

# Rešenja zadatka za drugi kolokvijum iz Operativnih sistema 1 Maj 2015.

1. (10 poena) Deo koda oba procesa izgleda isto:

```
#include <fcntl.h>
#include <sys/stat.h>
#include <semaphore.h>

// Initialization:
const char* mutexName = "/myprogram_mutex";
sem_t* mutex = sem_open(mutexName,O_CREAT,1);

...
// Use for mutual exclusion
sem_wait(mutex);
// Critical section
sem_post(mutex);
...

// Release the semaphore when it is no longer needed:
sem_close(mutex);
```

2. (10 poena)

```
void proc_relocate (PCB* pcb, char* to) {
    if (pcb->mem_base_addr==to) return;
    memcpy(to,pcb->mem_base_addr,pcb->mem_size);
    pcb->mem_base_addr = to;
}

void mem_compact () {
    if (mem_free_head==0) return; // No free memory, no need for compaction
    char* to = user_proc_mem_start;
    for (PCB* pcb=proc_mem_head; pcb!=0; pcb=pcb->mem_next) {
        proc_relocate(pcb,to);
        to+=pcb->mem_size;
    }
    size_t free_mem = user_proc_mem_end-to+1;
    if (free_mem>=sizeof(FreeSegment) {
        mem_free_head = (FreeSegment*)to;
        mem_free_head->size = free_mem;
        mem_free_head->next = 0;
    }
    else mem_free_head = 0; // No more free memory (should not ever happen)
}
```

**3. (10 poena)**

a)(2)

<i>Page#</i>	<i>Frame#</i>	<i>RWX</i>
00h	10h	001
01h	11h	001
02h	12h	001
03h	13h	001
12h	14h	100
13h	15h	100
14h	16h	100
1Ah	17h	110
1Bh	18h	110

b) PMT procesa-roditelja:

<i>Page#</i>	<i>Frame#</i>	<i>RWX</i>
00h	10h	001
01h	11h	001
02h	12h	001
03h	13h	001
12h	14h	100
13h	15h	100
14h	16h	100
1Ah	17h	100
1Bh	18h	100

PMT procesa-deteta:

<i>Page#</i>	<i>Frame#</i>	<i>RWX</i>
00h	10h	001
01h	11h	001
02h	12h	001
03h	13h	001
12h	14h	100
13h	15h	100
14h	16h	100
1Ah	17h	100
1Bh	18h	100

c) PMT procesa-deteta:

<i>Page#</i>	<i>Frame#</i>	<i>RWX</i>
00h	10h	001
01h	11h	001
02h	12h	001
03h	13h	001
12h	14h	100
13h	15h	100
14h	16h	100
1Ah	19h	110
1Bh	18h	100