CS20202: Software Engineering - Introduction

Intructors: Abir Das and Jibesh Patra

Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

{*abir, jibesh*}*@cse.iitkgp.ac.in*

Slides taken from NPTEL course on Programming in C++

by Prof. Partha Pratim Das

Software Engineering

- A crucial component of Software Engineering is software development – writing programs.
- Programming paradigms help in writing programs:
  - Procedural programming: Data and procedures are separate. Procedures act on data to achieve functionality.
  - Functional programming: Functions are first class members, and are composed with each other to achieve a desired function which is then applied on the input data.
  - Object-oriented programming: Data and related functions are grouped into objects. Objects interact with each other to achieve the desired functionality.

Object-oriented programming

- Most suited for development of large and complex software systems.
- Many popular programming e.g. C++, Java, Python, etc support this style.
  - These languages also support procedural, and functional programming to some extent.
- C++ is popular for:
  - System programming.
  - Writing high performance programs.
  - Feature rich – good for learning tools.
  - (Negative) Programs can be large.

OOP concepts

- Objects and Classes – user defined data types.
- Encapsulation – Data hiding – Data is hidden inside objects and accessed through member functions or methods.
- Composition – objects can contain other objects
- Inheritance – objects can borrow properties of other objects
- Polymorphism – the function executed depends on the type of objects.

OOP with C++

- Tentative list of features and their applicability:
  - Procedural enhancements in c++ over c
  - Classes
  - Overloading
  - Inheritance
  - Type casting
  - Exceptions
  - Templates

List of Topics

- Introduction, Logistics, Procedural Extensions over C in C++
- Object-oriented Programming in C++
- Software Development Life Cycle (SDLC)
- Software Testing and Maintenance
- Design Patterns
- Some selected topics in Software Engineering (depending on time and availibility)

Marks Distribution and Logistics

- Class tests - 20% marks
  - Tentative Dates: 31^{st} Jan and 4^{th} April
- Mid-sem - 30% marks
- End-sem - 50% marks
- No graded Assignments, but practice problems - important to solve yourself

Logistics:

- Course website (`https://jibesh.com/teaching_se_2025.html`) will provide announcements and course materials
- Students with odd Roll Numbers will go to NC343 and those with even Roll Numbers should go to NC344.
- Class tests will be held concurrently at lab.

References

- Roger S Pressman, **Software Engineering: A Practitioner's Approach**, 7th Edition, McGraw Hill Education, 2009.
- Rajib Mall, **Fundamentals of Software Engineering**, Prentice Hall India, 2014
- Bjarne Stroustrup, **The C++ Programming Language**, 4th Edition, Addison-Wesley, 2013
- Erich Gamma, Richard Helm, Ralph Johnson, & John Vlissides, **Design Patterns: Elements of Reusable Object-Oriented Software**, Addison Wesley, 1994

Thanks !
Questions ?