

Semester

VI

Subject Code

CS603 (C)

Subject Name

Compiler Design

Unit-5

Topic: Peephole Optimisation



As Per

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

(www.rgpv.ac.in)

New Scheme Based on AICTE Flexible Curricula Computer Science and Engineering



Mr Sandeep Wadekar

Assistant Professor

Department of Computer Science and Engineering

SAGAR INSTITUTE OF RESEARCH AND TECHNOLOGY BHOPAL



PEEPHOLE OPTIMIZATION

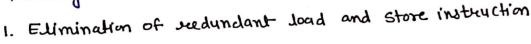
Dec-2002, Dec-2003, June-2004 Dec-2004, 05, 06
June-2007 Dec-2007 Dec-2008 Dec-2010, JuneDec-2011, Dec-2013, Dec-2015, 16, 17, 18, 19

the the code is generated statement wise then it consist several redundancies and suboptional construct. In order to remove such redundancies in a code for increasing efficiency we need optimization over a target code. Peephole optimization is a technique by which we can locally optimise a code.

In Peephole Optimization we replace shorter and slower sequence with faster sequences according to the available possibilities.

Characteristics of Peophole Optimization -

The Peephole Optimization can be applied on the target code using following characteristics -



- 2. Elimination of multiple jumps. / How to control optimization
- 3. Elimination of unreachable code.
- 4. Algebraic simplifications.
- 5. Reduction for strength improvement.
- 6. Use of machine idioms.

1. Elimination of Redundant load and store instruction -

Example of reedyndant loads and store like the instruction sequence.

- (a) MOV Ro, 9
- (b) MOV a, Ro

In the above, we can eliminate second one (b).

Since 'a' is already in 'Ro



Mr Sandeep Wadekar



2. Elimination of Multiple jumps -

Multiple unnecessary jumps are avoided to increase efficiency and speed of execution.

Example-

3. Elimination of unreachable code-

Unreachable means the code which does not lies in the instructions boundaries. Any unlabeled instructions which immediately appears after the unconditional jumps can be removed. For debugging purpose unreachability always avoided.

In above example, "If statement" will never get executed, hence can be eliminated.



Mr Sandeep Wadekar



4. Algebraik Simplification -

Useless algebraic calculations always waste memory in the program. Such instructions always be simplified to increase execution speed in a code.

Example-

or

There are generated by intermediak code generation-algorithm and can be eliminated through peophole optimization.

5. Reduction of Strength Improvement -



To increase efficiency of the code we can replace costliete instruction with the cheaper instruction.

Example-

- 1. x=Y * * 2 can be replaced with x=Y * Y
- 2. x2 is cheaper than X+X
- 3. Addition and <u>Subtraction</u> is cheaper than multiplication and <u>division</u>.

6. Use of Machine Idims -

Some machines having specific hardware instruction to implement cortain operations efficiently. Some machine contain alto increment and decrements modes with the help of counters in addressing modes. These machine idioms are very helpful to save the memory.

Example-

- · a= a+1
- q = q-1

