

Задание 2 по инфе.

15 сентября 2022 г.

getopt

Аргументы: `argc`; `argv`; строка коротких опций (например, в случае `ls` третьим аргументом будет `"alR"`); массив длинных опций (`const struct option long_options[] = { {"-recursive", no_argument, NULL, 'R'}, {аналогично следующая опция}, ..., {NULL, 0, NULL, 0} }`); `&option_index` (вводим где-нить в начале переменную `int option_index = 0`).

Возвращает `getopt_long` символ, соответствующий флагу, который разыскала: `'R'`, например. В ее описании говорится, что возвращает `int`, но забудьте про это=). Возвращает `-1`, если больше не может найти флагов в `argv`. То есть нужен цикл, пока `getopt_long` не вернет `-1`.

Также есть переменная `extern int optind`. Она "внешняя", то есть уже где-то объявлена, а значит в нашей проге мы можем ей пользоваться без объявления. Это не `option_index`. После того, как `getopt` отработала (не только `getopt_long`), все, что было в `argv` помимо флагов, переносится в конец `argv`, а флаги-в начало. Не флаги в обновленной `argv` начинаются с `argv[optind]`.

ls. 1 балл

Задание: реализовать `ls` на языке си. Реализовать флаги `-a`, `-l`, `-R`, а также их "длинные" аналоги, сделать это с помощью `getopt_long`.

Не забыть, что `ls` может подаваться в качестве аргумента директория. Например, если мы находимся в терминале в папке `home` (в которой находится папка `prog`) и ждем `ls prog`, то нам выдастся содержимое папки `prog`.

`opendir` открывает директорию и возвращает указатель на "directory stream", то есть поток файлов в этой директории. Если `ls` без аргументов (без имени директории), то для нашей программы имя текущей директории: `"."`.

потом `readdir(DIR*)` возвращает указатель на структуру `struct dirent`, соответствующую первому файлу/директории в "directory stream" данной директории

и мы его вызываем от одной и той же директории, пока не вернется значение `NULL`

предположим, у нас есть флаг `-R`

в этом случае, надо проверить для каждой `struct dirent`, является ли оно регулярным файлом или директорией

если директория, то запоминаем и потом по ней тоже проходимся

узнать, является ли директорией или регулярным файлом, можно с помощью `stat`. Также `stat` пригодится для реализации флага `-l`.