

Wisdom in Optimization: A Learning Activity

Computer Science, Creation Theology, and Stewardship

1. LEARNING OBJECTIVES

- Understand algorithm efficiency from a computational perspective
 - Analyze the ethical and theological dimensions of optimization
 - Reflect on stewardship as a framework for technological design
 - Recognize how design choices reflect values and worldviews
-

2. THE DOCTRINE OF CREATION AND STEWARDSHIP

This assignment is grounded in the Christian doctrine of creation and the concept of stewardship. The Christian doctrine of creation teaches that God is the Creator of all things, and that humans, created in God's image, are called to be stewards—not exploiters—of creation. This is not merely about environmental care; it extends to how we design and deploy all systems, including computational systems. For this reason, we will start by defining some **key theological principles** that inform our understanding of efficiency and optimization.

1. God Creates with Wisdom and Order

“How many are your works, Lord! In wisdom you made them all” (Psalm 104:24). God designed the cosmos with intricate patterns, relationships, and efficiency. Creation itself reflects divine wisdom in how systems are organized and sustained.

2. Humans Bear God's Image and Share in His Responsibility

Humans are made in God's image (Genesis 1:26-28) and are called to “tend the garden” and “subdue the earth”—but the imagery is of a *gardener*, not a miner. A gardener must show “great sensitivity to resources available and the bounty of God.” Our authority is limited; we are stewards, not ultimate owners. (Pobee, 1985)

3. All Authority is Exercised Under God's Authority

When humans rule over creation, they do so as delegated stewards under God's sovereignty. We are “neither omniscient nor all-powerful” and “do not have power and freedom to perform at will.” Our power and creativity are circumscribed by responsibility. (Pobee, 1985)

4. Creation is Ordered and Good

God created a cosmos—a world system marked by order, beauty, and internal relationships held together by design. Each element has been assigned “its place and proper name” by the Creator. This ordering reflects God's nature: “Whatever in

creation manifests goodness, truth, and beauty is a pointer to and reflection of what its Creator God essentially and eternally is.” (Gunton, 2001)

5. Technological Development Must Be Responsible

“If creation is by the decisive will of God, then the use of the created order must take seriously the intention and will of God in creating the world. This means that **technological and scientific invention and use must be responsible to the moral order.**” (Pobee, 1985) We are accountable for how we design and deploy our technologies.

2.a. *Inefficiency as a Form of Disorder*

From a Creation perspective, inefficiency represents several failures:

1. **Disorder** (Against God’s Creative Order)

God created a cosmos—ordered and beautiful. An inefficient algorithm is chaotic in its relationship to the problem. It imposes unnecessary work on the system rather than discovering and reflecting the problem’s inherent structure.

1. **Waste** (Against Stewardship)

Stewardship means using resources wisely. An inefficient algorithm wastes computational resources—energy, processing power, memory—that could be used for other purposes or preserved. In a finite system, waste here means less available elsewhere.

1. **Disrespect for the Created Order** (Against Recognition of Limits)

An inefficient system treats computational resources as infinite. But they are finite. The same finiteness principle that limits extraction of natural resources also limits computational resources.

1. **Failure of Wisdom** (Against Divine Pattern)

Wisdom in creation means discovering and working *with* the inherent structure of things, not against it. An inefficient algorithm represents ignorance or indifference to the problem’s actual structure.

2.b. *Efficiency as an Expression of Stewardship*

From this perspective, we have to think about how efficiency shows a respect for Creation.

1. **Honors the Creator’s Design**

Efficient algorithms reflect the reality that creation is designed with order and beauty. By creating efficient algorithms, we align ourselves with that order rather than against it.

1. **Fulfills Our Role as Image-Bearers**

Wisdom is fundamental to who God is. When we seek efficient solutions, we participate in divine wisdom. We think carefully about the problem, observe its patterns, and design accordingly.

1. Exercises Genuine Stewardship

“Stewardship emphasizes and seeks to enhance value; proper stewardship may demand considerable sacrifice by the steward.” (Lim, 1990) Creating an efficient algorithm requires more thought, care, and sometimes sacrifice (of programmer time) to preserve resources.

1. Recognizes Our Limits

An efficient design acknowledges that resources are finite and that we are not omnipotent. It says: “This system is not infinite. We must work within its constraints wisely.” (Lim, 1990)

1. Protects the Voiceless

Wasted computational resources have real impacts—higher energy consumption, larger carbon footprints, slower systems that exclude those with limited bandwidth, devices that become obsolete faster. “People should speak and act on behalf of the voiceless.” (Lim, 1990) Efficient design is a form of justice.

3. THE ASSIGNMENT

Now that we have established the theological framework, here is the assignment that you will complete.

3.a. Part 1: Analyze an Inefficient Algorithm

You will be given (or choose) an inefficient algorithm that solves a common computational problem. Your task is to:

1. **Understand the Problem** What is the algorithm trying to solve? What are the inputs, outputs, and constraints?
2. **Analyze the Inefficiency**
 - Trace through the algorithm step by step.
 - Calculate its time and space complexity.
 - Identify where unnecessary work is being performed.
 - Describe the mismatch between the problem’s structure and the algorithm’s approach.
3. **Visualize the Impact** Create a table or graph showing:
 - How the algorithm performs as input size grows (n=10, 100, 1000, 10,000...)
 - Rough estimates of CPU cycles or memory used at each scale
 - Compare this to what a reasonable solution might require

3.b. Part 2: Propose an Optimized Solution

Design and implement a more efficient algorithm that solves the same problem. You should:

1. **Discover the Problem's Structure** What patterns or relationships does the problem have that a better solution could exploit?
 - Is there an ordering we can use?
 - Are there overlapping subproblems?
 - Can we transform the problem into a different representation?
2. **Design the Optimized Algorithm**
 - Clearly describe your approach
 - Analyze its time and space complexity
 - Explain why your solution is more efficient
3. **Implement and Test**
 - Provide working code (or pseudocode)
 - Test on the same input sizes as the original
 - Demonstrate the improvement

3.c. Part 3: Reflection on Stewardship

Write a 2-3 page reflection addressing the following questions. Ground your reflection in the theological materials provided (you should cite at least 3 specific passages or concepts):

Question 1: Discovering Creation's Order

The theological texts emphasize that God created the world with wisdom and order. As you optimized the algorithm, what did you discover about the problem's inherent structure? How is an optimized algorithm a form of "discovering" rather than "inventing"?

Question 2: Stewardship and Resources

The doctrine of creation teaches that we are called to be stewards—showing great sensitivity to resources available. How many additional CPU cycles, memory, or energy does the inefficient algorithm waste compared to your optimized version? What does this waste represent, and why should we care?

Consider: If this inefficient algorithm were used in a system deployed at scale (millions of devices, running continuously), what would be the cumulative impact? Who might be affected by that waste?

Question 3: The Ethics of Power

Human authority over creation is "circumscribed" and we are "neither omniscient nor all-powerful." An inefficient algorithm often represents a kind of laziness—taking the easy approach rather than the wise approach. How does optimization represent a form of humility and responsibility?

Question 4: Am I Designing Systems as a Steward or as an Extractor?

Reflect on this core question: “Am I designing and deploying these systems as a steward of God’s creation, or am I complicit in extraction, exploitation, or disorder?”

Consider:

- Could the inefficient algorithm be acceptable in some contexts but not others?
 - What if optimization required trade-offs (e.g., less readable code, more development time)? How would you decide?
 - How does this question apply to real systems you interact with (social media, search engines, cloud services)?
-

4. ASSIGNMENT CHECKLIST

To track your progress, use the checklist below:

Analysis (Part 1):

- Clearly articulate the problem and the inefficient algorithm
- Calculate and explain time/space complexity
- Identify sources of inefficiency with specific examples
- Provide visualization or data showing impact at scale

Optimization (Part 2):

- Propose a more efficient algorithm with clear explanation
- Analyze complexity of new algorithm
- Implement working code with reasonable test cases
- Demonstrate measurable improvement
- Explain the insight or pattern you discovered

Reflection (Part 3):

- Address all four questions thoroughly
 - Cite at least 3 specific theological concepts from provided texts
 - Connect abstract theological ideas to concrete algorithm design decisions
 - Reflect honestly on personal responsibility and choices
 - Demonstrate understanding of stewardship as a framework for technical work
-

5. GRADING RUBRIC (100 POINTS TOTAL)

5.a. Part 1: Analysis (25 points)

- **Clarity (8 pts):** Algorithm and problem are clearly explained; complexity analysis is correct and well-justified
 - **Depth (10 pts):** Thorough identification of inefficiencies; good visualization of impact; evidence of careful tracing through the code
-

- **Insight (7 pts):** Shows understanding of *why* the algorithm is inefficient, not just what it does

5.b. *Part 2: Optimization (35 points)*

- **Correctness (12 pts):** Optimized algorithm is correct and actually more efficient; code is functional
- **Insight (12 pts):** Clear explanation of the pattern or structure discovered; demonstrates genuine understanding of the problem
- **Implementation (11 pts):** Code is reasonably clean and well-commented; testing demonstrates the improvement

5.c. *Part 3: Reflection (40 points)*

- **Theological Engagement (12 pts):** Accurate and meaningful use of at least 3 theological concepts; demonstrates understanding of creation doctrine and stewardship
 - **Connection to Practice (12 pts):** Reflections are grounded in the specific algorithm and optimization choices; not generic or superficial
 - **Critical Thinking (10 pts):** Acknowledges complexity; considers trade-offs; doesn't reduce stewardship to simplistic rules
 - **Personal Responsibility (6 pts):** Shows honest reflection on choices and their implications; not defensive
-

6. SUGGESTED ALGORITHMS FOR ANALYSIS

Choose one, or propose your own:

1. **Sorting:** Bubble sort vs. Merge sort or Quick sort
 2. **Search:** Linear search vs. Binary search (on sorted data)
 3. **Data Structure Traversal:** Nested loop finding duplicates vs. Hash set approach
 4. **Fibonacci:** Naive recursive vs. Dynamic programming
 5. **String Matching:** Naive substring search vs. KMP or Boyer-Moore
 6. **Graph Traversal:** Depth-first search (DFS) with inefficient data structures vs. optimized DFS
 7. **Matrix Operations:** Naive matrix multiplication vs. optimized approaches
 8. **Database Queries:** Full table scan vs. Indexed query
-

7. DISCUSSION PROMPTS (FOR CLASS CONVERSATION)

Before or after submitting:

1. "Is optimization always ethically necessary, or are there contexts where 'good enough' is appropriate?"
 2. "Many real-world systems prioritize features and speed-to-market over efficiency. How do we balance business pressures with stewardship values?"
-

3. “Cloud computing abstracts away resource costs. Does this make it easier to be wasteful? What’s our responsibility?”
 4. “Some argue that worrying about optimization in the age of powerful hardware is ‘premature optimization.’ How would a creation theology perspective respond to that?”
 5. “AI and machine learning models are often massive and energy-intensive. Does the theological framework we’ve discussed change how you think about these systems?”
-

8. APPENDIX: PSALM 104 (SELECTED VERSES)

Praise the Lord, my soul. Lord my God, you are very great; you are clothed with splendor and majesty.

How many are your works, Lord! In wisdom you made them all; the earth is full of your creatures.

These all look to you to give them their food at the proper time. When you give it to them, they gather it up; when you open your hand, they are satisfied with good things.

May the glory of the Lord endure forever; may the Lord rejoice in his works.

The psalm celebrates God’s wisdom in design, the intricate sustainability of creation’s systems, and the responsiveness of all creatures to the Creator. When we design efficient systems, we participate in this celebration—recognizing and reflecting the wisdom already present in creation.

REFERENCES

- Gunton, C. (2001). The triune Creator: A historical and systematic study. *Pro Ecclesia*, 10(2), 237–239.
- Lim, D. S. (1990). The Doctrine of Creation and Some Implications for Modern Economics. *Transformation*, 7(2), 28–32.
- Pobee, J. S. (1985). Creation faith and responsibility for the world. *Journal of Theology for Southern Africa*, 50, 16–26.