

Rešenja prvog kolokvijuma iz Operativnih sistema 2 Januar 2023.

1. (10 poena)

```
class EDFScheduler {
public:
    EDFScheduler () : head(0) {}

    PCB* getRunning () const { return head; }
    void removeRunning () { PCB* p = head; head = head->next; return p; }
    void add (PCB*);

private:
    PCB* head;
};

void EDFScheduler::add (PCB* p) {
    if (!head || head->deadline > p->deadline) {
        p->next = head;
        head = p;
        return;
    }
    for (PCB* cur = head; cur; cur = cur->next)
        if (!cur->next || cur->next->deadline > p->deadline) {
            p->next = cur->next;
            cur->next = p;
            return;
        }
}
```

2. (10 poena)

```
class FlipFlop {

    public synchronized void flip () {
        while (i < N) wait();
        i = 0;
        // do flip
    }

    public synchronized void flop () {
        // do flop
        if (i < N) {
            i++;
            if (i == N) notify();
        }
    }

    private int i = N;
}
```

3. (10 poena)

```
public class Client {
    private static String[] SERVERS = {"srv1.etf.bg.rs", "srv2.etf.bg.rs"};
    private final static int PORT = 5555;
    private final static int MAX_INDEX = 100;

    public void processDataFromServer() {
        Service services[] = new Service[SERVERS.length];

        for (int i = 0; i < services.length; i++) {
            try {
                services[i] = new Service(new Socket(SERVERS[i], PORT));
            } catch (IOException e) {
                services[i] = null;
            }
        }

        String data = collectData(services);
        processData(data);
    }

    private String collectData(Service[] services) {
        StringBuilder sb = new StringBuilder();
        for (int i = 0; i < MAX_INDEX; i++) {
            String data = "";
            for (Service service : services) {
                service.sendMessage("GET#" + i);
                try {
                    String response = service.receiveMessage();
                    if (response.equals("PRESENT")) {
                        data = service.receiveMessage();
                        break;
                    }
                } catch (IOException ignored) {
                }
            }
            sb.append(data);
        }

        return sb.toString();
    }

    public void processData(String data) {
        // Data processing
    }
}
```

Klasa Service je data na vežbama.