What is a secure programming language?

1) What factors determine whether a programming language is secure or not?

   According to Cifuentes & Bierman (2019), all mainstream languages today are inherently insecure as they are all susceptible to common vulnerabilities such as buffer errors, injection errors and information leak errors. While it is true that some languages may be more vulnerable than others, good coding practices must be taken with all languages to reduce the risk of successful attacks.

2) Could Python be classed as a secure language?

   Python, as one of the mainstream languages referred to above, is susceptible to vulnerabilities and therefore can't be considered inherently secure. According to Pillai (2017), python is susceptible to overflow errors, serialization issues and other common vulnerabilities. However, steps can be taken to mitigate these vulnerabilities, and with the good, secure coding practices Python can be used to build secure software.

3) Python would be a better language to create operating systems than C. Discuss.
   Python is a high-level interpreted language and is therefore significantly slower and less efficient to run. A basic requirement of an operating system is to efficiently interact on a low level of abstraction with the kernel. On the other hand, C works at a lower level of abstraction and would therefore be much more efficient for this use case.

References

Cifuentes, C. & Bierman, G. (2019) What is a secure programming language? *3rd Summit on Advances in Programming Languages (SNAPL 2019)* 3:1-15

Pillai, A. (2017) Software Architecture with Python : Architect and Design Highly Scalable, Robust, Clean, and Highly Performant Applications in Python. Birmingham, UK: Packt Publishing.