https://folk.ntnu.no/michaeng/tdt4186_21/
michael.engel@ntnu.no

Theoretical exercises
Spring 2021

Theoretical Exercises 5

I/O and file systems

Please submit solutions on Blackboard by Thursday, 25.03.2021 12:00h

5.1 Disk scheduling (6 points)

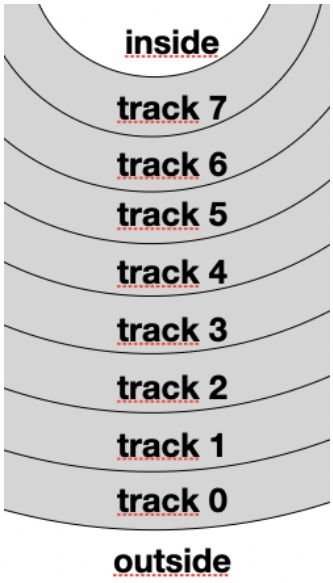
a. Assume a magnetic disk with 8 tracks. After each *second* read request (starting from L_1), the I/O scheduler receives additional read requests which are grouped (requested) together (L_2 and finally L_3).

Initially, the read/write head of the disk is at track 0.

Give the I/O scheduling order that would be performed according to the **SSTF (shortest seek time first) algorithm**:

$L_1 = \{2,4,3,1\}, L_2 = \{5,6\}, L_3 = \{0,7\}$

Access order:



b. Assume the same magnetic disk as above. The read requests in L_1 are already known to the disk scheduler. After *three* completed requests, the requests in L_2 arrive, and after three additional requests (i.e., after the sixth request), the requests in L_3 arrive.

Initially, the read/write head of the disk is at track 0.

Give the I/O scheduling order that would be performed according to the **elevator algorithm**:

$L_1 = \{1,4,7,2\}, L_2 = \{4,6,0\}, L_3 = \{5,2\}$

Access order:

5.2 FAT file system (4 points)

This question is related to a task you would need to perform if you were a computer security expert doing a *forensic analysis* of a disk.
Given a hexadecimal dump of blocks on the disk and a description of the block contents, you need to figure out the meaning of that data.
For an MS-DOS FAT16 floppy disk, you obtain the following hexadecimal dump of a *directory block* (note: addresses are in *decimal*):

```
address data bytes ..................................  ASCII representation

0009728 49 4f 20 20 20 20 20 20 53 59 53 27 00 00 00 00  IO      .SYS
0009744 00 00 00 00 00 00 08 5d 62 1b 1d 00 16 9f 00 00
0009760 4d 53 44 4f 53 20 20 20 53 59 53 27 00 00 00 00  MSDOS   .SYS
0009776 00 00 00 00 00 00 08 5d 62 1b 6d 00 38 95 00 00
0009792 43 4f 4d 4d 41 4e 44 20 43 4f 4d 20 00 00 00 00  COMMAND .COM
0009808 00 00 00 00 00 00 07 5d 62 1b b8 00 39 dd 00 00
0009824 44 42 4c 53 50 41 43 45 42 49 4e 27 00 00 00 00  DBLSPACE.BIN
0009840 00 00 00 00 00 00 08 5d 62 1b 27 01 f6 fc 00 00
0009856 4d 53 44 4f 53 20 20 20 20 20 20 28 00 00 00 00  MSDOS
0009872 00 00 00 00 00 00 1a 88 99 1c 00 00 00 00 00 00
0009888 46 44 49 53 4b 20 20 20 45 58 45 20 00 00 00 00  FDISK   .EXE
0009904 00 00 00 00 00 00 36 59 62 1b 02 00 17 73 00 00
```

The structure of a FAT16 directory entry is as follows (all numbers are stored in *little endian byte order*:

Bytes	Content
0–10	File name (8 bytes) with extension (3 bytes)
11	Attribute - a bitvector. Bit 0: read only. Bit 1: hidden. Bit 2: system file. Bit 3: volume label. Bit 4: subdirectory. Bit 5: archive. Bits 6-7: unused.
12–21	Reserved
22–23	Time (5/6/5 bits, for hour/minutes/doubleseconds)
24–25	Date (7/4/5 bits, for year-since-1980/month/day)
26–27	Starting cluster (0 for an empty file)
28-31	File size in bytes

For each directory entry, find out:

a. The name of the entry

b. The type of the entry including the set of file attributes

c. The starting cluster number

d. The file size in bytes