```
// $Id: stringset.h,v 1.2 2014-10-03 18:47:35-07 - - $

#ifndef __STRINGSET__
#define __STRINGSET__

#include <string>
#include <unordered_set>
using namespace std;

#include <stdio.h>

const string* intern_stringset (const char*);

void dump_stringset (FILE*);

#endif

```

```
// $Id: stringset.cpp,v 1.7 2014-10-09 15:09:33-07 - - $

#include "stringset.h"

using stringset = unordered_set<string>;

stringset set;

const string* intern_stringset (const char* string) {
   pair<stringset::const_iterator,bool> handle = set.insert (string);
   return &*handle.first;
}

void dump_stringset (FILE* out) {
   size_t max_bucket_size = 0;
   for (size_t bucket = 0; bucket < set.bucket_count(); ++bucket) {
      bool need_index = true;
      size_t curr_size = set.bucket_size (bucket);
      if (max_bucket_size < curr_size) max_bucket_size = curr_size;
      for (stringset::const_local_iterator itor = set.cbegin (bucket);
           itor != set.cend (bucket); ++itor) {
         if (need_index) fprintf (out, "stringset[%4lu]: ", bucket);
                    else fprintf (out, "          %4s   ", "");
         need_index = false;
         const string* str = &*itor;
         fprintf (out, "%22lu %p->\"%s\"\n",
                  set.hash_function()(*str),
                  str, str->c_str());
      }
   }
   fprintf (out, "load_factor = %.3f\n", set.load_factor());
   fprintf (out, "bucket_count = %lu\n", set.bucket_count());
   fprintf (out, "max_bucket_size = %lu\n", max_bucket_size);
}

```

```
// $Id: main.cpp,v 1.2 2014-10-09 15:09:33-07 - - $

#include <string>
using namespace std;

#include <assert.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include "stringset.h"

int main (int argc, char **argv) {
   for (int i = 1; i < argc; ++i) {
      const string* str = intern_stringset (argv[i]);
      printf ("intern (\"%s\") returned %p->\"%s\"\n",
              argv[i], str, str->c_str());
   }
   dump_stringset (stdout);
   return EXIT_SUCCESS;
}

```

```
# $Id: Makefile,v 1.8 2014-10-07 17:37:02-07 - - $

GPP   = g++ -g -O0 -Wall -Wextra -std=gnu++11
GRIND = valgrind --leak-check=full --show-reachable=yes

all : teststring

teststring : main.o stringset.o
        ${GPP} main.o stringset.o -o teststring

%.o : %.cpp
        ${GPP} -c $<

ci :
        cid + Makefile stringset.h stringset.cpp main.cpp

spotless : clean
        - rm teststring Listing.ps Listing.pdf test.out test.err

clean :
        -rm stringset.o main.o

test : teststring
        ${GRIND} teststring * * * >test.out 2>test.err

lis : test
        mkpspdf Listing.ps stringset.h stringset.cpp main.cpp \
                Makefile test.out test.err

# Depencencies.
main.o: main.cpp stringset.h
stringset.o: stringset.cpp stringset.h

```

```
intern ("HEADER.html") returned 0x4c2d158->"HEADER.html"
intern ("Listing.pdf") returned 0x4c2d228->"Listing.pdf"
intern ("Listing.ps") returned 0x4c2d2f8->"Listing.ps"
intern ("Makefile") returned 0x4c2d3c8->"Makefile"
intern ("RCS") returned 0x4c2d488->"RCS"
intern ("main.cpp") returned 0x4c2d558->"main.cpp"
intern ("main.o") returned 0x4c2d618->"main.o"
intern ("stringset.cpp") returned 0x4c2d6e8->"stringset.cpp"
intern ("stringset.h") returned 0x4c2d7b8->"stringset.h"
intern ("stringset.o") returned 0x4c2d888->"stringset.o"
intern ("test.err") returned 0x4c2d958->"test.err"
intern ("test.out") returned 0x4c2db28->"test.out"
intern ("teststring") returned 0x4c2dbf8->"teststring"
intern ("HEADER.html") returned 0x4c2d158->"HEADER.html"
intern ("Listing.pdf") returned 0x4c2d228->"Listing.pdf"
intern ("Listing.ps") returned 0x4c2d2f8->"Listing.ps"
intern ("Makefile") returned 0x4c2d3c8->"Makefile"
intern ("RCS") returned 0x4c2d488->"RCS"
intern ("main.cpp") returned 0x4c2d558->"main.cpp"
intern ("main.o") returned 0x4c2d618->"main.o"
intern ("stringset.cpp") returned 0x4c2d6e8->"stringset.cpp"
intern ("stringset.h") returned 0x4c2d7b8->"stringset.h"
intern ("stringset.o") returned 0x4c2d888->"stringset.o"
intern ("test.err") returned 0x4c2d958->"test.err"
intern ("test.out") returned 0x4c2db28->"test.out"
intern ("teststring") returned 0x4c2dbf8->"teststring"
intern ("HEADER.html") returned 0x4c2d158->"HEADER.html"
intern ("Listing.pdf") returned 0x4c2d228->"Listing.pdf"
intern ("Listing.ps") returned 0x4c2d2f8->"Listing.ps"
intern ("Makefile") returned 0x4c2d3c8->"Makefile"
intern ("RCS") returned 0x4c2d488->"RCS"
intern ("main.cpp") returned 0x4c2d558->"main.cpp"
intern ("main.o") returned 0x4c2d618->"main.o"
intern ("stringset.cpp") returned 0x4c2d6e8->"stringset.cpp"
intern ("stringset.h") returned 0x4c2d7b8->"stringset.h"
intern ("stringset.o") returned 0x4c2d888->"stringset.o"
intern ("test.err") returned 0x4c2d958->"test.err"
intern ("test.out") returned 0x4c2db28->"test.out"
intern ("teststring") returned 0x4c2dbf8->"teststring"
stringset[   0]:   14383043818818809721 0x4c2dbf8->"teststring"
                   13646535705723827550 0x4c2d158->"HEADER.html"
stringset[   1]:    2099682443743551108 0x4c2d618->"main.o"
stringset[   4]:     994128771139992428 0x4c2d488->"RCS"
stringset[   6]:   12311697016174216101 0x4c2d6e8->"stringset.cpp"
stringset[  10]:    8902767590177878864 0x4c2d3c8->"Makefile"
stringset[  11]:   12900681736301144686 0x4c2d888->"stringset.o"
stringset[  12]:   11225442421023521097 0x4c2db28->"test.out"
stringset[  15]:    6788790452799346656 0x4c2d228->"Listing.pdf"
stringset[  16]:   17041606903804112922 0x4c2d558->"main.cpp"
stringset[  17]:    9633715851746635993 0x4c2d958->"test.err"
                    1553654622159544091 0x4c2d2f8->"Listing.ps"
stringset[  20]:    1356443961750385397 0x4c2d7b8->"stringset.h"
load_factor = 0.565
bucket_count = 23
max_bucket_size = 2
```

```
==5727== Memcheck, a memory error detector
==5727== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==5727== Using Valgrind-3.9.0 and LibVEX; rerun with -h for copyright info
==5727== Command: teststring HEADER.html Listing.pdf Listing.ps Makefile RCS main.cpp main.o stringset.cpp stringset.h stringset.o test.err test.out teststring HEADER.html Listing.pdf Listing.ps Makefile RCS main.cpp main.o stringset.cpp stringset.h stringset.o test.err test.out teststring HEADER.html Listing.pdf Listing.ps Makefile RCS main.cpp main.o stringset.cpp stringset.h stringset.o test.err test.out teststring
==5727== 
==5727== 
==5727== HEAP SUMMARY:
==5727==     in use at exit: 0 bytes in 0 blocks
==5727==   total heap usage: 54 allocs, 54 frees, 1,913 bytes allocated
==5727== 
==5727== All heap blocks were freed -- no leaks are possible
==5727== 
==5727== For counts of detected and suppressed errors, rerun with: -v
==5727== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 6 from 6)
```