SNS COLLEGE OF TECHNOLOGY

(Autonomous)

COIMBATORE-35

Peephole Optimization

Peephole Optimization

Code generator creates the target code

- Target code contains
- Redundant Instruction
- Sub optimal code
- Poor Quality

Requirement-Quality of code should be improved

Soln-Applying optimizing Transformations

Characteristics of peephole optimizations:

- Redundant-instructions elimination
- Flow-of-control optimizations
- Algebraic simplifications
- Reduction in Strength
- Use of machine idioms
- Unreachable

Peephole Optimization

Examples

Elimination of redundant loads and stores

```
r2 := r1 + 5                    r2 := r1 + 5
i := r2          becomes        i := r2
r3 := i                         r4 := r2 × 3
r4 := r3 × 3
```

Constant folding

```
r2 := 3 × 2      becomes        r2 := 6
```

Peephole Optimization

Constant propagation

r2 := 4 r3 := r1 + r2 r2 := ...	becomes	r2 := 4 r3 := r1 + 4 r2 := ...	and then	r3 := r1 + 4 r2 := ...
r2 := 4 r3 := r1 + r2 r3 := *r3	becomes	r3 := r1 + 4 r3 := *r3	and then	r3 := *(r1+4)
r1 := 3 r2 := r1 × 2	becomes	r1 := 3 r2 := 3 × 2	and then	r1 := 3 r2 := 6

Peephole Optimization

Copy propagation

r2 := r1 r3 := r1 + r2 r2 := 5	becomes	r2 := r1 r3 := r1 + r1 r2 := 5	and then	r3 := r1 + r1 r2 := 5

Strength reduction

r1 := r2 x 2	becomes	r1 := r2 + r2	or	r1 := r2 << 1
r1 := r2 / 2	becomes	r1 := r2 >> 1		
r1 := r2 x 0	becomes	r1 := 0		

Summarization