1. Critically evaluate the functionality of different types of software, i.e., operating system, utility programmes, languages and applications.

Operating systems are software programmes that manage computer hardware and software resources while providing standard services to computer programmes. They manage memory, control input/output operations, and work with file systems. An operating system's functionality is critical to a computer's overall functionality. A good operating system should be dependable, simple to use, secure, and provide the functionality required to support applications. Different operating systems have different strengths and weaknesses; their needs and preferences determine the best one for the user.

Utility programmes are software applications that add functionality to the operating system. They are intended to boost a computer's performance and functionality. Examples of utility programmes include antivirus software, disc clean-up tools, and system optimisation software. Utility programmes vary in functionality but are generally used to improve system performance and security.

Programming languages are computer programmes that enable programmers to create code for applications and systems. They are also used to develop the algorithms that underpin the operation of software applications. Different programming languages have different functionalities, and which one to use depends on the project requirements and the programmer's level of expertise.

Applications are software programmes that are intended to carry out specific tasks. They can be simple or complex desktop or mobile applications. An application's functionality is determined by its design and purpose. Some applications are intended to be entertaining, while others are intended to be productive. The user's needs and preferences determine the best application to use.

Finally, the functionality of various types of software is determined by their design and purpose. Operating systems are critical to a computer's overall functionality, whereas utility programmes are used to improve system performance and security. Programming languages allow programmers to create code for applications and systems, whereas applications are designed to accomplish specific tasks. The right software is determined by the user's needs and preferences and the project's requirements.

Reference:

Power, R. & Ford, R. (2009). *Operating System Fundamentals.* [online] ResearchGate. Available at:

https://www.researchgate.net/publication/339413127_Operating_System_Fundamenta