

**A - 7: Control structures to make decisions (selections)**

Control structure	Description
Simple if construct	<pre>if ( <i>boolean expression</i> ) {     One or more statements that will be executed if the boolean expression,     given in parentheses above, is true. These statements will not be     executed at all if the boolean expression is false (i.e. not true). }</pre>
if-else construct	<pre>if ( <i>boolean expression</i> ) {     One or more statements that will be executed if the boolean expression,     given in parentheses above, is true. } else {     One or more statements that will be executed if the boolean expression,     given in parentheses above, is false (i.e. not true). }</pre>
if-else if ... construct	<pre>if ( <i>boolean expression 1</i> ) {     One or more statements that will be executed if and only if boolean     expression 1 is true. } else if ( <i>boolean expression 2</i> ) {     One or more statements that will be executed if and only if boolean     expression 2 is true and boolean expression 1 is false. } else {     One or more statements that will be executed if and only if neither     boolean expression 1 nor boolean expression 2 is true. }</pre>
switch-case construct	<pre>switch ( <i>arithmetic expression</i> ) {     case <math>v_1</math>:         Statements which will be executed if the arithmetic expression has         value <math>v_1</math>         break ;     case <math>v_2</math>:         Statements which will be executed if the arithmetic expression has         value <math>v_2</math>         break ;     case <math>v_n</math>:         Statements to be executed when the arithmetic expression has value <math>v_n</math>         break ;     default:         Statements which will be executed if none of the cases matched the         value of the arithmetic expression }</pre>

**A - 8: Control structures to perform repetitions (iterations)**

Control structure	Description
<b>while loop</b>	<pre>while ( <i>boolean expression</i> ) {     One or more internal statements that will be repeatedly executed as long as     the boolean expression, given in parentheses above, is true. }</pre>
<b>for loop</b>	<pre>for ( <i>assignment statement</i> ;       <i>boolean expression</i> ;       <i>increment or decrement statement</i> ) {     One or more internal statements that will be repeatedly executed as long as     the boolean expression given above is true. When the boolean expression     becomes false, the statements that follow this <b>for</b> loop will be executed. }</pre> <p>An index variable may be defined in a <b>for</b> loop in the following way</p> <pre>for ( int <b>some_index</b> = 0 ;       ...</pre> <p>The scope of this kind of variable is within the internal statements of the loop.</p>
<b>do-while loop</b>	<pre>do {     One or more statements that will be first executed once, and then     repeatedly executed as long as the boolean expression, given below in     parentheses, is true. } while ( <i>boolean expression</i> ) ;</pre>

These are sample pages from Kari Laitinen's book  
 "A Natural Introduction to Computer Programming with C++"  
 More information at <http://www.naturalprogramming.com/cppbook.html>