

JEGYZŐKÖNYV

Operációs rendszerek BSc

2021. tavasz féléves feladat

Készítette: **Simonyák János**

Neptunkód: **MZ727W**

A feladat leírása:

27: Készítsen olyan C programot, amely egy nevesített csővezetéken keresztül fogad szavakat és ezeket egy fájlba menti egy sorszámmal együtt. Illetve készítsen egy olyan kliensprogramot, ami ebbe a csőbe adatokat helyez el, amit majd a felhasználótól kér be.

A fő program a SIGTERM szignál hatására szüntesse meg a nevesített csövet és lépjen ki.

A feladat elkészítésének lépései:

1. A feladatleírás alapján két fájl létrehozása. (MZ727W_r.c és MZ727W_w.c)
2. Megfelelő könyvtárak importálása.

```
1  #include <stdio.h>
2  #include <unistd.h>
3  #include <string.h>
4  #include <fcntl.h>
5  #include <signal.h>
```

```
1  #include <stdio.h>
2  #include <unistd.h>
3  #include <sys/stat.h>
4  #include <fcntl.h>
5  #include <signal.h>
```

Az író és az olvasó programhoz felhasznált könyvtárak.

3. Szignálkezelő függvény létrehozása

```
void SignalKezelo(int sig) {
    signal(SIGTERM, SIG_IGN);
    printf("\n% jelzést kaptam.\n", sig);
    fflush(stdout);
    signal(SIGTERM, SIG_DFL);
    _exit(2);
}
```

4. Író program main függvényének megírása.

```
16  int main()
17  {
18      signal(SIGINT, handle_sigint);
19      char s[1024];
20      char t[1024];
21      char name[] = "output";
22      FILE *fp;
23      fp = fopen("szavak", "w");
24      int fd;
25      int sorszam = 1;
26      while(1) {
27          fd = open(name, O_RDONLY);
28          if(fd == -1) {
29              printf("Nem létezik \"%s\" nevű fájl.\n", name);
30              return 1;
31          }
32          else {
33              strcpy(t, s);
34              read(fd, s, sizeof(s));
35              if (strcmp(s, t) != 0) {
36                  printf("\n%d. %s", sorszam, s);
37                  fprintf(fp, "%d. %s\n", sorszam, s);
38                  sorszam++;
39              }
40              close(fd);
41          }
42      }
43  }
```

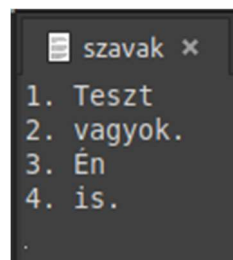
5. Olvasó és fájlba író program main megírása.

```
8 void SignalKezelo(int sig) {
9     signal(SIGTERM, SIG_IGN);
10    printf("\n% jelzést kaptam.\n", sig);
11    fflush(stdout);
12    signal(SIGTERM, SIG_DFL);
13    _exit(2);
14 }
15
16 int main(){
17     int fd;
18     signal(SIGINT, SignalKezelo);
19     char szoveg[256];
20     char name[] = "output";
21     mkfifo(name, S_IWUSR | S_IRUSR );
22
23     while(1) {
24         scanf("%s", szoveg);
25         fd = open(name, O_WRONLY);
26         write(fd, szoveg, 12);
27         close(fd);
28     }
29 }
```

A futtatás eredménye:

<pre>> ./read 1. Teszt. 2. vagyok. 3. Én ^C4. is. 2 jelzést kaptam.</pre>	<pre>> ./write Teszt. vagyok. Én is. ^C 2 jelzést kaptam.</pre>
--	--

Balról jobbra: az olvasó és az író program futásának eredménye.



szavak x

1. Teszt
2. vagyok.
3. Én
4. is.

A létrehozott szövegfájl tartalma.