

---

---

Elektrotehnički fakultet u Beogradu  
Katedra za računarsku tehniku i informatiku

*Predmet:* Operativni sistemi 1  
*Nastavnik:* prof. dr Dragan Milićev  
*Odsek:* Softversko inženjerstvo, Računarska tehnika i informatika  
*Kolokvijum:* Drugi, jun 2022.  
*Datum:* 12. 6. 2022.

*Drugi kolokvijum iz Operativnih sistema 1*

*Kandidat:* \_\_\_\_\_

*Broj indeksa:* \_\_\_\_\_ *E-mail:* \_\_\_\_\_

*Kolokvijum traje 1,5 sat. Dozvoljeno je korišćenje literature.*

*Zadatak 1* \_\_\_\_\_ /10                      *Zadatak 3* \_\_\_\_\_ /10  
*Zadatak 2* \_\_\_\_\_ /10

**Ukupno:** \_\_\_\_\_ /30 = \_\_\_\_\_ % = \_\_\_\_\_ /15

**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)

Koristeći samo sistemske pozive *fork*, *wait/waitpid* i *exit*, kao i funkciju *printf*, napisati C program koji pronalazi maksimalnu vrednost svih elemenata ogromne matrice dimenzije  $M$  puta  $N$  elemenata tipa `double` tako što uporedo pronalazi maksimum u svakoj vrsti matrice (maksimum svake vrste pronalazi u po jednom od pokrenutih procesa-dece), a onda pronalazi maksimum tih maksimuma. U slučaju greške, ovaj program treba da ispiše poruku o grešci i vrati status -1, a u slučaju uspeha treba da ispiše pronađeni maksimum i vrati status 0. Pretpostaviti da je matrica već nekako inicijalizovana.

```
const int M = ..., N = ...;
extern double mat[M][N];
```

Rešenje:

## 2. (10 poena)

U školsko jezgro dodaje se koncept *uslova* (engl. *condition*) za uslovnu sinhronizaciju, podržan klasom `Condition` čiji je interfejs dat dole. Uslov može biti ispunjen ili neispunjen; inicijalna vrednost zadaje se konstruktorom. Niti koje smeju da nastave izvršavanje iza neke tačke samo ako je uslov ispunjen treba da pozovu operaciju `wait`, koja ih suspenduje ako uslov nije ispunjen. Bilo koja nit koja ispuni uslov to objavljuje pozivom operacije `set`; sve niti koje čekaju na taj uslov tada nastavljaju izvršavanje. Kada neka nit promeni uslov tako da on više nije ispunjen, treba da pozove operaciju `clear`. Implementirati klasu `Condition`.

```
class Condition {
public:
    Condition (bool init = false);
    void set ();
    void clear ();
    void wait ();
};
```

Rešenje:

### 3. (10 poena)

Više uporednih niti-pisaca upisuje izračunate celobrojne dvodimenzionalne koordinate  $(x, y)$  na koje treba pomeriti robota u deljeni objekat klase `SharedCoord` čiji je interfejs dat dole; svaka ovakva nit nezavisno upisuje svoj par izračunatih koordinata pozivom operacije `write` ove klase. Jedna nit-čitalac periodično očitava par koordinata iz tog deljenog objekta i pomera robota na očitane koordinate; ova nit to radi pozivom operacije `read` ove klase, nezavisno od pisaca, svojim tempom, tako da svaki put čita poslednje upisane koordinate (može pročitati više puta isti par koordinata ili neke izračunate koordinate i preskočiti).

Implementirati klasu `SharedCoord` uz neophodnu sinhronizaciju korišćenjem najmanjeg broja semafora školskog jezgra.

```
class SharedCoord {
public:
    SharedCoord ();
    void read (int& x, int& y);
    void write (int x, int y);
};
```

Rešenje: