# Homework 2 Processor performance

1. The performance of a processor is affected by many different factors. For example, the speed at which the processor is clocked determines how much time is taken to complete instructions. A higher clock rate generally results in the quicker processing of instructions but this is not the only factor.   
     
   Describe **two** other factors that affect the performance of a processor. [6]

Cache size – the amount of frequently used data and instructions the cache can hold. The cache itself is a super-fast memory so whenever a CPU needs to fetch instructions or retrieve data, if the data or instruction is frequently used, it will be in the cache which makes fetching and retrieving it very efficient since the CPU doesn’t have to search in the massive RAM. The cache is closer and easier to search through. A larger cache size means you can store more different frequently used data, so even less likely to have to search through RAM for data and instructions. This results in improving the performance of the processor by speeding up the fetch phase of the fetch-decode-execute cycle, since it only takes a few tens of cycles to fetch from cache, compared to possible hundreds or more of cycles to fetch from RAM.

Number of cores – the cores there are, the more instructions that can be processed at a time. For example, a dual core can execute 2 instructions at a time, whereas single core can execute 1 instruction at a time only. Also, for example, pipelining is more efficient with a dual core for example, since instead of cramming every instruction into 1 core, you have 2 cores to use, so you can pipeline double the amount. The more cores basically just mean the more processors you have, reducing stress on each processor.

1. Describe what is meant by each of the following terms, giving an example of how each is used in a computer system.

(a) parallel processing [2]

Where multiple instructions are executed at the same time and processed at same time. For example, in a quad core processor, you can have all 4 processors working on 4 different instructions at the same time instead of 1 processor working on an instruction, then working on another once it is done with the current one.

(b) pipelining [2]

Where as soon as one stage processing is done, it is kept busy by having another instruction fed into it. For example, when instruction1 is fetched, then it gets moved to the decoded stage, before instruction1 is fully processed and executed, as soon as it gets moved to decode phase, instruction2 gets moved into the fetch phase and processor already begins to fetch instruction2 before instruction 1 finishes executing. Basically, overlapping instructions in different stages. This is used for example in, a single core processor, where you only have one processor, but need it to do a lot so instead of sequential execution where instructions are executed 1 by 1, immediately start the processing of the next instruction as soon as a stage is complete for the previous instruction, so that the processor is always at work.

3. Isobel needs to work on a report. She loads a word processing package and the report, finding that it takes a long time. She writes the report and closes the word processor, before remembering something else she should have included. She reloads her document, this time finding that it loads in a fraction of the time that it took before.

Explain a possible reason for this. [2]

Could be because of the cache, on the first load, the computer has never seen the data before and instructions, so it takes some time to understand what the report says. On the second time the content on the report that has been repeated frequently, like maybe some words, have already been written in the cache, so loading the report takes a lot less time because most of it is already familiar data to the processor. It can retrieve it easily from cache instead of searching for it through main memory.

[Total 12 marks]