

# JEGYZŐKÖNYV

Operációs rendszerek BSc

2021. tavasz féléves feladat

Készítette: Orosz Dániel

Neptunkód: C5S7FM

## A feladat leírása:

26. Írjon egy olyan C programot, amely létrehoz egy FIFO állományt és tetszőleges két folyamat között működik. A feladat egy olyan kliens-szerver rendszer (két különálló állományból áll!) létrehozása, amelyben a kliens küld egy számot a szervernek, mire a szerver válaszként visszaküldi a szám negyzetét.

## A feladat elkészítésének lépései:

Ahogy a feladat leírásában is szerepel két különálló állományt kell létrehozni, így kezdésként ezt a két állományt hoztam létre. (C5S7FM\_client.c, C5S7FM\_server.c)

A kliens állomány FIFO-ba író main() függvényével kezdtem, ami egy végtelen ciklusban várja az inputot a felhasználótól amit beleír a fájlba, hogy a szerver azt kitudja onnan olvasni. A fájlnak a neve „a.out” ebben az esetben.

```
int main()
{
    int fd;
    char * fifoname = "/tmp/myfifo";
    mkfifo(fifoname, S_IWUSR | S_IRUSR);
    char number[40];
    double power;

    while(1){
        fd = open(fifoname, O_WRONLY);
        fgets(number, 40, stdin);
        write(fd, number, strlen(number)+1);
        close(fd);
    }
}
```

A kliens állománynak a későbbiekben ki kell olvasni a FIFO állományból az új adatot amit a szerver állomány ír bele oda. Ez a beolvasás szintén a végtelen ciklusban helyezkedik el.

```
        fd = open(fifoname, O_RDONLY);
        read(fd, number, sizeof(number));
        power = atof(number);
        double paw = power * power;
        printf("A szám negyzete: %.0f\n", paw);
        close(fd);
    }
    return 0;
}
```

A szerver állománynak ezt a létrehozott FIFO állományt szintén meg kell nyitnia és írnia bele, ezért a kliens állományban már megírt hasonló módon itt is egy végtelen ciklusban van ez megvalósítva az alábbi módon. A kapott számot kiírja a felületre, hogy biztos legyen a sikeres volt a kiolvasás a FIFO állományból és valóban azt olvasta ki amit kellett.

```
int main()
{
    int fd;
    char * fifoname = "/tmp/myfifo";
    mkfifo(fifoname, S_IWUSR | S_IRUSR);

    char number[40], number2[40];

    while(1){
        fd = open(fifoname, O_RDONLY);
        read(fd, number, 40);
        printf("Kapott szám: %s", number);
        close(fd);
    }
}
```

A kiolvasott számot négyzetre emeli, azután visszaírja a FIFO állományba, hogy utána a kliens azt betudja olvasni és kiírni a felületre a felhasználónak. Ez szintén a végtelen ciklusban helyezkedik el.

```
        double power = atof(number);
        double ret = power * power;
        *(double *)number = ret;

        fd = open(fifoname, O_WRONLY);
        write(fd, number, strlen(number));
        close(fd);
    }
    return 0;
}
```

### A futtatás eredménye:

```
dani@dani-VirtualBox: ~/Desktop/felevi
File Edit View Search Terminal Help
dani@dani-VirtualBox:~/Desktop/felevi$ cc C5S7FM_client.c
dani@dani-VirtualBox:~/Desktop/felevi$ ./a.out
3
A szam negyzete: 9
4
A szam negyzete: 16
5
A szam negyzete: 25
6
A szam negyzete: 36

dani@dani-VirtualBox: ~/Desktop/felevi
File Edit View Search Terminal Help
dani@dani-VirtualBox:~/Desktop/felevi$ cc C5S7FM_server.c
dani@dani-VirtualBox:~/Desktop/felevi$ ./a.out
Kapott szam: 3
Kapott szam: 4
Kapott szam: 5
Kapott szam: 6
█
```