1. Explain control & data flow analysis for optimizing transformation.
2. Explain frequency reduction & strength reduction with examples.
3. How partial results are handled in compilation of expressions? Show code generation actions for the expression a*b+c*d by describing code generation routine.
4. Discuss major issues in code generation for expression.
5. What are triples, quadruples and indirect triples? Explain.
6. Discuss the PL features that contribute to the semantic gap between PL domain & execution domain which is bridged by compiler.
7. Explain different optimizing transformation used in compilers.
8. Explain triple, quadruples & expression trees in intermediate code expression.
9. List major issues in code generation for expression. Explain operand & register descriptor with examples.
10. What is the use of code optimization? How it is achieved? Explain optimizing compiler & give different optimizing transformation used in code optimization.
11. List & explain the PL features used in implementation of aspects of compilation. Explain any two in details.
12. Define program flow graph. Explain control flow analysis in global optimization.
13. Explain local optimization using value numbers. Give limitations of local optimization.
14. Give the aspects of compilation. Explain data types & scope rules, the PL features used to implement the aspects of compilation.
15. Explain data flow analysis in global optimization.
16. Explain the following optimizing transformation with examples
    i. Common sub expression elimination
    ii. Dead code elimination
    iii. Frequency reduction
    iv. Strength reduction
    v. Compile time evaluation
17. Compare between local & global optimization technique for code.
18. Explain operand & register descriptor with example.
19. Explain aspects of compilation.