Assignment 3

Generally speaking, Cloud computing is built around the ideas of virtualisation, elasticity and ‘pay as you go’ computing:

(a) [5 Marks] Outline why this model leads to more resources being available for innovation or development in an existing company. Please limit your answer to a short paragraph.

Prior to Cloud computing, companies required a large amount of IT infrastructure/maintenance to provide their services for example in land, capital and skilled workers. This was extremely costly created great difficulty for financial planners and service providers themselves (Overestimation resulted in surplus of unused resources to maintain/Underestimations would lead to loss of revenue as it effected their ability to provide services promised to consumers).
Cloud computing is a solution to these issues by providing a ‘pay as you go’ model for server time and processing power. This allow companies to pay for what they need eliminating budget/service concerns and meant companies no longer had to maintain their own IT infrastructure.
Where companies were able to reduce IT cost, these funds could be reallocated to research/ development which allows for innovation and cheaper operating charges (higher profitability).

(b) [5 Marks] Explain why this was also an ideal model for start-ups and enabled their growth. Please limit your answer to a short paragraph.

Without the costly pre-requisite of owning/maintaining IT infrastructure, start-up companies were able to compete/enter the market using the affordable alternative of cloud computing which allows for pay-as-you go, lower rates of investments for expansion and ease of contraction (easily scalable).
The elasticity of cloud provisioning allows start-ups to more easily manage their growth/cost based on their environment. Additionally, costs saved on IT infrastructure is now available to invest in innovation and R&D.
This places start-up in a more favourable financial position than if they were to implement traditional IT systems.

(c) [5 Marks] The Cloud model of elastic compute and the appearance of infinite resources is somewhat misleading - what is the limiting factor? Please limit your answer to a short paragraph.

Cloud computing allows companies to operate without self-owned physical infrastructure, this doesn’t mean IT infrastructure isn’t needed
Cloud computing still requires physical infrastructure to be set up somewhere in the real world. Thus there is a finite amount of Cloud computing infrastructure constructed and available to meet consumer demand.
The elasticity model presents the notion that system resources will always be available. However, If there is a greater demand for Cloud resources than supply (based on available physical structures), the shortage would break the idea of infinite resources.

(d) [8 Marks] There are plenty of examples of successful companies that were early adopters of the cloud. Some of these companies have changed the way in which we view or interact with the world, and simply would not have happened without cloud computing. Choose at least one example company and firstly, explain how it changed everything, and secondly what aspects of the cloud were critical to that success. Please limit your answer to a moderate length paragraph

Netflix:
Prior to 2007 Netflix operated as a “rent-by-mail DVD service that used a pay-per-rental model However after the adoption of Cloud in 2007, Netflix moved their business model online - allowing consumers to stream online media to their local devices. Innovation meant that consumers could watch these films on their personal computers without the need for various devices (more convenient and accessible).

Without Cloud computing Netflix would have needed to increase their IT infrastructure exponentially to meet rising demand; however, this was difficult as servers couldn't be constructed at the same rate of incoming subscriptions.
Cloud computing offered a solution where Netflix could use Cloud-based servers instead of their own local data centres. This allowed Netflix to meet demand by rapidly increasing its system resources.

Flynn's taxonomy includes SISD, SIMD and MIMD models of parallelism (we will quietly ignore MISD), suggest an example problem that is best tackled by each of these models, and comment why this is so:

(a) [3 Marks] SISD

SISD is a computer architecture in which a single uni-core processor executes a single instruction stream, to operate on data stored in a single memory. A non-parallel process that is most compatible with non-parallel programs.
>An example of when SISD chipsets are useful is any non-threaded program that requires previous information to be calculated before the following calculation such as a recursive graphing program or any other single instruction program.
>Currently, SISD chipsets are most often only found on legacy machines. This is because were significantly cheaper to produce in the early days of computing. However they are difficult to source in modern day as single-core processes are outperformed by multi-core processors produced with faster clock speeds.

(b) [3 Marks] SIMD

SIMD represents an organisation that includes many processing units under the supervision of a common control unit. All processors receive the same instruction from the control unit but operate on different items of data (multiple parallel operations on many different data points. However, all the operations must remain the exact same).
>An example of when SIMD chipsets are useful is changing the brightness of an image by changing all the light-level data points by the same specified value.
> SIMD systems were mainly used in vector supercomputers in the past (1970s) as they operated on a single "vector" of data with a single instruction.

(c) [3 Marks] MIMD

In computing, MIMD is a technique employed to achieve parallelism.Composed of multi-threaded, multi-core processors. MIMD processors can individually execute different instructions on different tasks at the same time (split one enormous task into smaller chunks then re-compile the finished or "Solved" task).
This means MIMD is compatible with complex and large programs such as cloud-based video game services or running film production rendering software.
>An example of when MIMD chipsets are useful is where multiple threads handle different connecting client interfaces. The advantage that MIMD provides is that one particular client wouldn't rely on another, meaning multiple processors could solve each task non simultaneously.
>MIMD is common in larger data centres but not often found on a singular home pc.

3. While Amdahl's Law probably ought be 'gracefully' retired, it still offers us some insights into parallelism - it is also endemic, and therefore you need to be aware of it and its legacy. Just don't use it "authoritatively", OK!

(a) [4 Marks] Define the term "embarrassingly parallel", and state what p would be for such a program wrt to Amdahl's law?

A program is "Embarrassingly parallel" if it is effortless to divide into smaller individual tasks that computing can solve in parallel.
Amdahl's law states, "if P is the proportion of a system or program that can be made parallel, then the maximum speedup that can be achieved with N processors is: (1 / (1−p)+ p n)". Here the P-

value refers to the portion of the program that computing can solve in parallel. If the program is “Embarrassingly parallel”, the P-value would be precisely one or very close to it.

(b) [4 Marks] Define the term “Inherently serial”, and state what p would be for such a program wrt to Amdahl’s law?

A program is “Inherently serial” if the problem can’t be divided into several smaller individual tasks to solve in parallel. This inability to be solved in parallel is generally because they require previous steps to be solved in order to calculate the next one. Effectively this is the opposite of a program being “Embarrassingly parallel”. Because computing cannot solve the program in parallel, we can expect the P-value from Ahmdahl’s law to be precisely zero or very close.

REFERENCE

- https://en.wikipedia.org/wiki/MIMD#:~:text=In%20computing%2C%20MIMD%20(multiple%20instruction,that%20function%20asynchronously%20and%20independently.&text=MIMD%20machines%20can%20be%20of%20either%20shared%20memory%20or%20distributed%20memory%20categories.
- https://en.wikipedia.org/wiki/SIMD#:~:text=Single%20instruction%2C%20multiple%20data%20(SIMD,parallel%20processing%20in%20Flynn's%20taxonomy.&text=Such%20machines%20exploit%20data%20level,(just%20with%20different%20data).
- https://www.sciencedirect.com/topics/computer-science/single-instruction-single-data#:~:text=Single%20Instruction%2C%20Single%20Data%20stream,a%20PC%20(single%20core).&text=Single%20Instruction%2C%20Multiple%20Data%20Stream,by%20replicating%20the%20computing%20hardware.