```c
// $Id: hashfn.c,v 1.1 2010-11-04 19:44:16-07 - - $

//
// This program is not part of your project.  It exists just to
// illustrate how to obtain and print hash values.  Each element
// of argv is hashed and printed along with its hashcode.
//

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

#include "../code/strhash.h"

int main (int argc, char **argv) {
   int argi;
   for (argi = 0; argi < argc; ++argi) {
      char *str = argv[argi];
      hashcode_t hashcode = strhash (str);
      xprintf ("%10u = strhash (\"%s\")\n", hashcode, str);
   }
   xprintf ("%10u = 0xFFFFFFFFu\n", 0xFFFFFFFFu);
   return EXIT_SUCCESS;
}

```

```
# $Id: Makefile,v 1.2 2010-11-04 19:48:29-07 - - $

CCOPT     = -Xa -v -g -xO0
LINTOPT   = -Xa -fd -m -u -x -errchk=%all

EXECBINS  = hashfn
HASHSRC   = hashfn.c ../code/strhash.c
LISTFILES = hashfn.c Makefile pspell.perl
LISTING   = ../asg4c-spellchk.misc.ps
CHECK     = cid + ${1}; checksource ${1}
REMOVE    = yes | rm -i ${1} 2>&1 | tr "?" "\12" | sed "s/^ *//"
HASHOUT   = hashfn.out

TESTDATA  = 0 9 A Z a z foo bar baz qux \
            quux quuux quuuux quuuuux quuuuuux quuuuuuux quuuuuuuux

all : ${EXECBINS}

% : %.c
        ${call CHECK, $^}
        lint ${LINTOPT} $^
        cc ${CCOPT} -o $@ $^
        ${call REMOVE, ${notdir ${^:.c=.o}}}

hashfn : ${HASHSRC}

ci : ${LISTFILES}
        ${call CHECK, ${LISTFILES}}

lis : ${LISTFILES} ${HASHOUT}
        mkpspdf ${LISTING} ${LISTFILES} ${HASHOUT}

${HASHOUT} : hashfn
        hashfn ${TESTDATA} * >${HASHOUT}

spotless :
        - ${call REMOVE, ${EXECBINS} ${HASHOUT}}

```

```perl
#!/usr/bin/perl
# $Id: pspell.perl,v 1.2 2010-11-15 13:23:11-08 - - $
use strict;
use warnings;
use Getopt::Std;

$0 =~ s|^(.*/)?([^/]+)/*$|$2|;
my $exit_status = 0;
sub note(@) {print STDERR "$0: @_"}
$SIG{__WARN__} = sub {note @_; $exit_status = 2};
$SIG{__DIE__} = sub {warn @_; exit};
END {exit $exit_status}

my %options;
getopts "nd:", \%options;

my %dictionary;
sub load_dictionary($) {
   my ($dictname) = @_;
   open my $dict, "<$dictname" or do {warn "$dictname: $!\n"; return};
   map {chomp; $dictionary{$_} = 1} <$dict>;
   close $dict;
}
load_dictionary "/usr/share/dict/words" unless $options{'n'};
load_dictionary $options{'d'} if defined $options{'d'};
die "dictionary is empty\n" unless %dictionary;

my $numpat = qr{([[:digit:]]+([-:.][[:digit:]]+)*)};
my $wordpat = qr{([[:alnum:]]+([-&'.][[:alnum:]]+)*)};
for my $filename (@ARGV ? @ARGV : "-") {
   open my $file, "<$filename" or do {warn "$filename: $!\n"; next};
   while (defined (my $line = <$file>)) {
      while ($line =~ s{^.*?($wordpat)}{}) {
         my $word = $1;
         next if $word =~ m{$numpat}
              || $dictionary{$word} || $dictionary{lc $word};
         $exit_status ||= 1;
         print "$filename: $.: $word\n";
      }
   }
   close $file;
}

```

```
3070542422 = strhash ("hashfn")
        48 = strhash ("0")
        57 = strhash ("9")
        65 = strhash ("A")
        90 = strhash ("Z")
        97 = strhash ("a")
       122 = strhash ("z")
    101574 = strhash ("foo")
     97299 = strhash ("bar")
     97307 = strhash ("baz")
    112340 = strhash ("qux")
   3482567 = strhash ("quux")
 107959604 = strhash ("quuux")
3346747751 = strhash ("quuuux")
 669965204 = strhash ("quuuuux")
3589052167 = strhash ("quuuuuux")
3886434804 = strhash ("quuuuuuux")
 220394663 = strhash ("quuuuuuuux")
 105691274 = strhash ("Makefile")
     80962 = strhash ("RCS")
3070542422 = strhash ("hashfn")
 148736715 = strhash ("hashfn.c")
1202077622 = strhash ("hashfn.out")
3338426469 = strhash ("pspell.perl")
4294967295 = 0xFFFFFFFFu
```