

Demonstrație Matematică: Teorema TDI (Triunghi isoscel dreptunghic) a Divizibilității Spațiale

Autori:

Gheorghe Parascan, Maria Margoș, Ally Constantin Margoș

I. ENUNȚUL PROBLEMEI DE GEOMETRIE PURĂ

"Se dă un triunghi isoscel dreptunghic OAB cu catetele OA și AB egale cu $L = (N-1) \cdot u$. Pe cateta orizontală OA se definesc punctele discrete P_n la distanțe egale cu multiplii ai unității u . Să se determine condițiile geometrice în care proiecțiile verticale $P_n Q_n$ (cu Q_n în OB) sunt intersectate de segmentele rețelelor orizontale de pas d ."

II. IPOTEZĂ ȘI CONCLUZIE

- Ipoteză:** $OP_n = (n-1)u$ pe cateta OA . Rândurile orizontale d sunt situate la ordonatele $y_d = (d-1)u$. Pe fiecare rând d , segmentele discrete au lungimea fixă $L_{\text{seg}} = d \cdot u$.
- Concluzie:** Intersecția dintre segmentul vertical $P_n Q_n$ și segmentul orizontal $S_{\{d\}}$ este nevidă în rețeaua discretă dacă și numai dacă lungimea OP_n este divizibilă exact cu pasul rețelei d .

III. DEMONSTRAȚIA GEOMETRICĂ (STAREA FUNDAMENTALĂ DISCRETĂ)

1. Construim geometric segmentele orizontale pe fiecare rând d așezându-le cap la cap: $[0, d], [d, 2d], [2d, 3d], \dots$. Capetele acestor segmente (nodurile de rețea) se materializează exclusiv la abscisele $x = m \cdot d$ unități discrete.

2. Segmentul vertical de proiecție $P_n Q_n$ se află la abscisa fixă $x_n = n - 1$.

3. Prin suprapunere spațială, intersecția geometrică se produce doar atunci când abscisa verticalei coincide cu un nod al segmentelor orizontale: $n = m \cdot d$ (unde $m \in$

\mathbb{N}^*). **Coincidența spațială este evenimentul fizic primar.**

IV. TRADUCEREA ANALITICĂ (STAREA SECUNDARĂ, SIMBOLICĂ)

Relația clasică algebrică $n \equiv 0 \pmod{d}$ sau operația analitică de împărțire nu reprezintă cauza divizibilității, ci doar o metodă secundară, simbolică, de a nota această potrivire perfectă a segmentelor în spațiul triunghiului isoscel. Analiticul doar constată ceea ce discretul construiește deja geometric.

ARGUMENTUL PRIORITĂȚII DISCRETULUI

Dreptele Orizontale ca Generatori: În ambele module geometrice de la subsol, divizibilitatea nu mai este o proprietate abstractă, ci o consecință a **rețelelor orizontale paralele**. Fiecare rând orizontal d are o structură segmentată cu lungimea proprie d , generând puncte de contact la fiecare multiplu.

Intersecția ca Naștere a Divizorului: Divizorii unui număr n apar strict acolo unde axa sa verticală se întâlnește cu un segment al unei unde orizontale active. Algebra devine o simplă citire a acestei hărți de intersecții geometrice.

Semnătura Spatială a Numerelor Prime: Pe verticala unui număr prim p , între capătul de pe catetă (d_1) și cel de pe ipotenuză (d_p), nu se intersectează nicio altă dreaptă orizontală. Vidul geometric interior este dovada absolută a primorialității sale, fără a rula niciun algoritm de calcul analitic.

Fenomenul TDI: Explicarea Analiticului prin Discretul Fundamental

Studiu asupra modelului geometric fractal Parascan - Margoș în Triunghiul Isoscel Dreptunghic

1. Natura Fenomenului: Prioritatea Discretului

În gândirea matematică clasică (analitică), relația de divizibilitate este definită printr-o operație aritmetică abstractă de împărțire fără rest:

$$\frac{n}{d} = k, \quad \text{unde } k \in \mathbb{N}$$

Această abordare tratează numărul ca pe un punct abstract pe o axă continuă și divizibilitatea ca pe o proprietate formală algebrică. **Aceasta este însă o stare secundară a realității, o simplă metodă de notație.**

Fenomenul TDI (Triunghiul Isoscel Dreptunghic) dezvăluit de această aplicație demonstrează că **starea fundamentală a realității este pur discretă și spațială**:

1. **Axa Sursă (Cateta OA)**: Reprezintă fluxul discret al unităților. Fiecare număr natural n nu este o abstracție, ci o poziție fizică P_n determinată de acumularea a exact $n-1$ pași spațiali de lungime constantă u (unitatea elementară).
2. **Unde Orizontale Generatoare**: Spațiul interior al triunghiului este guvernat de rețele (sau unde) orizontale paralele. Fiecare rețea d este formată din segmente fizice lungi de exact d unități, așezate cap la cap. Această periodicitate spațială reprezintă starea discretă pură.
3. **Coincidența ca Naștere a Divizibilității**: Atunci când coborâm o proiecție verticală de la poziția P_n (de pe catetă) spre ipotenuza OB , aceasta intersectează capătul unui segment orizontal de pe rândul d **doar dacă** cele două structuri spațiale se potrivesc la nivel de rețea.

Divizibilitatea nu este un calcul, ci o coincidență fizică spațială. Rândurile orizontale nu „calculează” divizorii, ci se intersectează cu verticalele. Acolo unde există o intersecție fizică, se naște divizorul. Analiticul doar observă și notează simbolic această potrivire discretă primară.

4. **Vidul Primordial**: Un număr prim p se dezvăluie vizual prin absența totală a intersecțiilor intermediare. Între punctul de plecare de pe cateta unitară (d_1) și punctul de oprire de pe ipotenuza de auto-identitate (d_p), verticala numărului prim traversează un **vid geometric absolut**. Caracterul prim este o proprietate de ne-intersecție în spațiul discret.

2. Formularea Matematică a Metodei de Construcție și a Rezultatului

Pentru a formaliza geometric acest fenomen, definim triunghiul isoscel dreptunghic în planul discret $\mathbb{N}^* \times \mathbb{N}^*$ raportat la unitatea metrică elementară u .

Ipoteza de Construcție (Spațiul Discret):

Fie $O(0,0)$ originea triunghiului isoscel dreptunghic OAB , unde:

- Cateta orizontală de sus este segmentul $[OA]$, cu punctele de proiecție P_n situate la coordonatele:

$$P_n = ((n-1)u, 0), \quad \forall n \in \{1, 2, \dots, N\}$$

- Cateta verticală din dreapta este segmentul $[AB]$, de lungime $L = (N-1)u$.
- Ipotenuza $[OB]$ reprezintă limita de auto-identitate a proiecțiilor, unde punctele de graniță Q_n au coordonatele:

$$Q_n = ((n-1)u, (n-1)u)$$

- Verticala de proiecție a fiecărui număr n este segmentul închis:

$$V_n = [P_n Q_n] = \{(x, y) \mid (n-1)u \leq x \leq (n-1)u, 0 \leq y \leq (n-1)u\}$$

Pe fiecare rând discret $d \in \{1, 2, \dots, N\}$, situat la adâncimea ordonatei $y_d = (d-1)u$, se construiește o rețea de segmente orizontale congruente, cu nodurile de legătură $M_{\{d,k\}}$ plasate la abscisele:

$$M_{\{d,k\}} = ((k-1)u, (d-1)u), \quad \forall k \in \mathbb{N}^*$$

Teorema TDI a Coincidenței (Rezultatul):

Intersecția geometrică dintre verticala de proiecție V_n și mulțimea nodurilor de rețea orizontală $M_{\{d,k\}}$ este nevidă:

$\mathcal{I} = V_n \cap \{M_{\{d,k\}}\}_{k \geq 1} \neq \emptyset$ dacă și numai dacă există un nod $M_{\{d,k\}}$ a cărui abscisă coincide perfect cu abscisa verticalei V_n în limitele triunghiului. Această coincidență spațială discretă determină rezultatul:

$$\boxed{(n-1) \cdot u = (k-1) \cdot d \cdot u}$$

Concluzia și Traducerea Analitică (Starea Secundară):

Prin simplificarea unității spațiale fundamentale u , obținem relația numerică brută:

$$n - 1 = (k - 1) \cdot d$$

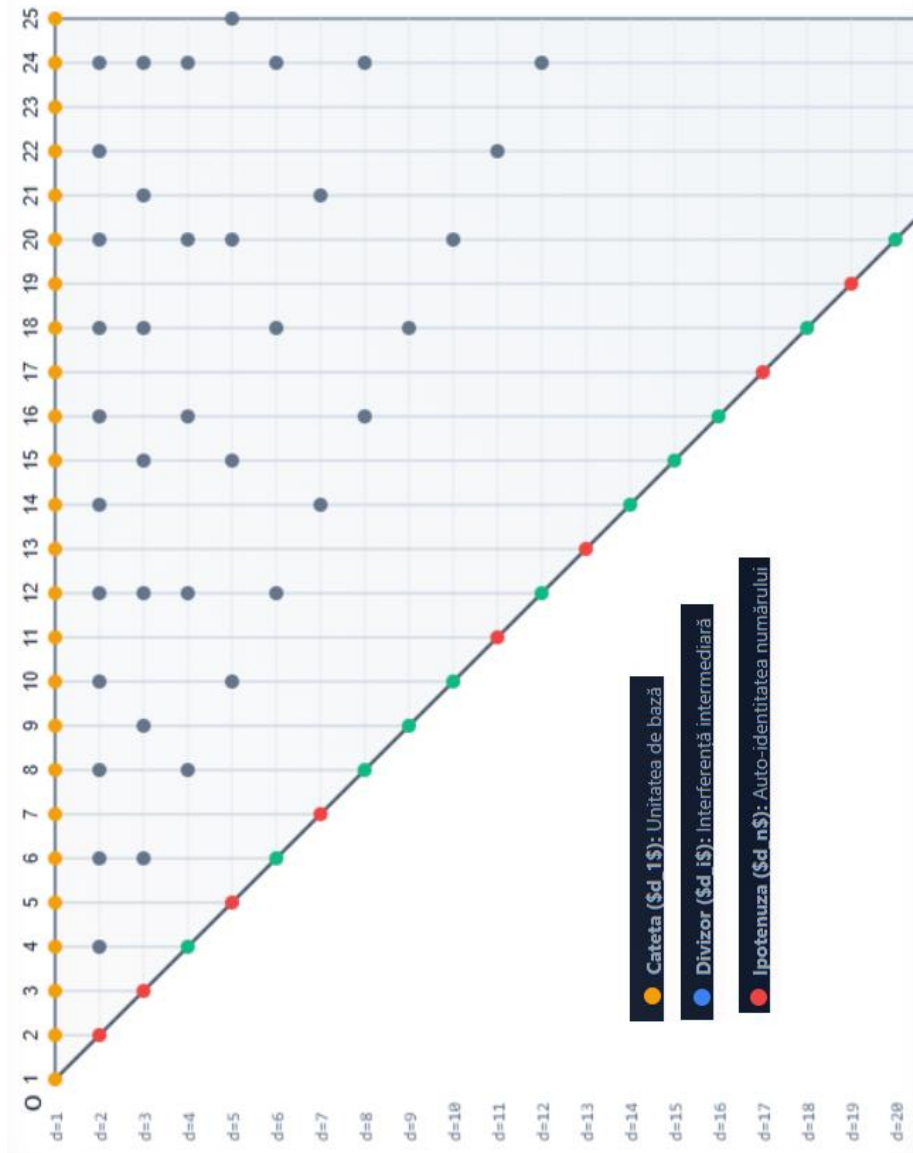
Făcând translația față de originea fizică O (care corespunde indicelui de start 1 , unde $n' = n$ și $d' = d$), obținem ecuația simbolică analitică:

$$n = k \cdot d \implies d \in \mathcal{D}(n)$$

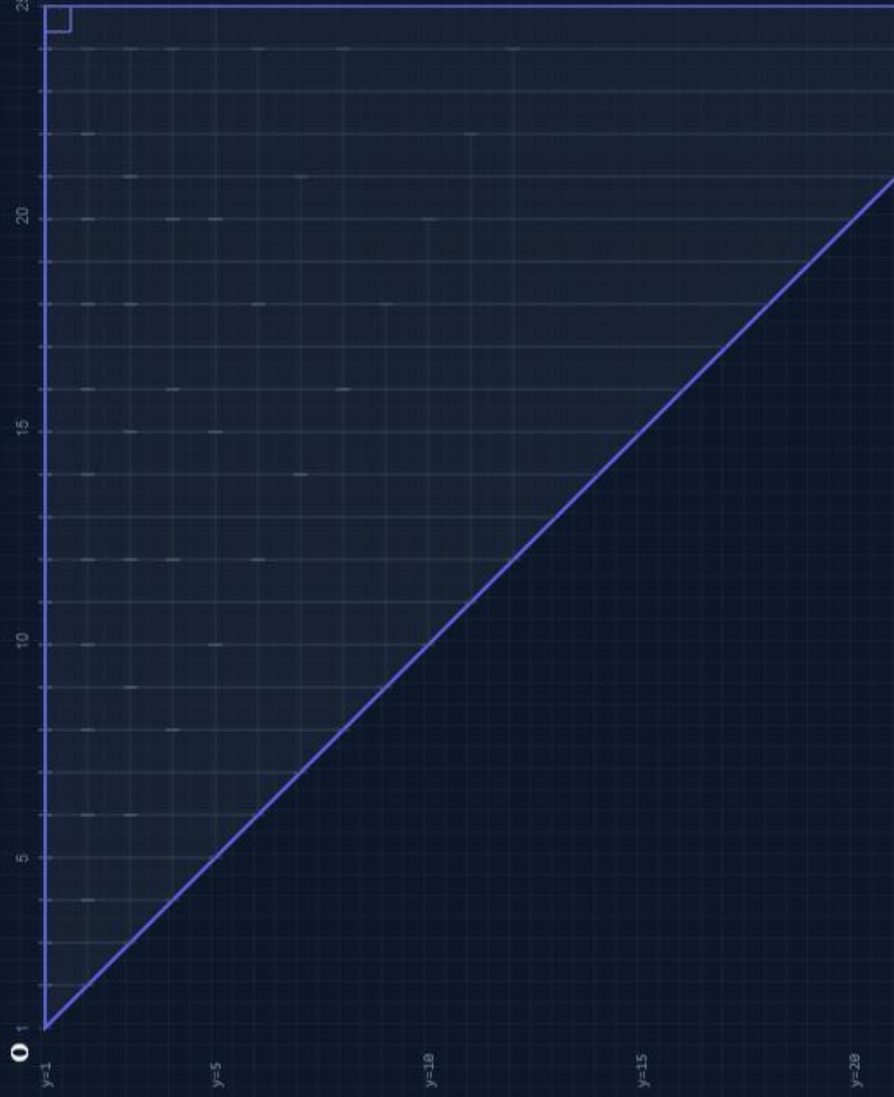
Rezultat: Relația analitică de divizibilitate $d|n$ este proiecția secundară (umbrele algebrice ale realității) rezultată din starea discretă primară, definită exclusiv ca o **coincidență geometrică a nodurilor de rețea în Triunghiul Isoscel Dreptunghic**.

Autori:

Gheorghe Parascan, Maria Margoș, Ally Constantin Margoș



25 A



O

$y=1$

$y=5$

$y=10$

$y=15$

$y=20$

Triunghiul dreptunghic isoscel care stă la baza întregii structuri primare a matematicii

```
<!DOCTYPE html>
<html lang="ro">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Tabelul Fractal Parascan-Margoș: Triunghi Isoscel
Dreptunghic</title>
  <!-- Tailwind CSS pentru o interfață modernă și elegantă -->
  <script src="https://cdn.tailwindcss.com"></script>
  <style>
    .wood-card {
      background: linear-gradient(135deg, #1e293b 0%, #0f172a
100%);
      border: 1px solid rgba(251, 191, 36, 0.2);
      box-shadow: 0 10px 30px rgba(0,0,0,0.5);
    }
    canvas {
      border-radius: 8px;
      background: #ffffff;
      box-shadow: inset 0 0 20px rgba(0,0,0,0.15), 0 4px 10px
rgba(0,0,0,0.3);
    }
  </style>
</head>
<body class="bg-slate-950 text-slate-100 min-h-screen flex flex-col
font-sans">

  <!-- Header -->
```

```
<header class="bg-slate-900/85 border-b border-amber-500/20 p-4 shadow-lg">
```

```
<div class="max-w-6xl mx-auto flex flex-col md:flex-row justify-between items-center gap-4">
```

```
<div>
```

```
<h1 class="text-xl md:text-2xl font-extrabold tracking-tight text-amber-400">
```

```
Numărătoarea Geometrică • Triunghiul Isoscel Dreptunghic (TDI)
```

```
</h1>
```

```
<p class="text-xs text-slate-400 mt-1">
```

```
Modelul Parascan - Margoș:  $d_1$  este situat direct pe catetă, iar divizorii sub-adiacenți se succed în jos
```

```
</p>
```

```
</div>
```

```
<div class="bg-amber-500/10 border border-amber-500/30 rounded-lg py-1 px-3 text-right">
```

```
<span class="text-[10px] text-amber-400 block font-bold tracking-wider">PRINCIPIU METODOLOGIC</span>
```

```
<span class="text-xs text-slate-200 font-semibold">Explicarea analiticului prin starea discretă</span>
```

```
</div>
```

```
</div>
```

```
</header>
```

```
<!-- Main Content -->
```

```
<main class="flex-1 max-w-6xl w-full mx-auto p-4 flex flex-col lg:flex-row gap-6 items-start justify-center">
```

```
<!-- Left Column: Interactive Canvases -->
```

```
<div class="flex-1 flex flex-col gap-6 w-full">
```

```
<!-- Primul Modul: Cu Mărgele (Reprezentarea Discretă) -->
```

```
<div class="wood-card p-5 rounded-2xl w-full flex flex-col items-center">
```

```
<div class="w-full flex justify-between items-center mb-2 px-2 text-xs text-slate-400">
```

```
<span class="text-amber-400 font-bold uppercase tracking-wider">Modulul 1: Reprezentarea Discretă (Mărele fizice)</span>
```

```
<div class="active-hud font-mono text-amber-400 bg-slate-950 px-2 py-0.5 rounded border border-slate-800 text-[11px]">
```

```
  Niciun număr selectat
```

```
</div>
```

```
</div>
```

```
<!-- Canvas Wrapper to center and scale -->
```

```
<div class="w-full overflow-x-auto flex justify-center py-1">
```

```
  <canvas id="tdiCanvas" width="760" height="540" class="cursor-crosshair"></canvas>
```

```
</div>
```

```
<!-- Canvas Legend -->
```

```
<div class="w-full grid grid-cols-3 gap-2 text-center mt-3 text-[11px] text-slate-400 border-t border-slate-800 pt-3">
```

```
<div>
```

```
  <span class="inline-block w-2.5 h-2.5 bg-amber-500 rounded-full mr-1"></span>
```

```
    <strong>Cateta ( $d_1$ ):</strong> Unitatea de bază
```

```
</div>
```

```
<div>
```

```
  <span class="inline-block w-2.5 h-2.5 bg-blue-500 rounded-full mr-1"></span>
```

```
  <strong>Divizor ( $d_i$ ):</strong> Interferență intermediară
```

```
</div>
```

```
<div>
```

```
  <span class="inline-block w-2.5 h-2.5 bg-red-500 rounded-full mr-1"></span>
```

```
  <strong>Ipotenuza ( $d_n$ ):</strong> Auto-identitatea numărului
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- Al Doilea Modul: Cu Segmente Orizontale  
(Reprezentarea Geometrică Pură) -->
```

```
<div class="wood-card p-5 rounded-2xl w-full flex flex-col  
items-center">
```

```
<div class="w-full flex justify-between items-center mb-2  
px-2 text-xs text-slate-400">
```

```
<span class="text-emerald-400 font-bold uppercase  
tracking-wider">Modulul 2: Reprezentarea Geometrică Pură (Segmente  
Orizontale de Pas  $d$ )</span>
```

```
<div class="active-hud font-mono text-emerald-400 bg-  
slate-950 px-2 py-0.5 rounded border border-slate-800 text-[11px]">
```

```
Niciun număr selectat
```

```
</div>
```

```
</div>
```

```
<!-- Canvas Wrapper for Segments -->
```

```
<div class="w-full overflow-x-auto flex justify-center py-  
1">
```

```
<canvas id="segmentCanvas" width="760"  
height="540" class="cursor-crosshair"></canvas>
```

```
</div>
```

```
<!-- Canvas Legend for Segments -->
```

```
<div class="w-full grid grid-cols-3 gap-2 text-center mt-3  
text-[11px] text-slate-400 border-t border-slate-800 pt-3">
```

```
<div>
```

```
<span class="inline-block w-5 h-0.5 bg-amber-500  
mr-1 align-middle"></span>
```

```
<strong>Axa de Start ( $d_1$ ):</strong> Cateta de sus
```

```
</div>
```

```
<div>
```

```
<span class="inline-block w-5 h-0.5 bg-blue-500 mr-1  
align-middle"></span>
```

```
<strong>Unde Orizontale ( $d_i$ ):</strong> Pasul  
discret care generează divizorii
```

```

    </div>
    <div>
      <span class="inline-block w-5 h-0.5 bg-red-500 mr-1
align-middle"></span>
      <strong>Ipoteza Primordială ( $d_n$ ):</strong>
Limita de auto-identitate
    </div>
  </div>
</div>


```

<!-- Al Treilea Modul: Rețeaua Sincronă de Segmente (Rezonanța Undelor Orizontale) -->

```

<div class="wood-card p-5 rounded-2xl w-full flex flex-col
items-center">
  <div class="wood-card p-5 rounded-2xl w-full flex flex-col
items-center border-indigo-500/40">
    <div class="w-full flex justify-between items-center mb-
2 px-2 text-xs text-slate-400">
      <span class="text-indigo-400 font-extrabold
uppercase tracking-wider flex items-center gap-1.5">

```

 Modulul 3: Teorema Coincidenței Segmentelor în Triunghiul Isoscel Dreptunghic (TDI)

```

    </span>
    <div class="active-hud font-mono text-indigo-400 bg-
slate-950 px-2 py-0.5 rounded border border-slate-800 text-[11px]">
      Niciun număr selectat
    </div>
  </div>

```

```

<!-- Canvas Wrapper for Segments 2 -->
<div class="w-full overflow-x-auto flex justify-center
py-1">

```

```

  <canvas id="segmentCanvas2" width="760"
height="540" class="cursor-crosshair"></canvas>
</div>

```

```

<!-- Canvas Legend for Segments 2 -->

```

```
<div class="w-full grid grid-cols-3 gap-2 text-center mt-3 text-[11px] text-slate-400 border-t border-slate-800 pt-3">
```

```
<div>
```

```
<span class="inline-block font-mono text-amber-500 mr-1 font-bold">O — A</span>
```

```
<strong>Cateta Orizontală:</strong> Segmentul unitar de bază
```

```
</div>
```

```
<div>
```

```
<span class="inline-block font-mono text-indigo-500 mr-1 font-bold">Pn — Qn</span>
```

```
<strong>Axa de Proiecție:</strong> Segmentele verticale congruente
```

```
</div>
```

```
<div>
```

```
<span class="inline-block font-mono text-red-500 mr-1 font-bold">O — B</span>
```

```
<strong>Ipotenuza:</strong> Limita superioară discretă
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- Panou de Demonstrație Formală (Geometrie Pură vs Algebră) -->
```

```
<div class="bg-slate-900/90 border border-indigo-500/30 rounded-2xl p-6 mt-4 space-y-4 shadow-xl">
```

```
<h3 class="text-base font-extrabold text-indigo-400 border-b border-indigo-950 pb-2 flex items-center gap-2">
```

```
<span>📄</span> Demonstrație Matematică:  
Teorema TDI a Divizibilității Spațiale
```

```
</h3>
```

```
<div class="grid grid-cols-1 md:grid-cols-2 gap-6 text-xs leading-relaxed text-slate-300">
```

```
<!-- Caseta 1: Enunț și Ipoteză -->
```

<div class="space-y-2 bg-slate-950/40 p-4 rounded-xl border border-slate-800">

I. Enunțul Problemei de Geometrie Pură

<p class="italic text-slate-200">

"Se dă un triunghi isoscel dreptunghic OAB cu catetele OA și AB egale cu $L = (N-1) \cdot u$. Pe cateta orizontală OA se definesc punctele discrete P_n la distanțe egale cu multiplii ai unității u . Să se determine condițiile geometrice în care proiecțiile verticale $P_n Q_n$ (cu Q_n în OB) sunt intersectate de segmentele rețelelor orizontale de pas d ."

</p>

II. Ipoteză și Concluzie

<ul class="list-disc pl-4 space-y-1 text-slate-300">

Ipoteză: $OP_n = (n-1)u$ pe cateta OA. Rândurile orizontale d sunt situate la ordonatele $y_d = (d-1)u$. Pe fiecare rând d , segmentele discrete au lungimea fixă $L_{\text{seg}} = d \cdot u$.

Concluzie: Intersecția dintre segmentul vertical $P_n Q_n$ și segmentul orizontal $S_{\{d\}}$ este nevidă în rețeaua discretă dacă și numai dacă lungimea OP_n este divizibilă exact cu pasul rețelei d .

</div>

<!-- Caseta 2: Demonstrația Priorității Discretului -->

<div class="space-y-2 bg-slate-950/40 p-4 rounded-xl border border-slate-800">

III. Demonstrația Geometrică (Starea Fundamentală Discretă)

<p>

1. Construim geometric segmentele orizontale pe fiecare rând d așezându-le cap la cap: $\{0, d\}$, $[d, 2d]$, $[2d, 3d]$, \dots.

Capetele acestor segmente (nodurile de rețea) se materializează exclusiv la abscisele $x = m \cdot d$ unități discrete.

</p>

<p>

2. Segmentul vertical de proiecție $P_n Q_n$ se află la abscisa fixă $x_n = n - 1$.

</p>

<p>

3. Prin suprapunere spațială, intersecția geometrică se produce doar atunci când abscisa verticalei coincide cu un nod al segmentelor orizontale: $n = m \cdot d$ (unde $m \in \mathbb{N}^*$). Coincidența spațială este evenimentul fizic primar.

</p>

IV. Traducerea Analitică (Starea Secundară, Simbolică)

<p class="text-slate-400 italic">

Relația clasică algebrică $n \equiv 0 \pmod{d}$ sau operația analitică de împărțire nu reprezintă cauza divizibilității, ci doar o metodă secundară, simbolică, de a nota această potrivire perfectă a segmentelor în spațiul triunghiului isoscel. Analiticul doar constată ceea ce discretul construiește deja geometric.

</p>

</div>

</div>

</div>

</div>

</div>


<!-- Right Column: Philosophy & Real-time Math Inspector -->

<div class="w-full lg:w-96 flex flex-col gap-5 shrink-0 lg:sticky lg:top-4">

<!-- Real-time Inspector Panel -->

```
<div class="bg-slate-900 border border-amber-500/20 rounded-xl p-5 shadow-lg">
```

```
<h3 class="text-sm font-bold text-amber-400 uppercase tracking-wider mb-3 flex items-center gap-2">
```

```
<span></span> Inspectorul Discret</h3>
```

```
<div id="inspector-default" class="text-xs text-slate-500 italic py-6 text-center">
```

Treceți cursorul peste axele numerelor din triunghiuri pentru a declanșa observarea geometrică sincronizată.

```
</div>
```

```
<div id="inspector-display" class="hidden space-y-4">
```

```
<div class="flex justify-between items-baseline">
```

```
<span class="text-xs text-slate-400">Numărul analizat:</span>
```

```
<span id="inspect-val" class="text-4xl font-black text-white">12</span>
```

```
</div>
```

```
<div class="flex justify-between items-center text-xs">
```

```
<span>Caracteristică:</span>
```

```
<span id="inspect-tag" class="px-2 py-0.5 rounded font-bold uppercase">Compus</span>
```

```
</div>
```

```
<hr class="border-slate-800">
```

```
<div>
```

```
<span class="text-xs text-slate-400 block mb-1.5 font-bold uppercase">Alinieri pe verticală (Divizori):</span>
```

```
<div id="inspect-divs" class="flex flex-wrap gap-1.5"></div>
```

```
</div>
```

```
<div class="bg-slate-950 p-3 rounded-lg border border-slate-800">
```

<div id="inspect-equations" class="text-xs font-mono space-y-1 text-slate-300"></div>

</div>

<div class="text-xs text-slate-400 pt-2 border-t border-slate-800">

Factorizare primă:

</div>

</div>

</div>

<!-- Deep Philosophy Column -->

<div class="bg-slate-900/60 border border-slate-800 rounded-xl p-5 text-xs text-slate-300 space-y-3 leading-relaxed">

<h3 class="text-sm font-bold text-amber-400 uppercase tracking-wider flex items-center gap-1.5">

🔺 Argumentul Priorității Discretului

</h3>

<p>

Dreptele Orizontale ca Generatori: În ambele module geometrice de la subsol, divizibilitatea nu mai este o proprietate abstractă, ci o consecință a ****rețelelor orizontale paralele****. Fiecare rând orizontal d are o structură segmentată cu lungimea proprie d , generând puncte de contact la fiecare multiplu.

</p>

<p>

Intersecția ca Naștere a Divizorului: Divizorii unui număr n apar strict acolo unde axa sa verticală se întâlnește cu un segment al unei unde orizontale active. Algebra devine o simplă citire a acestei hărți de intersecții geometrice.

</p>

<p>

**Semnătura Spțială a Numerelor
Prime:**

Pe verticala unui număr prim p , între capătul de pe catetă (d_1) și cel de pe ipotenuză (p), nu se intersectează nicio altă dreaptă orizontală. Vidul geometric interior este dovada absolută a primorialității sale, fără a rula niciun algoritm de calcul analitic.

```
</p>  
</div>  
</div>  
</main>
```

```
<!-- Footer -->
```

```
<footer class="bg-slate-950 border-t border-slate-900 p-4 mt-8  
text-center text-xs text-slate-500">
```

Triunghiul Isoscel Dreptunghic ca reprezentare materială a mulțimii \mathbb{N} . Concepția didactică Parascan - Margoș digitalizată.

```
</footer>
```

```
<!-- Interactive Canvases & Mathematics Logic -->
```

```
<script>
```

```
const canvas1 = document.getElementById('tdiCanvas');  
const ctx1 = canvas1.getContext('2d');
```

```
const canvas2 = document.getElementById('segmentCanvas');  
const ctx2 = canvas2.getContext('2d');
```

```
const canvas3 = document.getElementById('segmentCanvas2');  
const ctx3 = canvas3.getContext('2d');
```

```
// Configurație Triunghi Isoscel Dreptunghic adaptat perfect
```

```
const N = 25; // Numărul maxim de elemente
```

```
const startX = 60; // Margine stânga pentru etichetele y
```

```
const startY = 40; // Margine sus pentru etichetele x (Cateta de  
sus, d1)
```

```
const unit = 25; // Dimensiunea discretă a fiecărui pas (în pixeli)
```

```
const L = (N - 1) * unit; // Lungimea ambelor catete (Isoscel perfect!)
```

```
let hoveredNum = null;
```

```
// Verifică dacă un număr este prim
```

```
function isPrime(num) {  
  if (num <= 1) return false;  
  for (let i = 2; i <= Math.sqrt(num); i++) {  
    if (num % i === 0) return false;  
  }  
  return true;  
}
```

```
// Returnează descompunerea în factori primi
```

```
function getFactorsText(num) {  
  if (num <= 1) return "fără";  
  let temp = num;  
  let factors = [];  
  let d = 2;  
  while (temp > 1) {  
    let count = 0;  
    while (temp % d === 0) {  
      count++;  
      temp /= d;  
    }  
    if (count > 0) {  
      factors.push(count > 1 ? `${d}^${count}` : `${d}`);  
    }  
    d++;  
    if (d * d > temp && temp > 1) {  
      factors.push(`${temp}`);  
      break;  
    }  
  }  
  return factors.join(" × ");  
}
```

```

// Desenarea geometrică completă pentru Modulul 1 (Mărgele)
function drawBeads(highlightedCol = null) {
    ctx1.clearRect(0, 0, canvas1.width, canvas1.height);

    // Fundal Triunghi Isoscel
    ctx1.beginPath();
    ctx1.moveTo(startX, startY);
    ctx1.lineTo(startX + L, startY);
    ctx1.lineTo(startX + L, startY + L);
    ctx1.closePath();

    const grad = ctx1.createLinearGradient(startX, startY, startX +
L, startY + L);
    grad.addColorStop(0, '#fafafa');
    grad.addColorStop(1, '#f1f5f9');
    ctx1.fillStyle = grad;
    ctx1.fill();

    ctx1.strokeStyle = '#475569';
    ctx1.lineWidth = 2;
    ctx1.stroke();

    // Liniile orizontale d
    for (let d = 1; d <= N; d++) {
        let y = startY + (d - 1) * unit;
        ctx1.beginPath();
        ctx1.strokeStyle = 'rgba(226, 232, 240, 0.8)';
        ctx1.lineWidth = 1;
        ctx1.moveTo(startX + (d - 1) * unit, y);
        ctx1.lineTo(startX + L, y);
        ctx1.stroke();

        ctx1.fillStyle = '#64748b';
        ctx1.font = '10px monospace';
        ctx1.fillText(`d=${d}`, startX - 35, y + 4);
    }
}

```

```

// Liniile verticale și mărgelile discrete
for (let n = 1; n <= N; n++) {
  let x = startX + (n - 1) * unit;
  let yTop = startY;
  let yBottom = startY + (n - 1) * unit;

  const isActive = (highlightedCol === n);
  ctx1.beginPath();
  ctx1.strokeStyle = isActive ? '#f59e0b' : '#cbd5e1';
  ctx1.lineWidth = isActive ? 3 : 1.2;
  ctx1.moveTo(x, yTop);
  ctx1.lineTo(x, yBottom);
  ctx1.stroke();

  // Numerele naturale
  ctx1.fillStyle = isActive ? '#b45309' : '#1e293b';
  ctx1.font = isActive ? 'bold 12px sans-serif' : '11px sans-
serif';
  ctx1.fillText(n, x - (n >= 10 ? 6 : 3), startY - 10);

  let divizori = [];
  for (let d = 1; d <= n; d++) {
    if (n % d === 0) divizori.push(d);
  }

  divizori.forEach(d => {
    let yInterferenta = startY + (d - 1) * unit;

    if (d === 1) {
      ctx1.fillStyle = isActive ? '#d97706' : '#f59e0b';
    } else if (d === n) {
      if (n > 1 && isPrime(n)) {
        ctx1.fillStyle = '#ef4444';
      } else {
        ctx1.fillStyle = '#10b981';
      }
    }
  });
}

```

```

    } else {
        ctx1.fillStyle = isActive ? '#3b82f6' : '#64748b';
    }

    ctx1.beginPath();
    ctx1.arc(x, yInterferenta, isActive ? 5.5 : 4, 0, Math.PI *
2);

    ctx1.fill();

    if (isActive) {
        ctx1.strokeStyle = '#0f172a';
        ctx1.lineWidth = 1;
        ctx1.stroke();
    }
});
}

ctx1.fillStyle = '#1e293b';
ctx1.font = 'bold 12px sans-serif';
ctx1.fillText("O", startX - 18, startY - 8);
}

// Desenarea geometrică completă pentru Modulul 2 (Segmente
Orizontale)
function drawSegments(highlightedCol = null) {
    ctx2.clearRect(0, 0, canvas2.width, canvas2.height);

    // Fundal Triunghi Isoscel
    ctx2.beginPath();
    ctx2.moveTo(startX, startY);
    ctx2.lineTo(startX + L, startY);
    ctx2.lineTo(startX + L, startY + L);
    ctx2.closePath();

    const grad = ctx2.createLinearGradient(startX, startY, startX +
L, startY + L);
    grad.addColorStop(0, '#f8fafc');

```

```
grad.addColorStop(1, '#cbd5e1');
ctx2.fillStyle = grad;
ctx2.fill();
```

```
ctx2.strokeStyle = '#334155';
ctx2.lineWidth = 2.5;
ctx2.stroke();
```

```
// 1. Desenăm axele verticale principale (numerele naturale n)
```

```
for (let n = 1; n <= N; n++) {
  let x = startX + (n - 1) * unit;
  let yTop = startY;
  let yBottom = startY + (n - 1) * unit;
```

```
  const isActive = (highlightedCol === n);
```

```
  ctx2.beginPath();
  ctx2.strokeStyle = isActive ? '#10b981' : '#e2e8f0';
  ctx2.lineWidth = isActive ? 3 : 1.2;
  ctx2.moveTo(x, yTop);
  ctx2.lineTo(x, yBottom);
  ctx2.stroke();
```

```
  // Numerele de pe cateta de sus
```

```
  ctx2.fillStyle = isActive ? '#047857' : '#475569';
  ctx2.font = isActive ? 'bold 12px sans-serif' : '11px sans-
```

```
serif';
```

```
  ctx2.fillText(n, x - (n >= 10 ? 6 : 3), startY - 10);
```

```
}
```

```
// 2. Desenăm dreptele interne orizontale segmentate (Pasul discret al fiecărui divizor d)
```

```
for (let d = 1; d <= N; d++) {
  let y = startY + (d - 1) * unit;
```

```
  // Verificăm dacă d este divizor al numărului selectat activ
```

```
const isDivisorActive = (highlightedCol &&
(highlightedCol % d === 0));
```

```
// Linie discretă de ghidaj pe fundalul rândului d
ctx2.beginPath();
ctx2.strokeStyle = isDivisorActive ? 'rgba(37, 99, 235,
0.15)' : 'rgba(203, 213, 225, 0.4)';
ctx2.lineWidth = 0.8;
ctx2.moveTo(startX + (d - 1) * unit, y);
ctx2.lineTo(startX + L, y);
ctx2.stroke();
```

```
// Construim segmentele horizontale cu lungimea/pasul exact
d
```

```
for (let m = d; m + d <= N; m += d) {
  let x1 = startX + (m - 1) * unit;
  let x2 = startX + (m + d - 1) * unit;
```

```
// Determinăm dacă acest segment specific aparține undei
divizorului activ selectat
```

```
const isSegmentHighlighted = isDivisorActive &&
(highlightedCol >= m && highlightedCol <= m + d);
```

```
ctx2.beginPath();
if (isSegmentHighlighted) {
  ctx2.strokeStyle = '#2563eb'; // Albastru intens pentru
generatorul activ
```

```
  ctx2.lineWidth = 3.5;
} else if (isDivisorActive) {
  ctx2.strokeStyle = 'rgba(37, 99, 235, 0.45)'; // Albastru
fin pentru restul undei active
```

```
  ctx2.lineWidth = 2;
} else {
  ctx2.strokeStyle = 'rgba(148, 163, 184, 0.4)'; // Linie
normală inactivă
```

```
  ctx2.lineWidth = 1.2;
}
```

```
ctx2.moveTo(x1, y);
ctx2.lineTo(x2, y);
ctx2.stroke();
```

// Tacheți perpendiculari la limitele discrete ale
segmentului d

```
ctx2.beginPath();
ctx2.strokeStyle = isDivisorActive ? '#1e3a8a' :
'#94a3b8';
ctx2.lineWidth = isSegmentHighlighted ? 2 : 1;
ctx2.moveTo(x1, y - 3); ctx2.lineTo(x1, y + 3);
ctx2.moveTo(x2, y - 3); ctx2.lineTo(x2, y + 3);
ctx2.stroke();
}
```

// 3. Materializăm punctele de intersecție (unde se nasc
divizorii din orizontale)

```
for (let n = d; n <= N; n += d) {
  let x = startX + (n - 1) * unit;
  const isActiveIntersection = (highlightedCol === n);

  ctx2.beginPath();
  if (isActiveIntersection) {
    // Evidențiere nod activ generat de intersecție
    if (d === 1) {
      ctx2.fillStyle = '#f59e0b'; // Unitatea primordială pe
      catetă
    } else if (d === n) {
      ctx2.fillStyle = isPrime(n) ? '#ef4444' : '#10b981'; //
      Capătul de pe ipotenuză
    } else {
      ctx2.fillStyle = '#3b82f6'; // Divizor intermediar
      intersectat
    }
    ctx2.arc(x, y, 6, 0, Math.PI * 2);
    ctx2.fill();
  }
}
```

```

        ctx2.strokeStyle = '#ffffff';
        ctx2.lineWidth = 1;
        ctx2.stroke();
    } else {
        // Punct discret inert de rețea
        ctx2.fillStyle = 'rgba(71, 85, 105, 0.45)';
        ctx2.arc(x, y, 2.5, 0, Math.PI * 2);
        ctx2.fill();
    }
}
}
}

```

```

// Etichete rânduri pe axa din stânga
for (let d = 1; d <= N; d++) {
    let y = startY + (d - 1) * unit;
    ctx2.fillStyle = '#64748b';
    ctx2.font = '10px monospace';
    ctx2.fillText(`d=${d}`, startX - 35, y + 3);
}

```

```

ctx2.fillStyle = '#1e293b';
ctx2.font = 'bold 12px sans-serif';
ctx2.fillText("O", startX - 18, startY - 8);
}

```

// Desenarea geometrică completă pentru Modulul 3 (Schiță de Geometrie Pură / Blueprint)

```

function drawSegments2(highlightedCol = null) {
    ctx3.clearRect(0, 0, canvas3.width, canvas3.height);

```

// 1. Grid Fundal Blueprint (Hârtie milimetrică tehnică)

```

ctx3.fillStyle = '#0f172a'; // Albastru închis pentru fundal
blueprint

```

```

ctx3.fillRect(0, 0, canvas3.width, canvas3.height);

```

```

ctx3.strokeStyle = '#1e293b';
ctx3.lineWidth = 0.5;

```

```

for (let x = 0; x < canvas3.width; x += 12.5) {
  ctx3.beginPath();
  ctx3.moveTo(x, 0); ctx3.lineTo(x, canvas3.height);
  ctx3.stroke();
}
for (let y = 0; y < canvas3.height; y += 12.5) {
  ctx3.beginPath();
  ctx3.moveTo(0, y); ctx3.lineTo(canvas3.width, y);
  ctx3.stroke();
}

```

// 2. Desenăm triunghiul geometric formal OAB

```

ctx3.beginPath();
ctx3.moveTo(startX, startY); // O(60, 40)
ctx3.lineTo(startX + L, startY); // A(60 + L, 40)
ctx3.lineTo(startX + L, startY + L); // B(60 + L, 40 + L)
ctx3.closePath();

```

```

ctx3.fillStyle = 'rgba(30, 41, 59, 0.6)';
ctx3.fill();

```

geometric

```

ctx3.strokeStyle = '#6366f1'; // Indigo aprins pentru cadrul

```

```

ctx3.lineWidth = 2;
ctx3.stroke();

```

// Marcaje de Vârfuri: O, A, B

```

ctx3.fillStyle = '#f8f8f8';
ctx3.font = 'bold 14px serif';
ctx3.fillText("O", startX - 20, startY - 10);
ctx3.fillText("A", startX + L + 10, startY - 10);
ctx3.fillText("B", startX + L + 10, startY + L + 15);

```

// Marcaj unghi drept în A

```

ctx3.beginPath();
ctx3.strokeStyle = '#818cf8';
ctx3.lineWidth = 1;

```

```
ctx3.moveTo(startX + L - 15, startY);
ctx3.lineTo(startX + L - 15, startY + 15);
ctx3.lineTo(startX + L, startY + 15);
ctx3.stroke();
```

```
// 3. Desenarea axelor verticale discrete  $P_n - Q_n$ 
```

```
for (let n = 1; n <= N; n++) {
  let x = startX + (n - 1) * unit;
  let yTop = startY;
  let yBottom = startY + (n - 1) * unit;
```

```
  const isActive = (highlightedCol === n);
```

```
  ctx3.beginPath();
  ctx3.strokeStyle = isActive ? '#fbbf24' : '#334155'; // Galben
```

de evidențiere geometrică

```
  ctx3.lineWidth = isActive ? 2.5 : 1;
  ctx3.moveTo(x, yTop);
  ctx3.lineTo(x, yBottom);
  ctx3.stroke();
```

```
// Etichetele matematice ale punctelor de pe cateta OA ( $P_n$ )
```

```
if (isActive) {
  ctx3.fillStyle = '#fbbf24';
  ctx3.font = 'bold 11px serif';
  ctx3.fillText(`P${n}`, x - 8, startY - 22);
```

```
// Punctul de intersecție pe ipotenuză ( $Q_n$ )
```

```
  ctx3.fillStyle = '#ef4444';
  ctx3.beginPath();
  ctx3.arc(x, yBottom, 5, 0, Math.PI * 2);
  ctx3.fill();
  ctx3.fillText(`Q${n}`, x - 8, yBottom + 18);
} else if (n === 1 || n === N || n % 5 === 0) {
  ctx3.fillStyle = '#64748b';
  ctx3.font = '9px sans-serif';
  ctx3.fillText(n, x - 3, startY - 10);
```

```
}  
}
```

// 4. Desenarea segmentelor orizontale pure (interferențele rețelei)

```
for (let d = 1; d <= N; d++) {  
  let y = startY + (d - 1) * unit;  
  const isDivisorActive = (highlightedCol &&  
(highlightedCol % d === 0));
```

```
  // Schițăm razele orizontale discrete  
  for (let m = d; m + d <= N; m += d) {  
    let x1 = startX + (m - 1) * unit;  
    let x2 = startX + (m + d - 1) * unit;
```

```
    const isSegmentHighlighted = isDivisorActive &&  
      (highlightedCol >= m && highlightedCol <= m + d);
```

```
    ctx3.beginPath();  
    if (isSegmentHighlighted) {  
      ctx3.strokeStyle = '#f59e0b'; // Portocaliu la  
      coincidența geometrică activă  
      ctx3.lineWidth = 3;  
    } else if (isDivisorActive) {  
      ctx3.strokeStyle = 'rgba(129, 140, 248, 0.8)'; // Indigo  
      clar la unda activă  
      ctx3.lineWidth = 1.5;  
    } else {  
      ctx3.strokeStyle = 'rgba(71, 85, 105, 0.4)'; // Culoare  
      neutră de construcție  
      ctx3.lineWidth = 0.8;  
    }  
    ctx3.moveTo(x1, y);  
    ctx3.lineTo(x2, y);  
    ctx3.stroke();
```

```
// Marcaje de congruență geometrică (bare tehnice  
perpendiculare)
```

```
    ctx3.beginPath();  
    ctx3.strokeStyle = isSegmentHighlighted ? '#fbbf24' :  
'rgba(100, 116, 139, 0.6)';  
    ctx3.moveTo(x1, y - 4); ctx3.lineTo(x1, y + 4);  
    ctx3.moveTo(x2, y - 4); ctx3.lineTo(x2, y + 4);  
    ctx3.stroke();  
}
```

```
// Desenarea punctelor geometrice de nod de rețea  
for (let n = d; n <= N; n += d) {  
    let x = startX + (n - 1) * unit;  
    const isActiveIntersection = (highlightedCol === n);
```

```
    if (isActiveIntersection) {  
        ctx3.beginPath();  
        ctx3.fillStyle = (d === 1) ? '#fbbf24' : (d === n) ?  
'#10b981' : '#6366f1';  
        ctx3.arc(x, y, 4.5, 0, Math.PI * 2);  
        ctx3.fill();
```

```
// Desenăm marcajul unghiular sau de intersecție  
tehnică (un mic plus)
```

```
    ctx3.strokeStyle = '#ffffff';  
    ctx3.lineWidth = 0.8;  
    ctx3.beginPath();  
    ctx3.moveTo(x - 6, y); ctx3.lineTo(x + 6, y);  
    ctx3.moveTo(x, y - 6); ctx3.lineTo(x, y + 6);  
    ctx3.stroke();  
    }  
}
```

```
// Indicator formal de cotă pe axa y (rândurile d)  
for (let d = 1; d <= N; d++) {
```

```

        if (d === 1 || d === N || d % 5 === 0 || (highlightedCol &&
highlightedCol % d === 0)) {
            let y = startY + (d - 1) * unit;
            ctx3.fillStyle = (highlightedCol && highlightedCol % d
=== 0) ? '#fbbf24' : '#64748b';
            ctx3.font = 'bold 9px monospace';
            ctx3.fillText(`y=${d}`, startX - 35, y + 3);
        }
    }
}

```

// Desenare unită

```

function drawAll(highlightedCol = null) {
    drawBeads(highlightedCol);
    drawSegments(highlightedCol);
    drawSegments2(highlightedCol);
}

```

// Logica mouse interactivă pentru ambele canvas-uri
(Sincronizare perfectă)

```

function handleInteraction(e, canvasObj) {
    const rect = canvasObj.getBoundingClientRect();
    const mouseX = e.clientX - rect.left;
    const mouseY = e.clientY - rect.top;

```

```

    let closestCol = null;
    let minDistance = 15;

```

```

    for (let n = 1; n <= N; n++) {
        let x = startX + (n - 1) * unit;
        let dist = Math.abs(mouseX - x);

```

```

        if (dist < minDistance && mouseY >= startY - 15 &&
mouseY <= startY + (n - 1) * unit + 10) {
            closestCol = n;
            minDistance = dist;
        }
    }
}

```

```

    }

    if (closestCol !== hoveredNum) {
        hoveredNum = closestCol;
        drawAll(hoveredNum);
        updateInspector(hoveredNum);
    }
}

```

```

    canvas1.addEventListener('mousemove',      (e)      =>
handleInteraction(e, canvas1));
    canvas2.addEventListener('mousemove',      (e)      =>
handleInteraction(e, canvas2));
    canvas3.addEventListener('mousemove',      (e)      =>
handleInteraction(e, canvas3));

```

```

const handleLeave = () => {
    hoveredNum = null;
    drawAll(null);
    updateInspector(null);
};

```

```

canvas1.addEventListener('mouseleave', handleLeave);
canvas2.addEventListener('mouseleave', handleLeave);
canvas3.addEventListener('mouseleave', handleLeave);

```

```

// Actualizare Inspector text dinamic
function updateInspector(num) {
    const defView = document.getElementById('inspector-
default');
    const dispView = document.getElementById('inspector-
display');
    const huds = document.querySelectorAll('.active-hud');

    if (!num) {
        defView.classList.remove('hidden');
        dispView.classList.add('hidden');
    }
}

```

```

    huds.forEach(hud => {
        hud.textContent = "Niciun număr selectat";
        hud.className = "active-hud font-mono text-slate-400
bg-slate-950 px-2.5 py-1 rounded border border-slate-800 text-[11px]";
    });
    return;
}

defView.classList.add('hidden');
dispView.classList.remove('hidden');

const isNPrime = isPrime(num);
document.getElementById('inspect-val').textContent = num;

const tag = document.getElementById('inspect-tag');
if (num === 1) {
    tag.textContent = "Unitate";
    tag.className = "px-2 py-0.5 rounded font-bold uppercase
bg-slate-800 text-slate-300 border border-slate-700";
    huds.forEach(hud => {
        hud.textContent = "Elementul 1 (Originea Unității)";
        hud.className = "active-hud font-mono text-slate-300
bg-slate-800 px-2.5 py-0.5 rounded border border-slate-700 text-[11px]";
    });
} else if (isNPrime) {
    tag.textContent = "Număr Prim";
    tag.className = "px-2 py-0.5 rounded font-bold uppercase
bg-red-950/50 text-red-400 border border-red-800/30";
    huds.forEach(hud => {
        hud.textContent = `Numărul ${num} este PRIM (Vid
geometric interior)`;
        hud.className = "active-hud font-mono text-red-400 bg-
red-950/50 px-2.5 py-0.5 rounded border border-red-800/30 text-
[11px]";
    });
} else {
    tag.textContent = "Număr Compus";
}

```

```

        tag.className = "px-2 py-0.5 rounded font-bold uppercase
bg-blue-950/50 text-blue-400 border border-blue-800/30";
        huds.forEach(hud => {
            hud.textContent = `Numărul ${num} este COMPUS
(Multiple intersecții)`;
            hud.className = "active-hud font-mono text-blue-400
bg-blue-950/50 px-2.5 py-0.5 rounded border border-blue-800/30 text-
[11px]";
        });
    }

    let divizori = [];
    for (let d = 1; d <= num; d++) {
        if (num % d === 0) divizori.push(d);
    }

    const divContainer = document.getElementById('inspect-
divs');
    divContainer.innerHTML = "";
    divizori.forEach(d => {
        const span = document.createElement('span');
        const isDPrime = isPrime(d);
        span.className = `px-2 py-0.5 rounded text-xs font-mono
font-bold ${
            d === 1 ? 'bg-amber-950 text-amber-400 border border-
amber-800/30' :
            d === num ? 'bg-emerald-950 text-emerald-400 border
border-emerald-800/30' :
            isDPrime ? 'bg-red-950/30 text-red-400 border border-
red-900/30' : 'bg-slate-800 text-slate-300'
        }`;
        span.textContent = d;
        divContainer.appendChild(span);
    });

    const eqContainer = document.getElementById('inspect-
equations');

```

```
    eqContainer.innerHTML = "  
    divizori.forEach(d => {  
        const k = num / d;  
        const divLine = document.createElement('div');  
        divLine.innerHTML = `Pas de ${d} × <span class="text-  
amber-400 font-bold">${k}</span> = ${num}`;  
        eqContainer.appendChild(divLine);  
    });  
  
    document.getElementById('inspect-factors').innerHTML =  
    getFactorsText(num);  
    }  
  
    // Pornirea inițială a ambelor module  
    drawAll();  
</script>  
</body>  
</html>
```