



OS Lab 28

Status	approved
checkbox	<input checked="" type="checkbox"/>
class	OS
due date	@Apr 14, 2021

Task

28. Генератор случайных чисел

Напишите программу, которая генерирует
сортированный список из ста случайных
чисел в диапазоне от 0 до 99.
Распечатайте числа по десять в строке.
Используйте `p2open(3)`, чтобы запустить
`sort(1)` и `rand(3)` и `srand(3)` для генерации
случайных чисел.

```
d.khaetskaya@fit-main: ~/lab28
File Edit View Search Terminal Help
Generated array:
77 91 84 29 17 0 7 39 35 70
29 44 55 89 22 38 35 94 16 86
15 26 87 8 5 10 94 28 72 99
91 24 19 71 75 99 34 22 56 17
80 14 75 71 83 89 44 71 51 84
77 28 9 8 22 7 71 96 18 59
5 1 6 86 62 53 18 71 96 46
47 92 97 23 39 91 14 48 11 20
0 1 2 85 37 60 35 0 65 59
84 66 72 28 18 88 96 79 31 99

Sorted array:
0 0 0 1 1 2 5 5 6 7
7 8 8 9 10 11 14 14 15 16
17 17 18 18 18 19 20 22 22 22
23 24 26 28 28 28 29 29 31 34
35 35 35 37 38 39 39 44 44 46
47 48 51 53 55 56 59 59 60 62
65 66 70 71 71 71 71 71 72 72
75 75 77 77 79 80 83 84 84 84
85 86 86 87 88 89 89 91 91 91
92 94 94 96 96 96 97 99 99 99
d.khaetskaya@fit-main:~/lab28$
```

The `rand()` function uses a multiplicative congruential random-number generator with period 2^{32} that returns successive pseudo-random numbers in the range of 0 to `RAND_MAX` (defined in `<stdlib.h>`).

The `srand()` function uses the argument `seed` as a seed for a new sequence of pseudo-random numbers to be returned by subsequent calls to `rand()`. If `srand()` is then called with the same seed value, the sequence of pseudo-random numbers will be repeated. If `rand()` is called before any calls to `srand()` have been made, the same sequence will be generated as when `srand()` is first called with a seed value of 1.

The `p2open()` function forks and execs a shell running the command line pointed to by `cmd`. On return, `fp[0]` points to a FILE pointer to write the command's standard input and `fp[1]` points to a FILE pointer to read from the command's standard output. In this way the program has control over the input and output of the command.

The function returns 0 if successful; otherwise, it returns -1.

The `p2close()` function is used to close the file pointers that `p2open()` opened. It waits for the process to terminate and returns the process status. It returns 0 if successful; otherwise, it returns -1.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <libgen.h>

int main() {
    srand(time(NULL));
    FILE * fd[2];
    if( p2open("sort", fd) == -1){
        perror("p2open failed");
        return -1;
    }

    int currentNum;

    printf("Generated array:\n");
    for(int i = 0; i < 10; i++){
        for(int j = 0; j < 10; j++){
            currentNum = rand() % 100;
            fprintf(fd[0], "%d\n", currentNum);
            printf("%3d ", currentNum);
        }
        printf("\n");
    }
    if (pclose(fd[0]) == -1){
        perror("pclose failed");
        return -1;
    }

    printf("\n");

    printf("Sorted array:\n");
    for(int i = 0; i < 10; i++){
        for(int j = 0; j < 10; j++){
            currentNum = rand() % 100;
            fscanf(fd[1], "%d", &currentNum);
            printf("%3d ", currentNum);
        }
        printf("\n");
    }
}
```

<http://src.illumos.org/source/xref/illumos-gate/usr/src/lib/libc/port/stdio/popen.c?r=462453d2#89>

```
if (pclose(fd[1]) == -1){
    perror("pclose failed");
    return -1;
}
return 0;
}
```

```
58
59     typedef struct node {
60         pid_t    pid;
61         int      fd;
62         struct   node    *next;
63     } node_t;
64
```