```
// $Id: shared_ptr.cpp,v 1.5 2014-04-10 14:44:24-07 - - $

#include <cstdlib>
#include <iostream>
#include <memory>
#include <string>
#include <vector>

using namespace std;

int main (int argc, char** argv) {
   auto vecstr = make_shared<vector<shared_ptr<string>>>();
   cout << "vector<string*> = " << vecstr << endl;
   for (auto argp = &argv[0]; argp != &argv[argc]; ++argp) {
      vecstr->push_back (make_shared<string> (*argp));
   }
   for (auto itor = vecstr->cbegin(); itor != vecstr->cend(); ++itor) {
      cout << "string* = " << *itor << " -> " << **itor << endl;
   }
   cerr << "We are done." << endl;
   return EXIT_SUCCESS;
}

/*
//TEST// valgrind --leak-check=full --show-reachable=yes \
//TEST//       --log-file=shared_ptr.out.grind \
//TEST//       shared_ptr foo bar baz qux >shared_ptr.out 2>&1
//TEST// mkpspdf shared_ptr.ps shared_ptr.cpp* shared_ptr.out*
*/

```

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting shared_ptr.cpp
shared_ptr.cpp: $Id: shared_ptr.cpp,v 1.5 2014-04-10 14:44:24-07 - - $
g++ -g -O0 -Wall -Wextra -std=gnu++11 shared_ptr.cpp -o shared_ptr -lm
rm -f shared_ptr.o
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished shared_ptr.cpp
```

```
vector<string*> = 0x4c2b058
string* = 0x4c2b0c8 -> shared_ptr
string* = 0x4c2b1e8 -> foo
string* = 0x4c2b308 -> bar
string* = 0x4c2b448 -> baz
string* = 0x4c2b508 -> qux
We are done.
```

```
==31994== Memcheck, a memory error detector
==31994== Copyright (C) 2002-2012, and GNU GPL’d, by Julian Seward et al.
==31994== Using Valgrind-3.8.1 and LibVEX; rerun with -h for copyright info
==31994== Command: shared_ptr foo bar baz qux
==31994== Parent PID: 31993
==31994==
==31994==
==31994== HEAP SUMMARY:
==31994==     in use at exit: 0 bytes in 0 blocks
==31994==   total heap usage: 15 allocs, 15 frees, 595 bytes allocated
==31994==
==31994== All heap blocks were freed -- no leaks are possible
==31994==
==31994== For counts of detected and suppressed errors, rerun with: -v
==31994== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 6 from 6)
```