Systems II

2020/21

Exam 2

Collaboration is not allowed during the exam. Each task from the computational part, which contributes 70 % towards the grade, is worth the same amount of points. Each task from the theoretical part, which contributes 30 % towards the grade, is worth the same amount of points. Literature and notes are not allowed. You may use a simple calculator. The exam takes 120 minutes.

Good luck and have fun!

Full name: _____

Enrollment ID: _____

Signature: _____

Task 1 (25 points)

We have 4 periodical processes Π_1, Π_2, Π_3 and Π_4, that process a video. We know their periods P_i and their running times C_i (in milliseconds). For:

(a) the rate monotonic scheduling (RMS) algorithm and

(b) the earliest deadline first (EDF) algorithm

draw diagrams that displays the schedule for those processes for a period of 150 ms. The times are the following:

Process	P_i	C_i
Π_1	45	10
Π_2	40	10
Π_3	35	5
Π_4	50	10

In the case of equal priorities the process that is already running has the priority. If this is not the case then take the process with the lowest ID. Does any of the above algorithms fail?

Task 2

Our system is using virtual memory and has 48-bit virtual address space and 32-bit physical address space. Page size is 8 KiB.

(a) How many entries are needed for a *single-level* page table?

What does the *two-level* page table look like if the part of the virtual address which identifies the page number is split into two fields, such that the second part has 3 bits more than the first part?

(b) Our system also has the TLB (Translation Lookaside Buffer) with 64 entries. We run the following program which reads 32-bit integers from an array of size 200 000:

```
int[] t = new int[200000];

for (int i=19; i>=0; i--) {
    for (int j=0; j<10000; j++) {
        // we access t[ i + 20 * j ]
    }
}
```

How efficient is the TLB in this case? More precisely, how many times will a TLB miss occur (resulting in page table lookup)? Assume that the TLB is initially empty.

Task 3 (25 points)

Our system has 5 page frames and there are 8 pages (numbered $0, 1, \ldots, 7$). Assume that all frames are initially empty. The reference sequence is the following:

7 1 2 3 1 2 4 0 5 1 6 2 5 1 3 1 2 0

(a) How many page faults will occur if the FIFO page replacement is used?

(b) What if we use the "second chance" algorithm?

Your answer should be justified (which references cause page faults?).

Task 4 (25 points)

Consider a hard disk that rotates at 7 200 RPM. The disk has 14 heads. Each track contains 1220 sectors. Each sector contains 1 KiB (= 1024 B).

(a) Marcus determined the capacity of the disk, which is 74.12 GiB (he rounded the result slightly). How many cylinders does the disk have?

(b) How much time is needed to read the data from the entire disk (track per track, sector per sector) if the arm needs 1.9 ms to move to an adjacent cylinder?

(c) We want the maximum data transfer rate (between the disk and the controller's buffer) to be 160 MiB/s. To achieve this goal, we will speed up the rotation. How many revolutions per minute would that disk have to do?

Theoretical part

(a) Suppose that we are dealing with an OS that is *not* using virtual memory. What is *static relocation* and what is *dynamic relocation*?

(b) Draw the *state diagram* (directed graph) which shows the three basic *states* of a process and the possible *transitions* between them.

(c) Briefly describe the *dining philosophers problem*. (I am not interested in solutions. What I would like to see is the description of the problem.)

(d) What is a *device driver*? What is a *device controller*?

(e) The UNIX operating system has *special files* (which are usually kept in the `/dev` directory). Explain the purpose of special files.

(f) What is the *page table*? What is the purpose of R and M bits?