the second for loop value…

1. What facts of reality give rise for the need of Loops ?
2. Write a program that uses a for loop with empty( ; ; ) { // that loop 10 times and prints the loop number }

© Bill Schwarz, 2013