International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

A Graph-Theoretic Proof of the Fundamental Theorem of Arithmetic with Algorithmic Construction

Hitesh Choudhury

Department of Computer Science Dimoria College, Assam, India

Abstract: ***The Fundamental Theorem of Arithmetic states that every integer greater than 1 can be factored uniquely into prime numbers, up to order. In this paper, we present a graph-theoretic framework to understand and prove this theorem. By modeling factorization as a tree structure, we define factorization trees and establish four supporting lemmas. We then develop a recursive algorithm for constructing these trees. This approach not only yields a constructive proof of the theorem but also offers visual and algorithmic insights into the uniqueness of prime factorizations.***

Keywords: Fundamental Theorem of Arithmetic, Graph Theory, Factorization Tree, Prime Factorization, Directed Acyclic Graph, Number Theory Algorithm

1. Introduction

The Fundamental Theorem of Arithmetic (FTA) is one of the foundational results in number theory:

FTA: Every integer $n > 1$ can be expressed uniquely as a product of prime numbers, up to the order of the factors.

This paper introduces a graph-theoretic perspective using *factorization trees* to both visualize and formally prove the theorem. We define these trees, establish their structural properties via four lemmas, and introduce an algorithm to construct them. Our approach offers both theoretical and pedagogical value.

2. Preliminaries

Factorization Tree
Let $n>1$ be an integer. A *factorization tree* T_n is a rooted, directed tree where:

- The root is labeled with n.
- Each internal node labeled x has children whose labels multiply to x.
- All leaf nodes are labeled with prime numbers.

Notation

- Label (v): the integer label of node v.
- Leaves (T_n): the multiset of prime labels at the leaves of T_n.

3. Lemmas

Lemma 3.1: *Every integer $n>1$ has at least one factorization tree.*
Proof. We construct T_n recursively:

- If n is prime, return a single-node tree labeled n.
- If n is composite, choose any proper factor pair (a, b) such that $n=ab$. Recursively construct

T_a and T_b, and join them as children of a root labeled n.

Since $a, b<n$, and integers are well- ordered, recursion terminates.

Lemma 3.2. *All leaves in a factorization tree are primes.*

Proof. Every non-prime node is recursively factored into smaller integers. This process halts when a node cannot be further factored, i.e., when it is prime. Hence, all leaves are primes.

Lemma 3.3. *The product of the labels of the leaves of T_n equals n.*

Proof. Each internal node's label is the product of its children. This recursive multiplicative structure ensures that the product of the leaf labels equals the root's label n.

Lemma 3.4: *All factorization trees for a given n produce the same multiset of leaf primes.*

Proof. Assume two trees for n yield different prime multisets. Then n would have two distinct prime factorizations, violating the Fundamental Theorem of Arithmetic. Hence, the multiset must be unique.

4. Constructive Algorithm

We now present a recursive algorithm to construct the factorization tree for any integer
$n>1$.

Algorithm1 Build-Factor-Tree(n)

```
if n is prime then
    return TreeNode labeled n
else
    for i ← 2 to n − 1 do
        if n mod i = 0 then
            a ← i, b ← n/i
            CreateTreeNode with label n
            Leftchild ← Build-Factor-Tree(a)
            Rightchild ← Build-Factor-Tree(b)
            return Tree rooted at n with children
    endif
```

```
    endfor
endif
```

This algorithm:
- Always terminates (due to recursion on smaller integers).
- Produces a valid factorization tree with all leaves prime.
- Guarantees that the product of leaves is *n*.

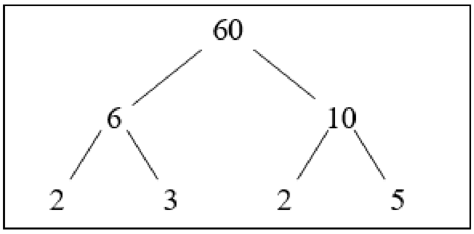
5. Main Theorem

Theorem 5.1 (Fundamental Theorem of Arithmetic)**.** *Every integer* $n>1$ *has a unique prime factorization, up to the order of the factors.*

Proof.
Existence: Follows from Lemmas 1–3 and there cursive construction algorithm.
Uniqueness: Lemma 4 ensures that all possible factorization trees yield the same multiset of prime leaves. Therefore, the prime factorization is unique up to order.

6. Example: Factorization Tree for 60

Consider60=6·10= (2·3)· (2·5). The factorization tree is:



Leaf multiset: $\{2,3,2,5\}$, corresponding to $2^2 \cdot 3 \cdot 5$.

7. Applications and Future Work

This approach provides an educational and algorithmic way to understand the uniqueness of prime factorizations. Future work may extend this model to:
- Unique factorization domains (UFDs).
- Polynomial rings and algebraic integers.
- Optimization of factor trees for minimal depth.

8. Conclusion

We presented a graph-theoretic approach to proving the Fundamental Theorem of Arithmetic by modeling integer decomposition as a factorization tree. We established foundational lemmas, implemented a recursive algorithm, and showed that this structure guarantees both the existence and uniqueness of prime factorizations. This constructive and visual approach offers deep insight into number structure and has wide-reaching implications in education, computation, and theoretical mathematics.

References

[1] Hardy, G. H., & Wright, E. M. (2008). *An Introduction to the Theory of Numbers*. Oxford University Press.
[2] West, D. B. (2001). *Introduction to Graph Theory*. Prentice Hall.
[3] Gallian, J. A. (2021). *Contemporary Abstract Algebra*. Cengage Learning.
[4] Knuth, D. E. (1997). *The Art of Computer Programming, Vol. 1: Fundamental Algorithms*. Addison-Wesley.

Paper ID: SR25608225333 DOI: https://dx.doi.org/10.21275/SR25608225333