

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”
КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ УПРАВЛІННЯ

Звіт з лабораторної роботи № 6
з дисципліни “Основи архітектури ЕОМ та операційні системи”

Виконав: ст. гр. КН-221В

Шулюпов Єгор Русланович

Перевірив: Асистент каф. ПІІТУ

Дмитро Миколайович Ковальчук

Харків

2022

Linux API – Введення у міжпроцесну взаємодію. Канали.

Мета: Знайомство з каналами та міжпроцесною взаємодією у Linux. Навчитися створювати та працювати з каналами. Навчитися розбивати розв’язання складного завдання на кілька простих операцій, кожна з яких довіряється окремій невеликій програмі.

Завдання роботи

Скомпілювати та перевірити працездатність програм наведених у методичних матеріалах. Скомпілювати програми `typeserver.c` та `typesclient.c` в одній директорії. Запустіть спочатку сервер, потім клієнт у різних вікнах терміналу. Друкуйте текст у вікні сервера. Після кожного натискання [Enter] клієнт повинен відображати рядок, надрукований на сервері.

Нижче наведено код файлу `semclient.c`:

```
1 #include <stdio.h>
2 #include <sys/unistd.h>
3 #define FIFO_NAME "./fifofile"
4 int main()
5 {
6     FILE* f;
7     char ch;
8     f = fopen(FIFO_NAME, "r");
9     while (ch != 'q')
10     {
11         ch = fgetc(f);
12         putchar(ch);
13     }
14     fclose(f);
15     unlink(FIFO_NAME);
16     return 0;
17 }
```

Нижче наведено код файлу `semserv.c`:

```
1 #include <stdio.h>
2 #include <sys/types.h>
3 #include <sys/stat.h>
4 #include <sys/unistd.h>
5 #define FIFO_NAME "./fifofile"
6 int main(int argc, char** argv)
7 {
8     FILE *f;
9     char ch;
10    mkfifo(FIFO_NAME, 0600);
11    f = fopen(FIFO_NAME, "w");
12    if(f == NULL)
13    {
14        printf("Cannot open this file\n");
15        return -1;
16    }
17    while ((ch != 'q'))
18    {
19        ch = getchar();
20        fputc(ch, f);
21        if(ch==10) fflush(f);
22    }
23    fclose(f);
24    unlink(FIFO_NAME);
25    return 0;
26 }
27 |
```

Нижче наведено код файлу pipe.c:

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <sys/types.h>
4  int main(int argc, char** argv)
5  {
6      int pipedes[2];
7      pid_t pid;
8      pipe(pipedes);
9      pid = fork();
10     if (pid > 0)
11     {
12         char* str = "Some string that was output\n";
13         close(pipedes[0]);
14         write(pipedes[1], (void*)str, strlen(str) + 1);
15         close(pipedes[1]);
16     }
17     else
18     {
19         char buf[1024];
20         int len;
21         close(pipedes[1]);
22         while ((len = read(pipedes[0], buf, 1024)) != 0) write(2, buf, len);
23         close(pipedes[0]);
24     }
25     return 0;
26 }
```

Нижче наведено код файлу makelog.c:

```
1  #include <stdio.h>
2  #include <errno.h>
3  #define BUF_SIZE 0x100
4  int main(int argc, char** argv)
5  {
6      FILE* f;
7      FILE* o;
8      int len;
9      char buf[BUF_SIZE];
10     if (argc != 2)
11     {
12         printf("using: makelog \"<command>\n");
13         return -1;
14     }
15     f = popen(argv[1], "r");
16     if (f == NULL)
17     {
18         perror("Error:\n");
19         return -1;
20     }
21     o = fopen("log.txt", "w");
22     while ((len = fread(buf, 1, BUF_SIZE, f)) != 0)
23     {
24         write(1, buf, len);
25         fwrite(buf, 1, len, o);
26     }
27     pclose(f);
28     fclose(o);
29     return 0;
30 }
```

```

vorplochek@vorplochek-VirtualBox:~/lab_ECM/#6$ ./makelog "ls -la"
total 184
drwxrwxr-x 2 vorplochek vorplochek 4096 тра 6 23:07 .
drwxrwxr-x 7 vorplochek vorplochek 4096 тра 6 22:44 ..
prw----- 1 vorplochek vorplochek    0 тра 6 22:51 fifofile
-rw-rw-r-- 1 vorplochek vorplochek    0 тра 6 23:12 log.txt
-rwxrwxr-x 1 vorplochek vorplochek 17888 тра 6 22:50 makelog
-rw-rw-r-- 1 vorplochek vorplochek   594 тра 6 22:50 makelog.c
-rwxrwxr-x 1 vorplochek vorplochek 16960 тра 6 22:49 pipe
-rw-rw-r-- 1 vorplochek vorplochek   594 тра 6 22:49 pipe.c
-rwxrwxr-x 1 vorplochek vorplochek 16880 тра 6 22:46 semclient
-rw-rw-r-- 1 vorplochek vorplochek   293 тра 6 22:45 semclient.c
-rwxrwxr-x 1 vorplochek vorplochek 17888 тра 6 23:07 semserv
-rw-rw-r-- 1 vorplochek vorplochek   485 тра 6 23:06 semserv.c

vorplochek@vorplochek-VirtualBox:~/lab_ECM/#6$ ./pipe
Some string that was output
vorplochek@vorplochek-VirtualBox:~/lab_ECM/#6$ ./semclient
123
654

vorplochek@vorplochek-VirtualBox:~/lab_ECM/#6$ ./semclient
123
654

```

Висновок: Під час виконання цієї лабораторної роботи я ознайомився з каналами та міжпроцесною взаємодією у Linux. Навчився створювати та працювати з каналами. Навчився розбивати розв'язання складного завдання на кілька простих операцій, кожна з яких довіряється окремій невеликій програмі.