【Linux Programming】Day17

Tags	
Date	@June 9, 2022
Summary	Program Arguments and getopt

【Ch4】The Linux Environment

This chapter considers the environment in which programs operate, how they can use that environment to gain information about operating conditions.

4.1 Program Arguments

When writing a C program, it is advisable to add longer, more meaningful switch names as alternatives to the single character versions and to use a double dash to distinguish them. So we might have `-h` and `--help` as options to get help.

Another foible is to make the option `+x`(for example), perform the opposite function to `-x`.

4.1.1 getopt

Linux provides the `getopt` facility, which supports the use of options with and without values and is simple to use.

```
#include <unistd.h>

int getopt(int argc,  char *const argv[], const char *optstring);
extern char *optarg;
extern int optind, opterr, optopt;
```

The `getopt` function takes the `argc` and `argv` parameters as passed to the program's main function and an options specifier string that tells `getopt` what options are defined for the program and whether they have associated values.

The `optstring` is simply a list of characters, each representing a single character option. If a character is followed by a colon, it indicates that the option has an associated value that will be taken as the next argument.

For example, the following call would be used to handle the preceding example:

```
getopt(argc, argv, "if:lr");
```

It allows for simple options `-i`, `-l`, `-r`, and `-f`, followed by a filename argument.

The return result for `getopt` is the next option character found in the `argv` array(if there is one). Call `getopt` repeatedly to get each option in turn. It has the following behaviour:

- If the option takes a value, that value is pointed to by the external variable `optarg`.
- `getopt` returns -1 when there are no more options to process. A special argument, `--`, will cause `getopt` to stop scanning for options.
- `getopt` returns `?` if there is an unrecognized option, which it stores in the external variable `optopt`.
- If an option requires a value and no value is given, `getopt` normally returns `?`. By placing a colon as the first character of the options string, `getopt` returns `:` instead of `?` when no value is given.

The external variable, `optind`, is set to the index of the next argument to process. `getopt` uses it to remember how far it's got.

In most Linux systems, `getopt` rewrites the `argv` array so that all of the non-options arguments are presented together, starting at `argv[optind]`.