

Prvi kolokvijum iz Operativnih sistema 1

Računarska tehnika i informatika

April 2019.

1. (10 poena)

```
static IORequest* pending[2] = {0,0}; // Pending requests for two channels

void startIO (int i) { // Helper: start a new transfer with channel i
    if (ioHead==0) {
        *ioCtrl &= ~(1<<i); // Stop channel i
        pending[i] = 0;
        return;
    }
    pending[i] = iohead;
    ioHead = ioHead->next; // Remove the request from the list
    pending[i]->next = 0;
    // Start I/O with channel i:
    *ioCtrl |= (1<<i);
    return;
}

void handleIO (int i) { // Helper: handle I/O with channel i
    if (!(ioStatus)&(1<<i)) return;
    if ((ioStatus)&(1<<(2+i)))
        pending[i]->status = -1; // Error in I/O
    else { // Transfer the next data item
        *(pending[i]->buffer)++ = ioData[i];
        if (--pending[i]->size)
            return;
        else {
            pending[i]->status = 0; // Transfer completed successfully
        }
    }
    startIO(i); // Initiate the next transfer with this channel
}

void transfer () {
    if (!ioHead) return;
    startIO(0);
    startIO(1);
    while (pending[0] || pending[1]) {
        if (pending[0]) handleIO(0);
        if (pending[1]) handleIO(1);
    }
}
```

2. (10 poena)

```
*sp-- = arg;
*sp-- = &exit;
pcb->sp = sp;
pcb->pc = pf;
```

3. (10 poena)

```
const int M = ..., N = ...;
int mat[M][N];
int sums[M];
int pid[M];

void sum (int row) {
    int s = 0;
    for (int i=0; i<N; i++)
        s += mat[row][i];
    sums[row] = s;
}

int par_sum () {
    for (int i=0; i<M; i++) {
        pid[i] = fork();
        if (pid[i]==0) {
            sum(i);
            exit();
        }
    }

    int s = 0;
    for (int i=0; i<M; i++) {
        wait(pid[i]);
        s += sums[i];
    }
    return s;
}
```