```c
// $Id: structopts.c,v 1.1 2012-01-31 18:13:32-08 - - $

//
// NAME
//    structopts - checks and prints out some option values
//
// SYNOPSIS
//    structopts [-cmnstv]
//
// DESCRIPTION
//    Uses getopt(3) to scan options.  Prints out the values of the
//    resulting option structure.
//

#include <libgen.h>
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>

typedef enum {FALSE = 0, TRUE = 1} bool;

struct options {
   bool moretitles;    // -m print file titles à la ‘more’
   bool numberlines;   // -n print line numbers in left column
   bool squeeze;       // -s squeeze multiple blank lines into one
};

//
// Function to print out the options.  Note that the inline struct
// from the main program is passed by reference.
//
void printoptions (struct options *options_ref) {
   printf ("moretitles   = %d\n", options_ref->moretitles  );
   printf ("numberlines  = %d\n", options_ref->numberlines );
   printf ("squeeze      = %d\n", options_ref->squeeze     );
}

```

```c

//
// Main function to analyze options and print them.
//

int main (int argc, char **argv) {
   char *progname = basename (argv[0]);
   int exit_status = EXIT_SUCCESS;
   struct options options;
   int argi;
   memset (&options, 0, sizeof options); // set all opts FALSE

   //
   // Scan arguments and set flags.
   //
   opterr = FALSE;
   for (;;) {
      int option = getopt (argc, argv, "mns");
      if (option == EOF) break;
      switch (option) {
         case ’m’: options.moretitles   = TRUE; break;
         case ’n’: options.numberlines  = TRUE; break;
         case ’s’: options.squeeze      = TRUE; break;
         default : fflush (NULL);
                   fprintf (stderr, "%s: -%c: invalid option\n",
                            progname, optopt);
                   fflush (NULL);
                   exit_status = EXIT_FAILURE;
      };
   };

   //
   // Print options and operands.
   //
   printoptions (&options);

   for (argi = optind; argi < argc; ++argi) {
      printf ("argv[%d] operand = \"%s\"\n", argi, argv[argi]);
   };

   return exit_status;
}

//TEST// runprog -x structopts.lis structopts -m -n foo bar
//TEST// mkpspdf Listing.structopts.ps structopts.c* structopts.lis*

```

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting structopts.c
structopts.c: $Id: structopts.c,v 1.1 2012-01-31 18:13:32-08 - - $
gcc -g -O0 -Wall -Wextra -std=gnu99 structopts.c -o structopts -lm
lint -Xa -fd -m -u -x -errchk=%all structopts.c

function returns value which is always ignored
    fflush              fprintf             printf              memset

rm -f structopts.o
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished structopts.c
```

```

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
log: structopts.log
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

     1  Script   : /afs/cats.ucsc.edu/courses/cmps012b-wm/bin/runprog
     2  limit c :    0 max core file size (KB)
     3  limit f : 4194303 max output file size (KB)
     4  limit t : 4294967295 max CPU time (sec)
     5  stdin   : /dev/null
     6  stdout  : structopts.out
     7  stderr  : structopts.err
     8  log     : structopts.log
     9  listing : structopts.lis
    10  Command : structopts -m -n foo bar
    11  starting: pid 26207: 13:37:21.00
    12  finished: pid 26207: 13:37:21.00, real 0.00, user 0.00, sys 0.00
    13  pstatus: 0x0000 EXIT STATUS = 0

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
stdin: /dev/null
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::


::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
stdout: structopts.out
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

     1  moretitles   = 1
     2  numberlines  = 1
     3  squeeze      = 0
     4  argv[3] operand = "foo"
     5  argv[4] operand = "bar"

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
stderr: structopts.err
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

```