



МИНИСТЕРСТВО НАУКИ
И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Новосибирский государственный технический университет»



**НГТУ
НЭТИ** | **Факультет прикладной
математики и информатики**

Кафедра теоретической и прикладной информатики

Лабораторная работа №1
по дисциплине «Управление ресурсами в вычислительных системах»

**Программные средства для управления системными ресурсами.
Файловая система ОС UNIX**

Бригада 9 БЕГИЧЕВ АЛЕКСАНДР

Группа ПМ-92 ШИШКИН НИКИТА

Вариант 9

Преподаватели СИВАК МАРИЯ АЛЕКСЕЕВНА

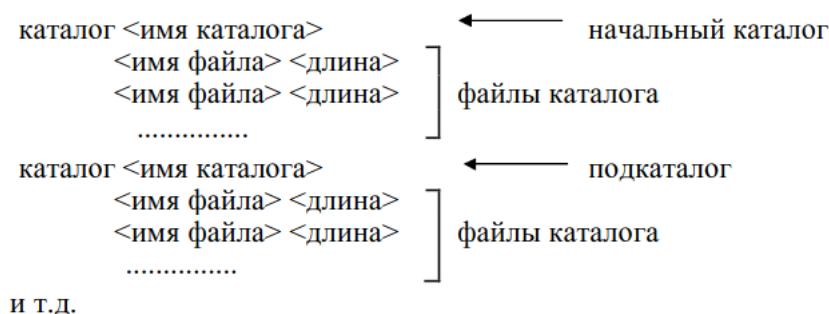
СТАСЫШИН ВЛАДИМИР МИХАЙЛОВИЧ

Новосибирск, 2022

Цель работы

Ознакомиться с устройством файловой системы ОС UNIX, механизмами ее функционирования, программными средствами для работы с ней (командный язык Shell, язык Си).

Вариант: Программа подсчитывает количество и выводит список всех файлов (без каталогов) в алфавитном порядке в поддереве, начиная с каталога, имя которого задано параметром программы. Форма вывода результата:



Shell-программа

Описание алгоритма

```
1: start_dir ← start_directory_path;
2: while symbol ≠ '\0' and type dir do
3:   read directory;
4:   print → folder_path;
5:   count ← 0;
6:   while symbol ≠ '\0' and type file and depth 1 do
7:     read directory;
8:     print → file_name;
9:     count ← count + 1;
10:  end while
11:  print → count;
12: end while
```

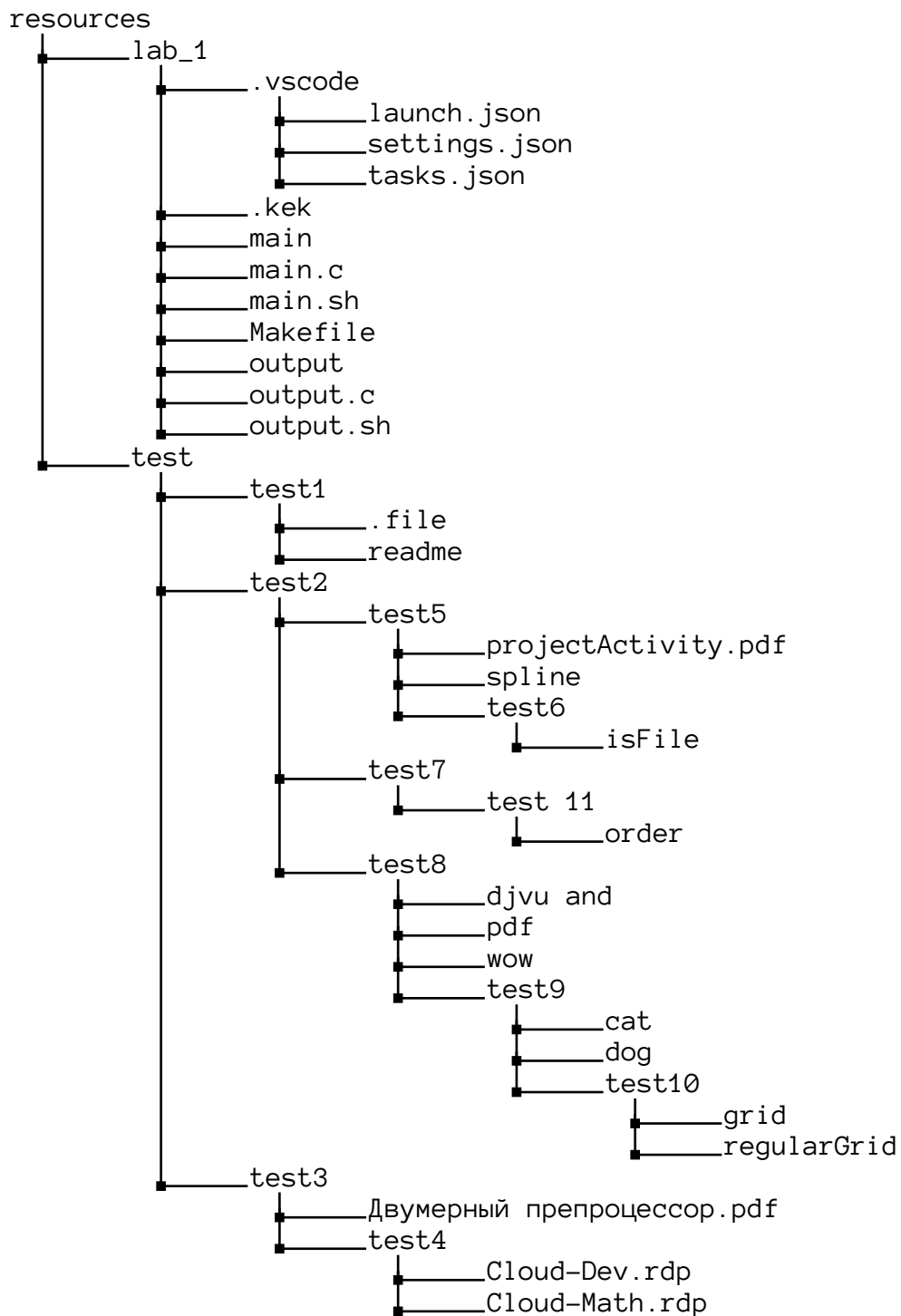
Используемые команды и опции

- `type -d` – проверка на то, что файл является каталогом.
- `type -f` – проверка на то, что файл является обычным.
- `-print0` – задает разделителем конец строки.
- `sort -z` – сортировка файлов в алфавитном порядке.
- `read -d $'\0'` – считывание до символа конца строки.
- `find -maxdepth 1` – команда для поиска файлов в каталоге, опция задает глубину поиска.

- `wc -l` – команда, которая выводит статическую информацию, из-за опции рассчитывает только количество строк в каталоге.
- `echo -e` – команда для вывода текста в терминал, опция используется для отображения цвета текста.

Тестирование

Дерево директории:



Вывод программы:

```
/home/hukutka/repos/resources
* There is no any file

/home/hukutka/repos/resources/lab_1
> .kek 0B
> main 16616B
> main.c 4272B
> main.sh 1626B
> Makefile 31B
> output 16648B
> output.c 2768B
> output.sh 817B
* There are 8 files total

/home/hukutka/repos/resources/lab_1/test
* There is no any file

/home/hukutka/repos/resources/lab_1/test/test1
> .file 13B
> readme 14B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test2
* There is no any file

/home/hukutka/repos/resources/lab_1/test/test2/test5
> projectActivity.pdf 1771237B
> spline.pdf 1125750B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test2/test5/test6
> isFile 0B
* There are 1 files total

/home/hukutka/repos/resources/lab_1/test/test2/test7
* There is no any file

/home/hukutka/repos/resources/lab_1/test/test2/test7/test 11
> order 0B
* There are 1 files total

/home/hukutka/repos/resources/lab_1/test/test2/test8
> djvu and 0B
> pdf 0B
> wow 0B
* There are 3 files total

/home/hukutka/repos/resources/lab_1/test/test2/test8/test9
> cat 0B
> dog 0B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test2/test8/test9/test10
> grid 0B
> regularGrid 0B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test3
> Двумерный препроцессор.pdf 1288771B
* There are 1 files total

/home/hukutka/repos/resources/lab_1/test/test3/test4
> Cloud-Dev.rdp 9389B
> Cloud-Math.rdp 9390B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/.vscode
> launch.json 1229B
> settings.json 267B
> tasks.json 713B
* There are 3 files total
```

Листинг

```
1  #!/bin/sh
2
3  # Group: ПМ-92
4  # Task version: 8
5  # Students: Begichev, Shishkin
6
7  # Started directory for program
8  start_dir=$root
9
10 # Is argument given?
11 if [ $# -gt 1 ]; then
12     echo -e "Usage: $0 [directory_name]"
13     exit
14
15 # And is this a directory?
16 elif [ ! -d $1 ]; then
17     echo "There is no such directory: $1"
18     exit
19
20 # Change started directory
21 # to given argument
22 else
23     start_dir=$1
24 fi
25
26 # Just colors
27 RED='\033[0;31m'
```

```

28 YELLOW='\033[0;33m'
29 YELLOW_BOLD='\033[1;33m'
30 NC='\033[0m'
31
32 # The -print0 will use the NULL as a file separator
33 # instead of a newline, sort -z and the -d '$'\0' will use NULL
34 # as the separator while reading.
35 find $start_dir -type d -print0 | sort -z | while read -d '$'\0' dir; do
36     echo -e "${YELLOW_BOLD}"$dir"${NC}"
37
38     # Iters are used to count
39     # files in the directory
40     iters=0
41
42     # If directory is empty, print info and go next
43     if [ $(find "$dir" -maxdepth 1 -type f | wc -l) -eq 0 ]; then
44         echo " * There is no any file"
45
46     # Else iterate files over directory
47     else
48         # pipe drops variables after running
49         # but with this short is don't
50         shopt -s lastpipe
51         find "$dir" -maxdepth 1 -type f -print0 | sort -z | while read -d '$'\0' file;
52         ↪ do
53             # Print filename and used amount of memory in bytes
54             echo -e " >${YELLOW} ${file###*/} ${RED} $(du -b "$file" | cut -f1)B ${NC}"
55
56             # Count file in the directory
57             ((iters++))
58
59         done
60
61         # Print the number of files in the directory
62         echo " * There are $iters files total"
63     fi
64
65     # Output separator (for directories)
66     echo ' '
67 done

```

Программа на языке СИ

Описание алгоритма

```
1: start_dir ← start_directory_path;
2: dir_num ← 0;
3: file_num ← 0;
4: stbuf;
5: dir ← opendir(start_dir);
6: while ((item = readdir(dir)) ≠ NULL) do
7:   stat(start_dir, stbuf);
8:   if (S_ISREG(stbuf.flag)) then
9:     files_names ← item->file_name;
10:    file_num ← file_num + 1;
11:   else if (S_ISDIR(stbuf.flag)) then
12:     dirs_names ← dir_name;
13:     dir_num ← dir_num + 1;
14:   end if
15: end while
16: closedir(dir);
17: print → dir_name;
18: qsort(files_names);
19: print → files_names;
20: print → file_num;
21: qsort(dirs_names);
22: repeat algorithm (from step 3) dir_num with start_dir ← dir_name;
```

▷ Structure variable

▷ Get information about the file

▷ We can get from the starting directory

▷ Update the structure variable and output the file size as well

Используемые системные функции

- `stat()` – вызов помещает информацию о файле в структурную переменную.
- `opendir()` – вызов выполняет открытие каталога.
- `closedir()` – вызов выполняет закрытие каталога.
- `readdir()` – вызов выполняет чтение записей каталога.
- `S_ISREG()` – макрос, который через флаг структурной переменной определяет является ли файл обычным.
- `S_ISDIR()` – макрос, который через флаг структурной переменной определяет является ли файл каталогом.

Тестирование

Дерево директории возьмем с тестирования shell-программы.

Вывод программы:

```
/home/hukutka/repos/resources
* There is no any file

/home/hukutka/repos/resources/lab_1
> .kek 0B
> Makefile 31B
> main 16776B
> main.c 4272B
> main.sh 1626B
> output 16648B
> output.c 2768B
> output.sh 817B
* There are 8 files total

/home/hukutka/repos/resources/lab_1/.vscode
> launch.json 1229B
> settings.json 267B
> tasks.json 713B
* There are 3 files total

/home/hukutka/repos/resources/lab_1/test
* There is no any file

/home/hukutka/repos/resources/lab_1/test/test1
> .file 13B
> readme 14B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test2
* There is no any file

/home/hukutka/repos/resources/lab_1/test/test2/test5
> projectActivity.pdf 1771237B
> spline.pdf 1125750B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test2/test5/test6
> isFile 0B
* There are 1 files total

/home/hukutka/repos/resources/lab_1/test/test2/test7
* There is no any file

/home/hukutka/repos/resources/lab_1/test/test2/test7/test 11
> order 0B
* There are 1 files total

/home/hukutka/repos/resources/lab_1/test/test2/test8
> djvu and 0B
> pdf 0B
> wow 0B
* There are 3 files total

/home/hukutka/repos/resources/lab_1/test/test2/test8/test9
> cat 0B
> dog 0B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test2/test8/test9/test10
> grid 0B
> regularGrid 0B
* There are 2 files total

/home/hukutka/repos/resources/lab_1/test/test3
> Двумерный препроцессор.pdf 1288771B
* There are 1 files total

/home/hukutka/repos/resources/lab_1/test/test3/test4
> Cloud-Dev.rdp 9389B
> Cloud-Math.rdp 9390B
* There are 2 files total
```

Листинг

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <sys/stat.h>
4  #include <dirent.h>
5  #include <stdlib.h>
6
7  // Console colors
8  #define RED "\033[0;31m"
9  #define YELLOW "\033[0;33m"
10 #define YELLOW_BOLD "\033[1;33m"
11 #define NC "\033[0m"
12
13 // Group: ПМ-92
14 // Task version: 8
15 // Students: Begichev, Shishkin
16
17 // From Linux man page
18 int cmpstringp(const void* p1, const void* p2) {
19     return strcmp(*(const char **) p1, *(const char **) p2);
20 }
21
22 int show_dir(const char *dirname) {
23     DIR *dir;
```

```

24 struct stat stbuf;
25 struct dirent *item;
26 unsigned int iters = 0;
27
28 unsigned int max_file_count = 1;
29 unsigned int max_dir_count = 1;
30 char **file_names = (char **) malloc(max_file_count * sizeof(char *));
31 char **dir_names = (char **) malloc(max_dir_count * sizeof(char *));
32
33
34 // dir name length
35 size_t dirname_len = strlen(dirname);
36
37 // curent path
38 char path[PATH_MAX];
39 strcpy(path, dirname);
40 strcat(path, "/");
41
42 // Count files and dirs
43 unsigned int file_count = 0;
44 unsigned int dir_count = 0;
45
46 // Open directory
47 dir = opendir(dirname);
48 if (dir == NULL) {
49     printf("There is no such directory: %s\n", dirname);
50     return 1;
51 }
52
53 // Count files and dirs and save their names
54 while ((item = readdir(dir)) != NULL) {
55     strcat(path, item->d_name); // Получаем полное имя файла
56     stat(path, &stbuf); // Получаем информацию о файле
57
58     // File
59     if (S_ISREG(stbuf.st_mode)) {
60         if (file_count == max_file_count) {
61             file_names = (char **) realloc(file_names, 2 * max_file_count *
62                 ↳ sizeof(char *));
63             max_file_count *= 2;
64         }
65
66         // Push file name and its size
67         file_names[file_count] = (char *) malloc(strlen(item->d_name) *
68             ↳ sizeof(char *));
69         strcpy(file_names[file_count], item->d_name);
70         file_count++;
71
72         // Directory
73     } else if (S_ISDIR(stbuf.st_mode)) {
74         if (strcmp(item->d_name, ".") != 0 && strcmp(item->d_name, "..") != 0) {
75             if (dir_count == max_dir_count) {
76                 dir_names = (char **) realloc(dir_names, 2 * max_dir_count *
77                     ↳ sizeof(char *));
78                 max_dir_count *= 2;
79             }
80
81             // Push directory name
82             dir_names[dir_count] = (char *) malloc(strlen(path) * sizeof(char *));
83             strcpy(dir_names[dir_count], path);

```



```

81         dir_count++;
82     }
83 }
84
85     // Drop symbols after the last '/'
86     path[dirname_len+1] = '\0';
87 }
88
89 // Close directory
90 closedir(dir);
91
92 printf("%s%s%s\n", YELLOW_BOLD, dirname, NC);
93
94 // Sort files
95 qsort(file_names, file_count, sizeof(char *), cmpstringp);
96
97 // Iterate over files
98 for (unsigned int i = 0; i < file_count; i++) {
99     strcat(path, file_names[i]); // Get full file name
100    stat(path, &stbuf); // Get file info
101
102    printf(" > %s%s %s%lldB %s\n", YELLOW, file_names[i], RED, stbuf.st_size,
103           ↪ NC);
104    free(file_names[i]);
105
106    path[dirname_len+1] = '\0';
107 }
108
109 free(file_names);
110
111 if (file_count > 0) {
112     printf(" * There are %d files total\n\n", file_count);
113 } else {
114     printf(" * There is no any file\n\n");
115 }
116
117 // Sort directories
118 qsort(dir_names, dir_count, sizeof(char *), cmpstringp);
119
120 // Iterate over directories
121 for (unsigned int i = 0; i < dir_count; i++) {
122     show_dir(dir_names[i]);
123     free(dir_names[i]);
124 }
125
126 free(dir_names);
127
128 return 0;
129 }
130
131 int main(int argc, char *argv[]) {
132     char dirname[PATH_MAX];
133     struct stat stbuf;
134
135     // Check input arguments
136     switch (argc) {
137         case 1:
138             dirname[0] = '.';
139             dirname[1] = '\0';
140             break;

```

```

140
141     case 2:
142         memcpy(dirname, argv[1], strlen(argv[1]));
143         dirname[strlen(argv[1])] = '\0';
144         break;
145
146     default:
147         printf("Usage: %s [directory_name]\n", argv[0]);
148         return 1;
149 }
150
151 // Remove unnecesary '/' in the end of
152 // the directory name
153 if (dirname[strlen(dirname)-1] == '/') {
154     dirname[strlen(dirname)-1] = '\0';
155 }
156
157 show_dir(dirname);
158
159 return 0;
160 }

```

Вывод

В ходе выполнения лабораторной работы мы ознакомились с устройством файловой системы ОС UNIX, механизмами ее функционирования, программными средствами для работы с ней (командный язык Shell, язык Си).