
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2
Nastavnik: Živojin Šuštran
Odsek: Softversko inženjerstvo, Računarska tehnika i informatika
Kolokvijum: Treći, jul 2025.
Datum: 22. 08. 2025.

Treći kolokvijum iz Operativnih sistema 2

Kandidat: _____

Broj indeksa: _____ *E-mail:* _____

Kolokvijum traje 90 minuta. Dozvoljeno je korišćenje literature.

Zadatak 1 _____/10 *Zadatak 3* _____/10
Zadatak 2 _____/10

Ukupno: _____/30 = _____%

Napomena: Ukoliko u zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Ocenjivanje unutar potpitanja je po sistemu "sve ili ništa", odnosno nema parcijalnih poena. Kod pitanja koja imaju ponuđene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

1. (10 poena) Arhitektura operativnih sistema

Neki uređaj ima napisan drajver za Linux operativni sistem. Uređaj ima DMA kontroler na sebi. Potrebno je napisati funkciju koja inicira DMA ulazni transfer:

```
void* init_dma_transfer(uint64_t transfer_size);
```

Funkcija vraća adresu gde će se podaci nalaziti nakon transfera, dok parametar određuje veličinu transfera. Veličina transfera neće biti veća od veličine jedne stranice. Na raspolaganju je funkcija:

```
void* alloc_page();
```

koja alokira prostor u memoriji veličine jedne stranice koji je poravnat na početak stranice. Drajveru je potrebno proslediti virtuelnu adresu gde će se smestiti podaci i veličinu transfera. Uređaj pristupa memoriji samo preko fizičkih adresa. Deklaracije koje se dostavljaju uz drajver uređaja su:

```
struct dma_init_in_transfer {  
    uint64_t virtual_address;  
    uint64_t size;  
};
```

```
#define DMA_INIT_OP ...
```

...

Uređaj se u sistemu prikazuje preko fajla `/dev/os2_device`. Uređaj se kontroliše pomoću `ioctl` sistemskog poziva, čija je dokumentacija u nastavku:

NAME

ioctl - control device

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <sys/ioctl.h>
```

```
int ioctl(int fd, unsigned long op, ...); /* glibc, BSD */
```

```
int ioctl(int fd, int op, ...); /* musl, other UNIX */
```

DESCRIPTION

The `ioctl()` system call manipulates the underlying device parameters of special files. In particular, many operating characteristics of character special files (e.g., terminals) may be controlled with `ioctl()` operations. The argument `fd` must be an open file descriptor.

The second argument is a device-dependent operation code. The third argument is an untyped pointer to memory. It's traditionally `char *argp` (from the days before `void *` was valid C), and will be so named for this discussion.

An `ioctl()` op has encoded in it whether the argument is an in parameter or out parameter, and the size of the argument `argp` in bytes. Macros and defines used in specifying an `ioctl()` op are located in the file `<sys/ioctl.h>`. See NOTES.

RETURN VALUE

Usually, on success zero is returned. A few `ioctl()` operations use the return value as an output parameter and return a nonnegative value on success. On error, `-1` is returned, and `errno` is set to indicate the error.

Rešenje:

2. (10 poena) Operativni sistem Linux

Napisati bash skriptu koja proverava da li se u skripti koja joj je prosleđena kao argument nalaze samo komande koje su izvršive. Pretpostaviti da se u zadatoj skripti nalaze samo komande sledećeg tipa:

```
<cmd> <arg0> <arg1> ... <argN>
```

Ukoliko nisu sve komande izvršive skripta treba da ispiše one koje nisu.

Postoji komanda `which`, čije je upustvo dato:

NAME

which - locate a command

SYNOPSIS

which [-as] filename ...

DESCRIPTION

which returns the pathnames of the files (or links) which would be executed in the current environment, had its arguments been given as commands in a strictly POSIX-conformant shell. It does this by searching the PATH for executable files matching the names of the arguments. It does not canonicalize path names.

OPTIONS

- a print all matching pathnames of each argument*
- s silently return 0 if all of the executables were found or 1 otherwise*

EXIT STATUS

- 0 if all specified commands are found and executable*
- 1 if one or more specified commands is nonexistent or not executable*
- 2 if an invalid option is specified*

Rešenje:

3. (10 poena) Operativni sistem Linux

Data funkcija obavlja transfer podataka iz zadanog fajla u memoriju. Funkcija `read` u konkretnom Linux operativnom sistemu i na konkretnom računaru radi transfer u rečima veličine 32 bita. Na raspolaganju je sledeća funkcija:

```
void copy512(void* dst, void* src, uint64_t size);
```

koja kopira podatke koristeći vektorske instrukcije procesora. Vektorske instrukcije podržavaju transfere u rečima veličine 512 bita. Samim tim funkcija `copy512` radi brže transfer nego funkcija `read`. Promeniti implementaciju funkcije transfer tako da koristi funkciju `copy512`.

```
void transfer(int fd, uint64_t size, uint64_t offset, void *buffer) {
    lseek(fd, offset, SEEK_SET);
    ssize_t bytesRead = 0;

    while (size > 0 && (bytesRead = read(fd,buffer+bytesRead,size)) > 0) {
        size -= bytesRead;
    }
}
```

Rešenje: