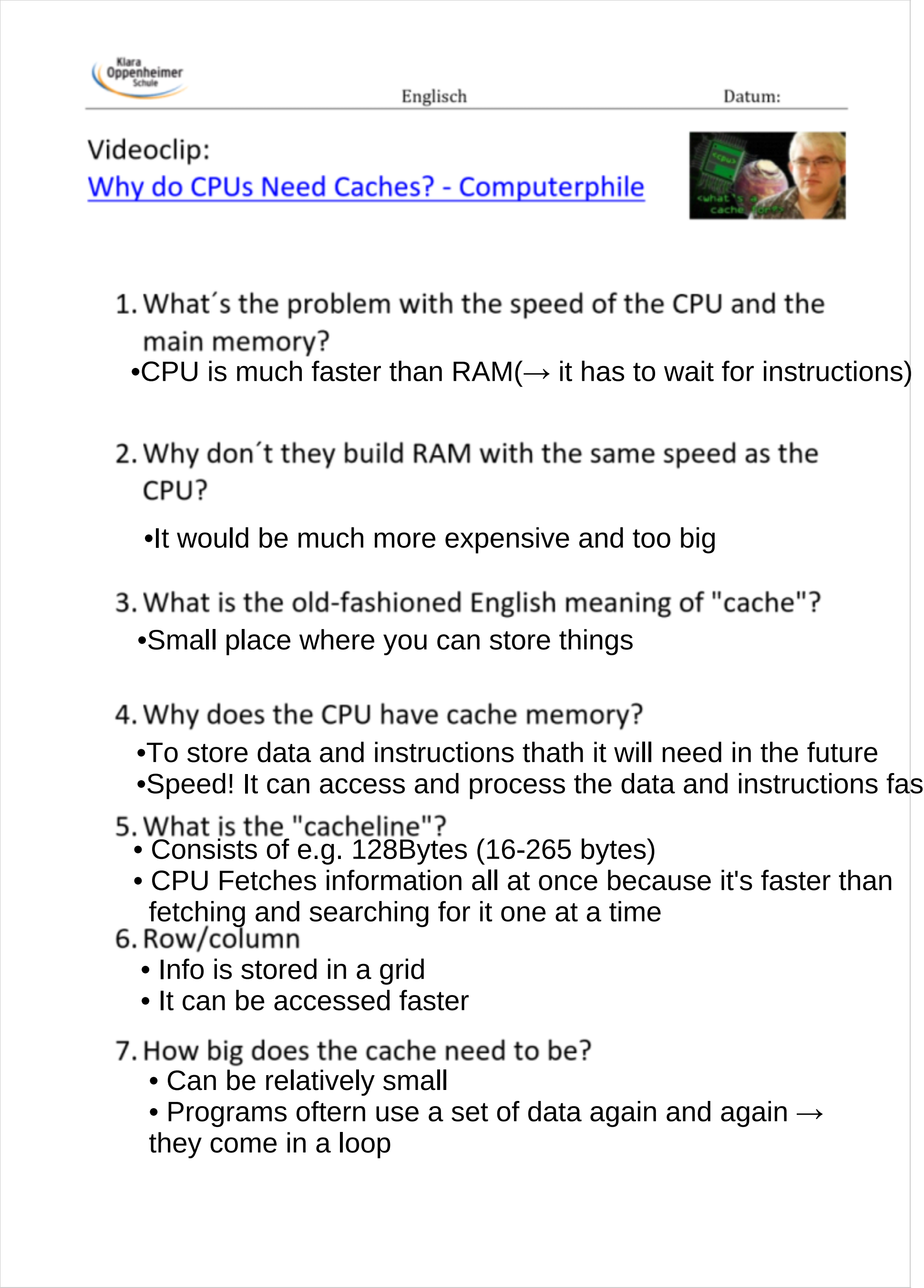
Cache memory -Video und Text

Freitag, 4. Februar 2022 10:34

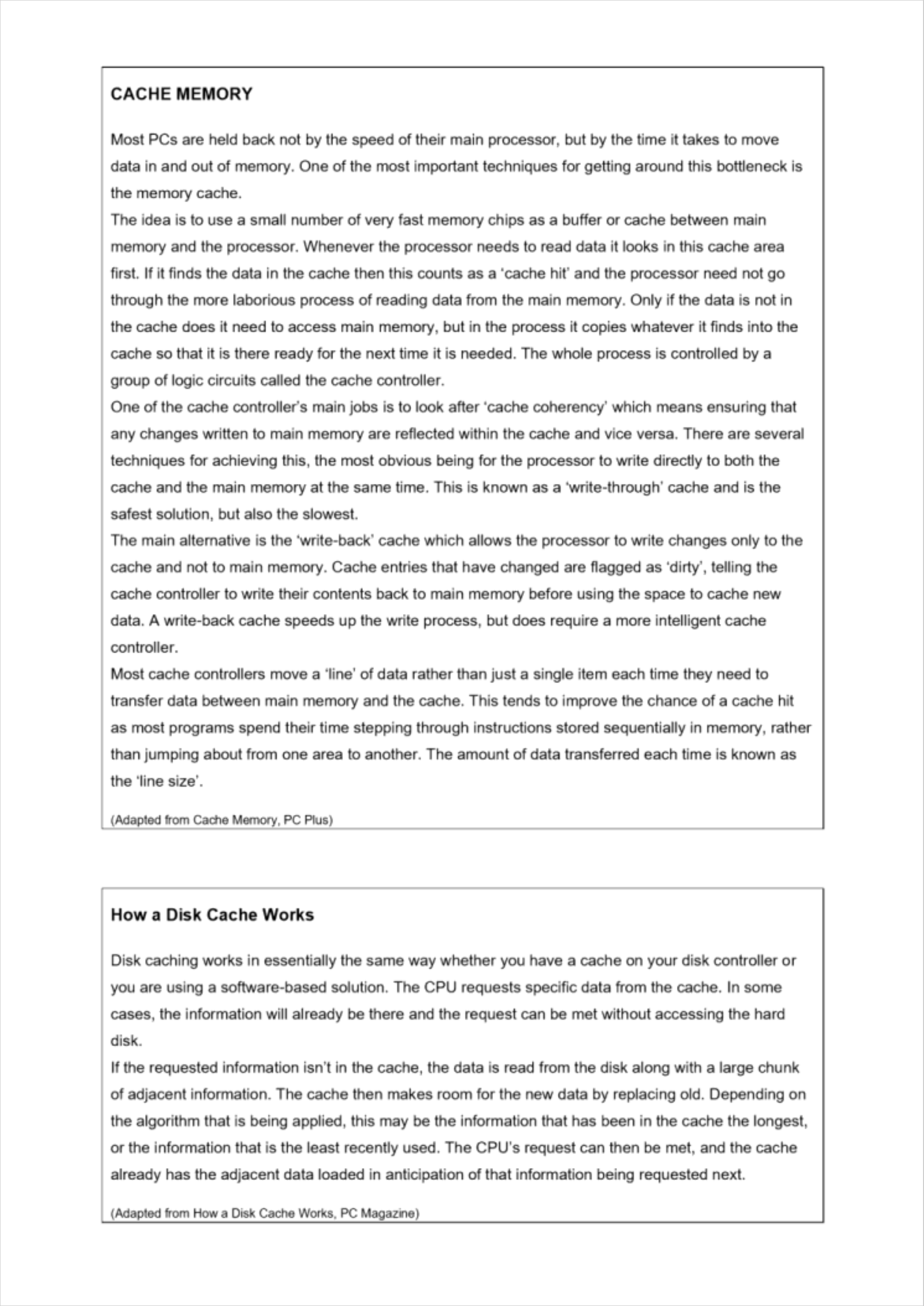
Videoclip:
Why do CPUs Need Caches? - Computerphile


Klara Oppenheimer Schule

Englisch Datum:

Videoclip:
Why do CPUs Need Caches? - Computerphile

1. What´s the problem with the speed of the CPU and the main memory?
   - CPU is much faster than RAM(→ it has to wait for instructions)
2. Why don´t they build RAM with the same speed as the CPU?
   - It would be much more expensive and too big
3. What is the old-fashioned English meaning of "cache"?
   - Small place where you can store things
4. Why does the CPU have cache memory?
   - To store data and instructions thath it will need in the future
   - Speed! It can access and process the data and instructions faster
5. What is the "cacheline"?
   - Consists of e.g. 128Bytes (16-265 bytes)
   - CPU Fetches information all at once because it's faster than fetching and searching for it one at a time
6. Row/column
   - Info is stored in a grid
   - It can be accessed faster
7. How big does the cache need to be?
   - Can be relatively small
   - Programs oftern use a set of data again and again → they come in a loop



CACHE MEMORY

Most PCs are held back not by the speed of their main processor, but by the time it takes to move data in and out of memory. One of the most important techniques for getting around this bottleneck is the memory cache.

The idea is to use a small number of very fast memory chips as a buffer or cache between main memory and the processor. Whenever the processor needs to read data it looks in this cache area first. If it finds the data in the cache then this counts as a 'cache hit' and the processor need not go through the more laborious process of reading data from the main memory. Only if the data is not in the cache does it need to access main memory, but in the process it copies whatever it finds into the cache so that it is there ready for the next time it is needed. The whole process is controlled by a group of logic circuits called the cache controller.

One of the cache controller's main jobs is to look after 'cache coherency' which means ensuring that any changes written to main memory are reflected within the cache and vice versa. There are several techniques for achieving this, the most obvious being for the processor to write directly to both the cache and the main memory at the same time. This is known as a 'write-through' cache and is the safest solution, but also the slowest.

The main alternative is the 'write-back' cache which allows the processor to write changes only to the cache and not to main memory. Cache entries that have changed are flagged as 'dirty', telling the cache controller to write their contents back to main memory before using the space to cache new data. A write-back cache speeds up the write process, but does require a more intelligent cache controller.

Most cache controllers move a 'line' of data rather than just a single item each time they need to transfer data between main memory and the cache. This tends to improve the chance of a cache hit as most programs spend their time stepping through instructions stored sequentially in memory, rather than jumping about from one area to another. The amount of data transferred each time is known as the 'line size'.

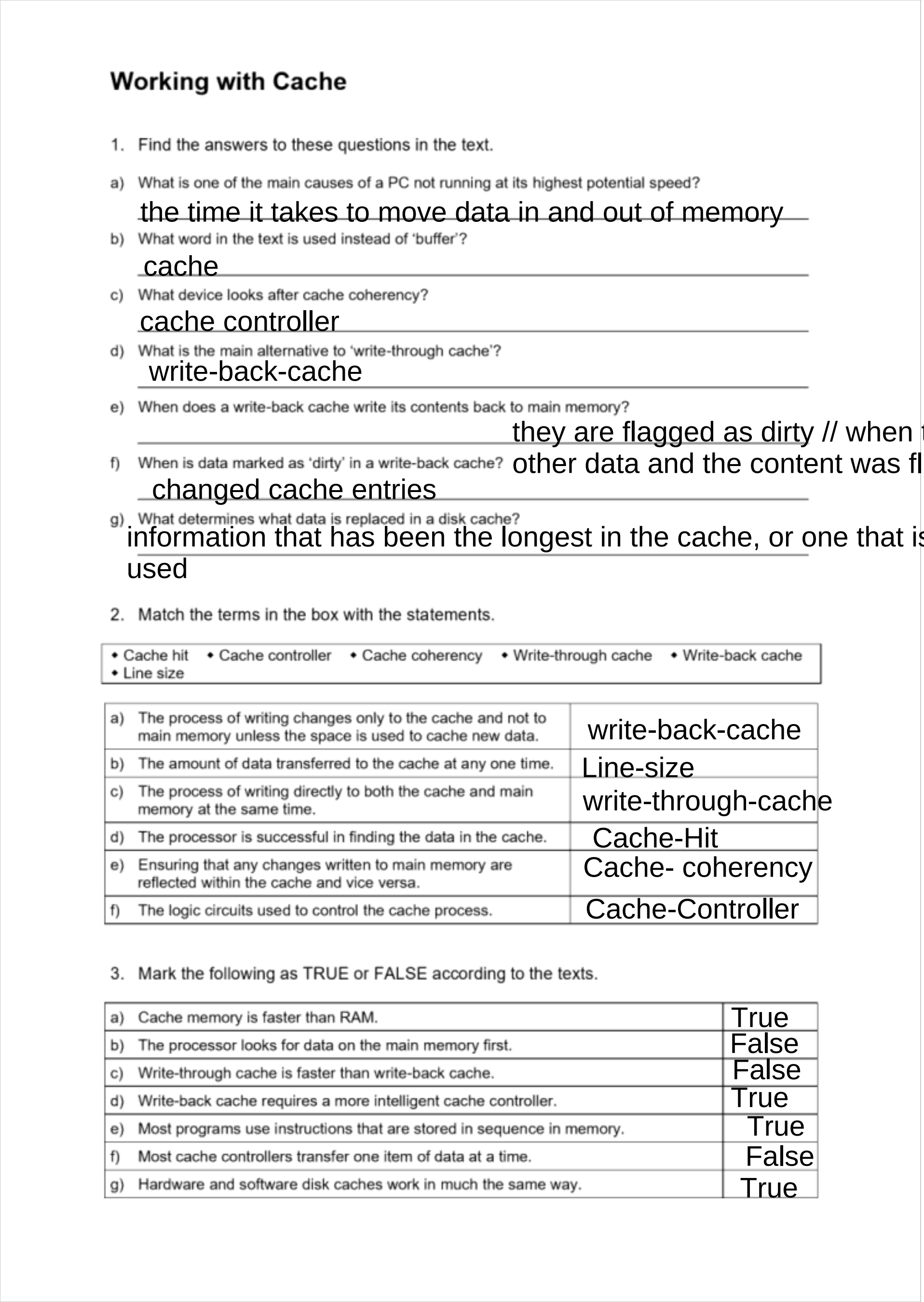
(Adapted from Cache Memory, PC Plus)

How a Disk Cache Works

Disk caching works in essentially the same way whether you have a cache on your disk controller or you are using a software-based solution. The CPU requests specific data from the cache. In some cases, the information will already be there and the request can be met without accessing the hard disk.

If the requested information isn't in the cache, the data is read from the disk along with a large chunk of adjacent information. The cache then makes room for the new data by replacing old. Depending on the algorithm that is being applied, this may be the information that has been in the cache the longest, or the information that is the least recently used. The CPU's request can then be met, and the cache already has the adjacent data loaded in anticipation of that information being requested next.

(Adapted from How a Disk Cache Works, PC Magazine)



Working with Cache

1. Find the answers to these questions in the text.

a) What is one of the main causes of a PC not running at its highest potential speed?

the time it takes to move data in and out of memory

b) What word in the text is used instead of 'buffer'?

cache

c) What device looks after cache coherency?

cache controller

d) What is the main alternative to 'write-through cache'?

write-back-cache

e) When does a write-back cache write its contents back to main memory?

they are flagged as dirty // when the space is needed for other data and the content was flagged as dirty

f) When is data marked as 'dirty' in a write-back cache?

changed cache entries

g) What determines what data is replaced in a disk cache?

information that has been the longest in the cache, or one that is least recentcly used

2. Match the terms in the box with the statements.

- Cache hit
- Cache controller
- Cache coherency
- Write-through cache
- Write-back cache
- Line size

a) The process of writing changes only to the cache and not to main memory unless the space is used to cache new data.	write-back-cache
b) The amount of data transferred to the cache at any one time.	Line-size
c) The process of writing directly to both the cache and main memory at the same time.	write-through-cache
d) The processor is successful in finding the data in the cache.	Cache-Hit
e) Ensuring that any changes written to main memory are reflected within the cache and vice versa.	Cache- coherency
f) The logic circuits used to control the cache process.	Cache-Controller

3. Mark the following as TRUE or FALSE according to the texts.

a) Cache memory is faster than RAM.	True
b) The processor looks for data on the main memory first.	False
c) Write-through cache is faster than write-back cache.	False
d) Write-back cache requires a more intelligent cache controller.	True
e) Most programs use instructions that are stored in sequence in memory.	True
f) Most cache controllers transfer one item of data at a time.	False
g) Hardware and software disk caches work in much the same way.	True