```
 1: // $Id: strvec.cpp,v 1.24 2014-06-20 21:56:35-07 - - $
 2:
 3: //
 4: // NAME
 5: //    strvec - a few random operations on a vector of strings
 6: //
 7:
 8: #include <algorithm>
 9: #include <cstdlib>
10: #include <iostream>
11: #include <string>
12: #include <vector>
13:
14: using namespace std;
15:
16: void print (const string& s) { cout << "for_each: " << s << endl; }
17:
18: int main () {
19:    // Load array into vector.
20:    vector<string> vecstr {"this", "is", "an", "array", "of", "words"};
21:
22:    // Print using indices.
23:    for (size_t index = 0; index < vecstr.size(); ++index) {
24:       cout << "indexing: " << vecstr[index] << endl;
25:    }
26:
27:    // Print using explicit types for iterators.
28:    for (vector<string>::const_iterator itor = vecstr.cbegin();
29:         itor != vecstr.cend(); ++itor) {
30:       cout << "iterator: " << *itor << endl;
31:    }
32:
33:    // Print using auto feature.
34:    for (auto itor = vecstr.cbegin(); itor != vecstr.cend(); ++itor) {
35:       cout << "auto itor: " << *itor << endl;
36:    }
37:
38:    // Print using for-each loop.
39:    for (const auto& str: vecstr) cout << "for colon: " << str << endl;
40:
41:    // Print using <algorithm>for_each and a lambda.
42:    for_each (vecstr.cbegin(), vecstr.cend(),
43:              [](const string& s) {cout << "lambda: " << s << endl;});
44:
45:    // Print using <algorithm>for_each and a function.
46:    for_each (vecstr.cbegin(), vecstr.cend(), print);
47:
48:    return EXIT_SUCCESS;
49: }
50:
51: //TEST// valgrind ./strvec foo bar baz qux quux >strvec.lis 2>&1
52: //TEST// mkpspdf strvec.ps strvec.cpp* strvec.lis
53:
```

```
    1: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting strvec.cpp
    2: strvec.cpp:
    3:       $Id: strvec.cpp,v 1.24 2014-06-20 21:56:35-07 - - $
    4: g++ -g -O0 -Wall -Wextra -rdynamic -std=gnu++11 strvec.cpp -o strvec -lg
lut -lGLU -lGL -lX11 -lrt -lm
    5: rm -f strvec.o
    6: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished strvec.cpp
```

```
    1: ==6117== Memcheck, a memory error detector
    2: ==6117== Copyright (C) 2002-2012, and GNU GPL'd, by Julian Seward et al.
    3: ==6117== Using Valgrind-3.8.1 and LibVEX; rerun with -h for copyright in
fo
    4: ==6117== Command: ./strvec foo bar baz qux quux
    5: ==6117==
    6: indexing: this
    7: indexing: is
    8: indexing: an
    9: indexing: array
   10: indexing: of
   11: indexing: words
   12: iterator: this
   13: iterator: is
   14: iterator: an
   15: iterator: array
   16: iterator: of
   17: iterator: words
   18: auto itor: this
   19: auto itor: is
   20: auto itor: an
   21: auto itor: array
   22: auto itor: of
   23: auto itor: words
   24: for colon: this
   25: for colon: is
   26: for colon: an
   27: for colon: array
   28: for colon: of
   29: for colon: words
   30: lambda: this
   31: lambda: is
   32: lambda: an
   33: lambda: array
   34: lambda: of
   35: lambda: words
   36: for_each: this
   37: for_each: is
   38: for_each: an
   39: for_each: array
   40: for_each: of
   41: for_each: words
   42: ==6117==
   43: ==6117== HEAP SUMMARY:
   44: ==6117==     in use at exit: 0 bytes in 0 blocks
   45: ==6117==   total heap usage: 8 allocs, 8 frees, 227 bytes allocated
   46: ==6117==
   47: ==6117== All heap blocks were freed -- no leaks are possible
   48: ==6117==
   49: ==6117== For counts of detected and suppressed errors, rerun with: -v
   50: ==6117== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 6 from 6)
```