Programming Language Generations

Lecture 1

Text Books

- 1. C++ How to Program by H.M.Deitel and P.J.Deitel
- 2. Turbo C++ by Robert Lafore

Recommended Books

- 1. Schaum's Outline of Programming with C++ by John R. Hubbard
- 2. Lecture Notes

Programming Language Generations

Generations of programming languages are categorized into 5 categories

- **First Generation of programming Language or 1GL**
  - string of 0s and 1s
- **Second Generation of programming Language or 2GL**
  - sometimes called Assembly language

    e.g. ADD 12, 8
  - An assembler converts the assembly language statements into machine language.

Programming Language Generations

- **Third Generation of Programming Language or 3GL**
  - Called High-Level programming languages such as C/C++, Pascal or Java etc
  - Near to English
  - Compiled type languages i.e. C/C++, Pascal, COBOL and Fortran etc
  - Interpreter based languages i.e. QBasic, GW-Basic and Visual Basic etc are 3GL.

Compiler vs Interpreter

Compiler	Interpreter
Compiler takes entire program as an input	Interpreter takes single instruction as an input
Intermediate object code is generated	No. Intermediate object code is generated
Memory Requirements More	Memory Requirements Less
Program need to be compiled every time	Every time higher level program is converted to lower level program
Errors are displayed after entire program is checked	Errors are displayed after every instruction interpreted
e.g. C/C++ Compiler	e.g. Basic

Programming Language Generations

- **Fourth Generation of Programming Language or 4GL**
  - designed to be closer to natural language than a 3GL.
  - for accessing databases are often described as 4GLs

  e.g.

  SELECT **NAME FROM EMPLOYEES WHERE SALARY > $7000**

- **Fifth Generation of Programming Language or 5GL**
  - use a visual or graphical development interface to create source language
  - IBM, Microsoft, Borland etc
  - Visual Studio.NET, JBuilder, NetBeans etc.