

```

void drawLine_basic(vertex2D v1, vertex2D v2,
color3i color){
    [...]
    int d = 2 * dy - dx;
    int incE = 2 * dy;
    int incNE = 2 * (dy - dx);
    [...]
    while (x < v2.x){
        if (d <= 0){ // Escolhe E
            d += incE; // d += 2 * dy
            x++; // Percorre a reta em X
        } else {
            /* Escolhe NE */
            d += incNE; // d += 2 * (dy - dx)
            x++; // Percorre a reta em X
            y++; // Percorre a reta em Y
        }
        [...]
    }
    [...]
}

```

Soma (2 * dy) ou [2 * (dx - dy)] a cada iteração.

```

void drawLine_low(int x1, int y1, int x2, int y2,
color3i color){
    [...]
    int d = 2 * dy - dx;
    [...]
    // Pré-cálculos. Aceleram a renderização.
    int twodx = 2 * dx;
    int twody = 2 * dy;
    [...]
    for(x = x1; x <= x2; x++){
        [...]
        f(d > 0) {
            y += incY; // y++ ou y--
            d -= twodx; // d -= 2 * dx
        }
        d += twody; // d += 2 * dy
    }
    [...]
}

```

Soma (2 * dy) a cada iteração. Subtrai (2 * dx) quando necessário.

$$incE = twody$$

$$\begin{aligned}
 incNE &= 2 \cdot (dy - dx) \\
 incNE &= 2dy - 2dx \\
 incNE &= twody - twodx
 \end{aligned}$$

Assim, ambas são equivalentes.