**Week 4 and 5**

/\*

3 ways to be a programmer with productivity

-Sideway

-Slowlane: use more time for better productivity, performance is not good.

-Fastlane: think about not spending time for programming.

\*/

How to use class in c++ (namespace -> structure -> class, user defined types)

Class, object, instance

Access modifiers: public, private, protected

Pseudo-code to code (decomposing complex tasks into simpler ones)

To recognize when to use **value** and **reference** parameters

To develop strategies for decomposing complex tasks into simpler ones

To be able to implement your own classes

To master the separation of **interface** and **implementation**

To understand the concept of **encapsulation**

To design and implement **accessor** and **mutator** member functions

Implicit parameter vs explicit parameter

To understand the concepts of **inheritance** and **polymorphism**

To learn how inherence is a tool for **code reuse**

To learn how to call base-class constructors and member functions

To understand the difference between static and **dynamic binding**

To be able to implement dynamic binding with **virtual functions**

**-Note:**

**Virtual functions with over-roading (between mother and child class)**

**(over-roading != over-loading)**