```cpp
 1: // $Id: arrayptr.cpp,v 1.4 2015-01-07 15:50:18-08 - - $
 2:
 3: //
 4: // Print out addresses of array elements.  Show how to use a
 5: // pointer to iterate over an array instead of an index.  Show
 6: // how to get the index by subtracting pointers.
 7: //
 8: // Note:  + and - have higher precedences than << and >>.
 9: //
10:
11: #include <cstdlib>
12: #include <iomanip>
13: #include <iostream>
14:
15: using namespace std;
16:
17: int main (int argc, char **argv) {
18:    cout << "argv = " << static_cast<void*> (argv) << endl;
19:    for (char** itor = argv; itor != &argv[argc]; ++itor) {
20:       cout << "argv[" << itor - argv << "] = "
21:            << static_cast<void*> (*itor)
22:            << " -> \"" << *itor << "\"" << endl;
23:    }
24:    cout << "main = " << reinterpret_cast<void*> (main) << endl;
25:    return EXIT_SUCCESS;
26: }
27:
28: //TEST// ./arrayptr foo bar baz qux quux >arrayptr.lis 2>&1
29: //TEST// mkpspdf arrayptr.ps arrayptr.cpp* arrayptr.lis
30:
```

```
1: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting arrayptr.cpp
2: arrayptr.cpp:
3:      $Id: arrayptr.cpp,v 1.4 2015-01-07 15:50:18-08 - - $
4: g++ -g -O0 -Wall -Wextra -rdynamic -std=gnu++11 arrayptr.cpp -o arrayptr
-lglut -lGLU -lGL -lX11 -lrt -lm
5: rm -f arrayptr.o
6: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished arrayptr.cpp
```

```
1: argv = 0x7fff7af10268
2: argv[0] = 0x7fff7af10d64 -> "./arrayptr"
3: argv[1] = 0x7fff7af10d6f -> "foo"
4: argv[2] = 0x7fff7af10d73 -> "bar"
5: argv[3] = 0x7fff7af10d77 -> "baz"
6: argv[4] = 0x7fff7af10d7b -> "qux"
7: argv[5] = 0x7fff7af10d7f -> "quux"
8: main = 0x400ac0
```