getopt() in C Programming Cheat Sheet

To parse command-line short options in C, use `getopt()` with `argc` and `argv`.

```
#include <unistd.h>
       int getopt(int argc, char * const *argv, const char *optstring);
```

Global variables

`extern int opterr`	If `0`, turn off error messages (default is non-zero)
`extern int optopt`	An unrecognized option (if `getopt()` returns ?)
`extern char *optarg`	Arg to option (if option letter in `optstring` is followed by `:`)
`extern int optind`	The next element in `argv` after `getopt()` is done

`getopt()` returns the letter from `optstring` if found, `?` if unrecognized, or `-1` if no options left.

```
#include <stdio.h>
#include <unistd.h>

int main(int argc, char **argv) {
  int option, index;
  opterr = 0; /* turn off error messages */

  while ((option = getopt(argc, argv, "ho:")) != -1) {
    switch (option) {
    case 'h':                          /* help */
      printf("usage: %s [-h] [-o arg]\n", argv[0]);
      break;
    case 'o':                          /* option */
      printf("optarg is %s\n", optarg);
      break;
    default:                           /* ? */
      printf("option not recognized: %c\n", optopt);
    }
  }

  for (index = optind; index < argc; index++) {
    puts(argv[index]);
  }

  return 0;
}
```

Sample output

```
$ foo -x -h -o hello one two
option not recognized: x
usage: foo [-h] [-o arg]
optarg is hello
one
two
```

opensource.com

getopt() in C Programming Cheat Sheet

To parse command-line long options in C, use `getopt_long()` with `argc` and `argv`.

```
#include <getopt.h>
       int getopt_long(int argc, char * const *argv, const char *optstring,
              const struct option *longopts, int *longindex);

         struct option {
            const char *name;
            int         has_arg;
            int        *flag;
            int         val;    };
```

Global variables	
`longopts.name`	The name of the long option (such as `"help"` or `"option"`)
`longopts.has_arg`	If `0`, no `optarg`. If `1`, requires an `optarg`. If `2`, optional `optarg`
`longopts.flag`	Returns `val` or `0`
`longopts.val`	The value returned (such as `'h'` or `'o'`)
`longindex`	Points to a variable to store the index of the long option in `longopts`

End the `longopts` array with `{0,0,0,0}`

```
#include <stdio.h>
#include <getopt.h>

int main(int argc, char **argv) {
 int option, index;
 static struct option long_options[] = {
  { "help", 0, NULL, 'h' },
  { "option", 1, NULL, 'o' },
  { 0, 0, 0, 0 } };

  opterr = 0;
  while ((option = getopt_long(argc,
argv, "ho:", long_options, NULL)) != -1){
    switch (option) {
    case 'h':   /* help */
  printf("usage: %s [-h] [-o arg]\n",
argv[0]);
  break;
case 'o':   /* option */
  printf("optarg is %s\n", optarg);
  break;
default:  /* ? */
  printf("option not recognized: %c\n",
optopt); }}

for (index = optind;
index < argc; index++) {
  puts(argv[index]);
}
  return 0;
}
```