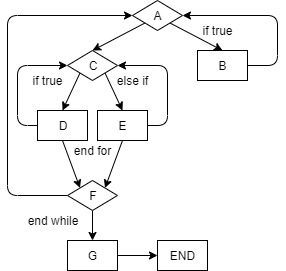
**void** game**()**  
 **{**  
 **typedef** std::**string**::**size\_type** index;  
 std::**string** symbols = "0123456789";  
 **unsigned** **int** **const** selection\_length = **4**;  
 std::**random\_shuffle(**symbols.**begin()**, symbols.**end())**;  
 std::**string** selection = symbols.**substr(0**, selection\_length**)**;  
 std::**string** guess;

**A** **while** **(**std::**cout** << "Your guess? ", std::**getline(**std::**cin**, guess**))**  
 **{**  
**B** **if** **(**guess.**length()** != selection\_length  
 || guess.**find\_first\_not\_of(**symbols**)** != std::**string**::**npos**  
 || contains\_duplicates**(**guess**))**  
 **{**  
 std::**cout** << guess << " is not a valid guess!";  
 **continue**;  
 **}**  
 **unsigned** **int** bulls = **0**;  
 **unsigned** **int** cows = **0**;  
**C** **for** **(**index i = **0**; i != selection\_length; ++i**)**  
 **{**  
 index pos = selection.**find(**guess**[**i**])**;  
**D** **if** **(**pos == i**)**  
 ++bulls;  
**E** **else** **if** **(**pos != std::**string**::**npos)**  
 ++cows;  
 **}**  
**F** std::**cout** << bulls << " bulls, " << cows << " cows.\n";  
**G** **if** **(**bulls == selection\_length**)**  
 **{**  
 std::**cout** << "Congratulations! You have won!\n";  
 **return**;  
 **}**  
 **}**  
 std::**cerr** << "Oops! Something went wrong with input, or you've entered end-of-file!\nExiting ...\n";  
 std::**exit(EXIT\_FAILURE)**;  
 **}**

**Cyclomatic Complexity: 6**

**Statement Coverage:**

* **ABACDFACEFG{End}**

**Branch Coverage:  
(Note: C always loops 4 times, some loops omitted)**

* **ABACDFACEFG{End}**
* **ABACDFG{End}**
* **ACDFG{End}**
* **ACDFACEFG{End}**

**(Note: The following branches are not possible since any time CEF occurs it will loop back to A)**

* **ACEFG{End}**
* **ABACEFG{End}**

Test Cases for Branch Coverage:

Assumptions: Selection is set to “1234”, {Enter} represents pressing Enter key, > represents another string of numbers being entered

Case 1

Input – “12345” {Enter} “1243” {Enter} “1234” {Enter}

Expected Output – “12345 is not a valid guess” “Your guess? “ > “2 bulls, 2 cows” “Your guess? “ > “4 bulls, 0 cows” “Congratulations! You have won!”

Case 2

Input – “12345” {Enter} “1234” {Enter}

Expected Output – “12345 is not a valid guess” “Your guess? “ > “4 bulls, 0 cows” “Congratulations! You have won!”

Case 3

Input – “1234” {Enter}

Expected Output – “4 bulls, 0 cows” “Congratulations! You have won!”

Case 4

Input – “1243” {Enter} “1234” {Enter}

Expected Output – “2 bulls, 2 cows” “Your guess? “ > “4 bulls, 0 cows” “Congratulations! You have won!”