1. What are the sources of optimization for basic blocks in code?

a) Loop unrolling
b) Instruction scheduling
c) Register allocation
d) All of the above
Answer: d) All of the above
Explanation: Basic block optimizations involve various techniques like loop unrolling to reduce loop overhead, instruction scheduling to improve instruction throughput, and register allocation to minimize memory accesses.

2. Which optimization technique is primarily used for optimizing loops in flow graphs?

a) Dead code elimination
b) Loop unrolling
c) Common subexpression elimination
d) Data flow analysis
Answer: b) Loop unrolling
Explanation: Loop unrolling is a technique used to optimize loops by reducing loop control overhead and exposing more instruction-level parallelism.

3. What does dead code elimination aim to accomplish in code optimization?

a) Removing unreachable code
b) Optimizing loops
c) Minimizing register pressure
d) Improving memory access patterns

Answer: a) Removing unreachable code
Explanation: Dead code elimination identifies and removes code that cannot be executed, thus reducing code size and potentially improving runtime performance.

4. Which of the following is a key aspect of loop optimization?

a) Reducing loop iterations
b) Introducing more loop nests
c) Increasing loop overhead
d) Improving data locality
Answer: d) Improving data locality
Explanation: Loop optimization aims to improve performance by enhancing data locality, reducing loop overhead, and maximizing instruction-level parallelism.

5. What is the primary goal of global data flow analysis in code optimization?

a) Identifying unused variables
b) Minimizing control flow
c) Analyzing variable usage across the entire program
d) Improving cache utilization
Answer: c) Analyzing variable usage across the entire program
Explanation: Global data flow analysis examines how variables are used across the entire program to optimize register allocation, identify dead code, and improve overall performance.

6. Which type of transformation is commonly used to improve code performance by rearranging instructions without changing program semantics?

a) Loop unrolling
b) Instruction scheduling
c) Dead code elimination
d) Common subexpression elimination
Answer: b) Instruction scheduling
Explanation: Instruction scheduling involves rearranging instructions to minimize pipeline stalls and improve instruction throughput without altering the program's functionality.

7. What is the purpose of symbolic debugging of optimized code?

a) Identifying performance bottlenecks
b) Analyzing memory usage
c) Improving code readability
d) Debugging without losing optimization benefits
Answer: d) Debugging without losing optimization benefits
Explanation: Symbolic debugging allows developers to debug optimized code while preserving the optimizations applied, enabling efficient debugging without sacrificing performance.

8. Which optimization technique focuses on identifying and eliminating redundant computations?

a) Loop unrolling
b) Common subexpression elimination
c) Register allocation
d) Dead code elimination
Answer: b) Common subexpression elimination

Explanation: Common subexpression elimination removes redundant computations by identifying expressions that have already been computed and stored, thereby reducing redundant computations.

9. What is the primary purpose of loop unrolling?

a) Decreasing code size
b) Reducing loop overhead
c) Introducing more loops
d) Minimizing data dependencies
Answer: b) Reducing loop overhead
Explanation: Loop unrolling aims to reduce loop overhead by executing multiple loop iterations in a single iteration, thereby improving performance.

10. Which optimization technique focuses on utilizing CPU registers efficiently?

a) Loop unrolling
b) Instruction scheduling
c) Register allocation
d) Dead code elimination
Answer: c) Register allocation
Explanation: Register allocation assigns variables to CPU registers to minimize memory accesses, reduce register pressure, and improve overall performance.

11. What does loop fusion involve in optimizing loops?

a) Combining multiple loops into a single loop
b) Breaking a loop into smaller loops

c) Reversing loop iterations
d) Introducing more loop nests
Answer: a) Combining multiple loops into a single loop
Explanation: Loop fusion combines multiple loops into a single loop to reduce loop overhead and improve data locality.

12. Which optimization technique focuses on rearranging code to exploit instruction-level parallelism?

a) Instruction scheduling
b) Loop unrolling
c) Dead code elimination
d) Common subexpression elimination
Answer: a) Instruction scheduling
Explanation: Instruction scheduling rearranges code to exploit instruction-level parallelism and minimize pipeline stalls, thus improving performance.

13. What is the primary goal of loop-invariant code motion?

a) Reducing loop iterations
b) Moving computations outside the loop
c) Increasing loop overhead
d) Introducing more loop nests
Answer: b) Moving computations outside the loop
Explanation: Loop-invariant code motion moves computations that do not change during loop execution outside the loop to reduce redundant computations and improve performance.

14. Which optimization technique aims to eliminate code that does not affect program output?

a) Loop unrolling
b) Dead code elimination
c) Register allocation
d) Instruction scheduling
Answer: b) Dead code elimination
Explanation: Dead code elimination identifies and removes code that does not affect program output, reducing code size and potentially improving performance.

15. What is the primary purpose of loop peeling in loop optimization?

a) Reducing loop overhead
b) Increasing loop iterations
c) Introducing more loop nests
d) Improving data locality
Answer: a) Reducing loop overhead
Explanation: Loop peeling involves removing the first few iterations of a loop to reduce loop overhead and improve performance.

16. Which optimization technique focuses on minimizing memory access patterns?

a) Loop unrolling
b) Register allocation
c) Dead code elimination
d) Data prefetching

Answer: d) Data prefetching
Explanation: Data prefetching aims to minimize memory access latency by fetching data into cache before it is actually needed, thereby improving performance.

17. What does loop interchange involve in loop optimization?

a) Reversing loop iterations
b) Exchanging the order of nested loops
c) Combining multiple loops into a single loop
d) Introducing more loop nests
Answer: b) Exchanging the order of nested loops
Explanation: Loop interchange exchanges the order of nested loops to improve data locality and reduce loop overhead, thereby enhancing performance.

18. Which optimization technique aims to reduce the number of instructions executed inside loops?

a) Loop unrolling
b) Loop fusion
c) Loop-invariant code motion
d) Loop interchange
Answer: c) Loop-invariant code motion
Explanation: Loop-invariant code motion moves computations outside loops to reduce the number of instructions executed inside loops, thereby improving performance.

19. What is the primary purpose of loop vectorization?

a) Minimizing loop overhead

b) Introducing more loop nests
c) Exploiting SIMD instructions for parallel execution
d) Eliminating redundant computations
Answer: c) Exploiting SIMD instructions for parallel execution
Explanation: Loop vectorization transforms scalar operations into SIMD operations to exploit parallelism and improve performance.

20. Which optimization technique focuses on reducing the number of conditional branches in code?

a) Loop unrolling

b) Dead code elimination
c) Loop fusion
d) Control flow optimization
Answer: d) Control flow optimization
Explanation: Control flow optimization aims to reduce the number of conditional branches in code to improve branch prediction accuracy and overall performance.

Related posts:

1. Introduction to Information Security
2. Introduction to Information Security MCQ
3. Introduction to Information Security MCQ
4. Symmetric Key Cryptography MCQ
5. Asymmetric Key Cryptography MCQ
6. Authentication & Integrity MCQ
7. E-mail, IP and Web Security MCQ