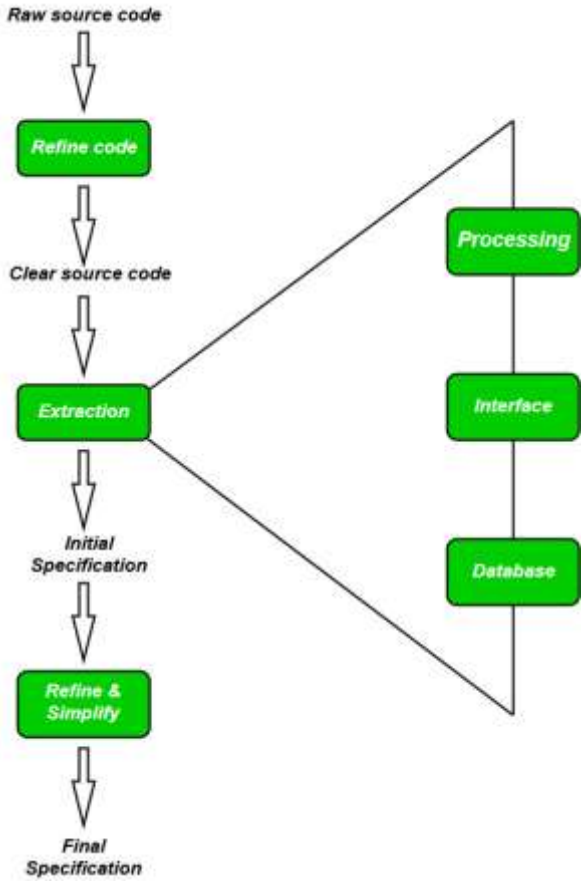
Unit: IV
Lecture: 10
Reverse Engineering

Software Reverse Engineering is a process of recovering the design, requirement specifications and functions of a product from an analysis of its code.
It builds a program database and generates information from this.
The purpose of reverse engineering is to facilitate the maintenance work by improving the understandability of a system and to produce the necessary documents for a legacy system.

Reverse Engineering Goals:

- Cope with Complexity.
- Recover lost information.
- Detect side effects.
- Synthesize higher abstraction.
- Facilitate Reuse.



Steps of Software Reverse Engineering:

1. **Collection Information:**
   This step focuses on collecting all possible information (i.e., source design documents etc.) about the software.

2. **Examining the information:**
   The information collected in step-1 as studied so as to get familiar with the system.

3. **Extracting the structure:**
   This step concerns with identification of program structure in the form of structure chart where each node corresponds to some routine.

4. **Recording the functionality:**
   During this step processing details of each module of the structure, charts are recorded using structured language like decision table, etc.

5. **Recording data flow:**
   From the information extracted in step-3 and step-4, set of data flow diagrams are derived to show the flow of data among the processes.

6. **Recording control flow:**
   High level control structure of the software is recorded.

7. **Review extracted design:**
   Design document extracted is reviewed several times to ensure consistency and correctness. It also ensures that the design represents the program.

8. **Generate documentation:**
   Finally, in this step, the complete documentation including SRS, design document, history, overview, etc. are recorded for future use.

Some of tools for reverse engineering are given below:

- **CIAO and CIA:** A graphical navigator for software and web repositories along with a collection of Reverse Engineering tools.
- **Rigi:** A visual software understanding tool.
- **Bunch:** A software clustering/modularization tool.
- **GEN++:** An application generator to support development of analysis tools for the C++ language.