

Programming (2)

This course concerns the object-orientation concepts and teaches students how to approach problems by breaking them down into interrelated objects, then implementing them. Topics include: Different data types (simple and compound), Abstract data type (ADT) and abstract classes, Static classes, Inner classes, finals, implementation and application. Dynamically allocated storage, arrays, strings, Pointers and their use. Object-oriented design, Decomposition into objects carrying state and having behavior, Class-hierarchy design for modeling, Subclasses, reusability and inheritance, method overriding, Dynamic dispatch and definition of method-call. Subtyping, Subtype polymorphism, implicit Upcasts in typed languages, Notion of behavioral replacement, subtypes acting like supertypes. Relationship between subtyping and inheritance. Object-oriented idioms for encapsulation. Privacy and visibility of class members. Interfaces revealing only method signatures, Using collection classes, iterators, and other common library components.