Parallel Programming Skills

Foundation

➢ Identifying the components on the raspberry PI B+
- The raspberry PI B+ has the following components: USB ports, Ethernet (and the controller) , HDMI port, CPU/RAM, Power port, the Display

➢ How many cores does the Raspberry Pi's B+ CPU have?
- The raspberry Pi has four cores

➢ List three main differences between X86 (CISC) and ARM Raspberry PI (RISC). Justify you answer and use your own words (do not copy and paste)

X86	ARM Raspberry PI
Differences in number of registers, for instance, in ARM, when adding two registers(add r1,r2) we need another register to store the result in, therefore,(add r1(could be any register),r1,r2)	In Intel we don't need additional register to store the result, for instance, when we add r1,r2 the result is stored in r1.
Differences in instruction set. The X86 (Complex instruction set computer) has a complex instruction set, which utilizes multiple CPU cycles to execute each instruction	The ARM (Reduced instruction set computer) is reduced instruction set and mostly the instructions are executed in one in one clock cycle
Differences in storage format. The X86 uses little-endian, which stores from right to left	The ARM uses BI-endian, which can read both ways (right to left or left to right)

➢ What is the difference between sequential and parallel computation and identify the practical significance of each?
- Parallel computation is the use of multiple compute resources to solve a computational problem, whereas the sequential computation is the use of a single compute resource to solve a computational problem.

Sequential computation	Parallel computation
Tasks are assigned(executed) to only one processor	Tasks are assigned(executed) to difference processors
Requires more time	Requires less time
Involves only one processor	Involve multiple processors

➢ Identify the basic form of data and task parallelism in computational problems.
- Data parallelism is a different kind of parallelism in which the same computation is applied to multiple data items. It is a general term that actually applies to any form of parallelism in which the amount of work grows with the size of the problem.

- Task parallelism is a kind of parallelism that applies to solutions where parallelism is organized around the functions to be performed rather than the data. It focuses on distributing parallel execution threads across parallel computing nodes. In the most general case, each of the threads of a Task-Parallel system can be doing completely different tasks but co-ordinating to solve a specific problem.

➢ Explain the differences between processes and threads.

Processes	Threads
They do not share memory with each other	They share the common memory of the processor they belong to
They are heavyweight operations	They are lighter weight operations
Each process has its own memory space	They use the memory of the process they belong to

➢ What is OpenMP and what is OpenMP pragmas?
- OpenMP is an application programming interface consisting of compiler directives and library routines for high level parallelism in C, C++ and Fortran programs.
- OpenMP pragmas are compiler directives that enable the compilers to generate threaded code. They control how the program works.

➢ What applications benefit from multi-core (list four)?
- Multimedia applications
- Scientific applications
- Web servers
- Database servers

➢ - (4p) Why Multicore? (why not single core, list four)
- With multicore, the OS can execute more processes at once, while the single core only operates on one process at a time.
- Multicores run in parallel for quickness, flexibility and time saving while a single core is slow and time consuming especially when writing a large program.
- Multicores enable pipelining of circuits.
- Multicore is more efficient than single core system as it depends less on any single core while in single core system breakage on single core leads to total breakage of the system.