```
// $Id: slicing.cpp,v 1.9 2015-02-10 17:51:17-08 - - $

//
// Slicing occurs when an object of a a derived class is copied
// into an object of a base class, which loses information.
// Thus, the copy ctor and operator= should be supporessed.
// We use struct here to avoid protection questions.
//

#include <iostream>
#include <typeinfo>

using namespace std;

struct base {
   int base_;
   void print() { // non-virtual
      cout << typeid(*this).name() << ": " << base_ << endl;
   }
};

struct second: public base {
   int second_;
   void print() { // non-virtual
      cout << typeid(*this).name() << ": " << base_
           << ", " << second_ << endl;
   }
};

struct other: public base {
   int other_;
   void print() { // non-virtual
      cout << typeid(*this).name() << ": " << base_
           << ", " << other_ << endl;
   }
};

int main () {
   cout << "sizeof (base) = " << sizeof (base) << endl;
   cout << "sizeof (second) = " << sizeof (second) << endl;
   cout << "sizeof (other) = " << sizeof (other) << endl;
   base b; b.base_ = 6; b.print();
   second s; s.base_ = 8; s.second_ = 10; s.print();
   other o; o.base_ = 15; o.other_ = 20; o.print();

   b = s; b.print();

   s.base_ = 42; s.second_ = 99; s.print();
   // s = b; s.print(); // error: no match for 'operator=' in 's = b'
   static_cast<base>(s) = b; s.print();

   return 0;
}

//TEST// grind="valgrind --leak-check=full --show-reachable=yes"
//TEST// $grind slicing >slicing.out 2>&1
//TEST// mkpspdf slicing.ps slicing.cpp* slicing.out*
```

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting slicing.cpp
slicing.cpp:
     $Id: slicing.cpp,v 1.9 2015-02-10 17:51:17-08 - - $
g++ -g -O0 -Wall -Wextra -rdynamic -std=gnu++11 slicing.cpp -o slicing -lglut -lGLU -lGL -lX11 -lrt -lm
rm -f slicing.o
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished slicing.cpp
```

```
==8097== Memcheck, a memory error detector
==8097== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==8097== Using Valgrind-3.9.0 and LibVEX; rerun with -h for copyright info
==8097== Command: slicing
==8097==
sizeof (base) = 4
sizeof (second) = 8
sizeof (other) = 8
4base: 6
6second: 8, 10
5other: 15, 20
4base: 8
6second: 42, 99
6second: 42, 99
==8097==
==8097== HEAP SUMMARY:
==8097==     in use at exit: 0 bytes in 0 blocks
==8097==   total heap usage: 1 allocs, 1 frees, 9 bytes allocated
==8097==
==8097== All heap blocks were freed -- no leaks are possible
==8097==
==8097== For counts of detected and suppressed errors, rerun with: -v
==8097== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 6 from 6)
```