**Introduction**

An Imperative programming (also known as procedural programming) style is one that uses statements to change what is needed within the program. By writing in an imperative style you are telling the computer exactly what you want it to do step by step to accomplish the required goal. Imperative programming is of such importance as the hardware implementation of most computers is imperative. From a low level perspective (x86 assembly language) the program is defined by the memory contents and the statements tell the computer what to do, example of this is “ mov eax, 3”. In higher level languages variables are used to do the same thing “ x = 3”

In comparison Object oriented programming paradigm organised around objects rather than actions. Languages that support OOP uses inheritance for code to be reused. One major advantage of an OOP style over an imperative style is that it allowed developers to create modules that don’t need to be changed when a new type of object is created. The developer can just create a new object which inherits many of its features from other classes for example with a chess application the class ‘Queen’ will inherit many of its features from class ‘Piece’. This makes it much easier to change.

**Comparison**

In