matfyz

Computer Graphics Charles University

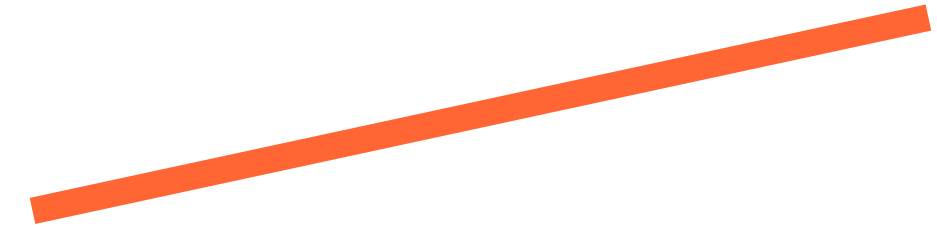
Kreslení čar

CGG MFF UK Praha

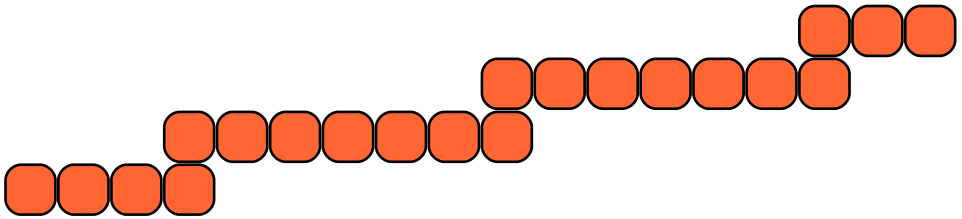
pepca@cgg.mff.cuni.cz

https://cgg.mff.cuni.cz/~pepca/

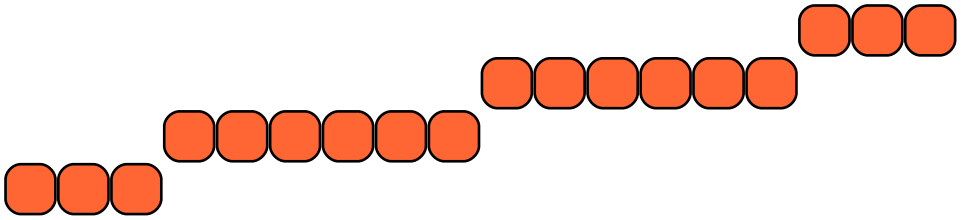
Kreslení úseček



vektorové zařízení

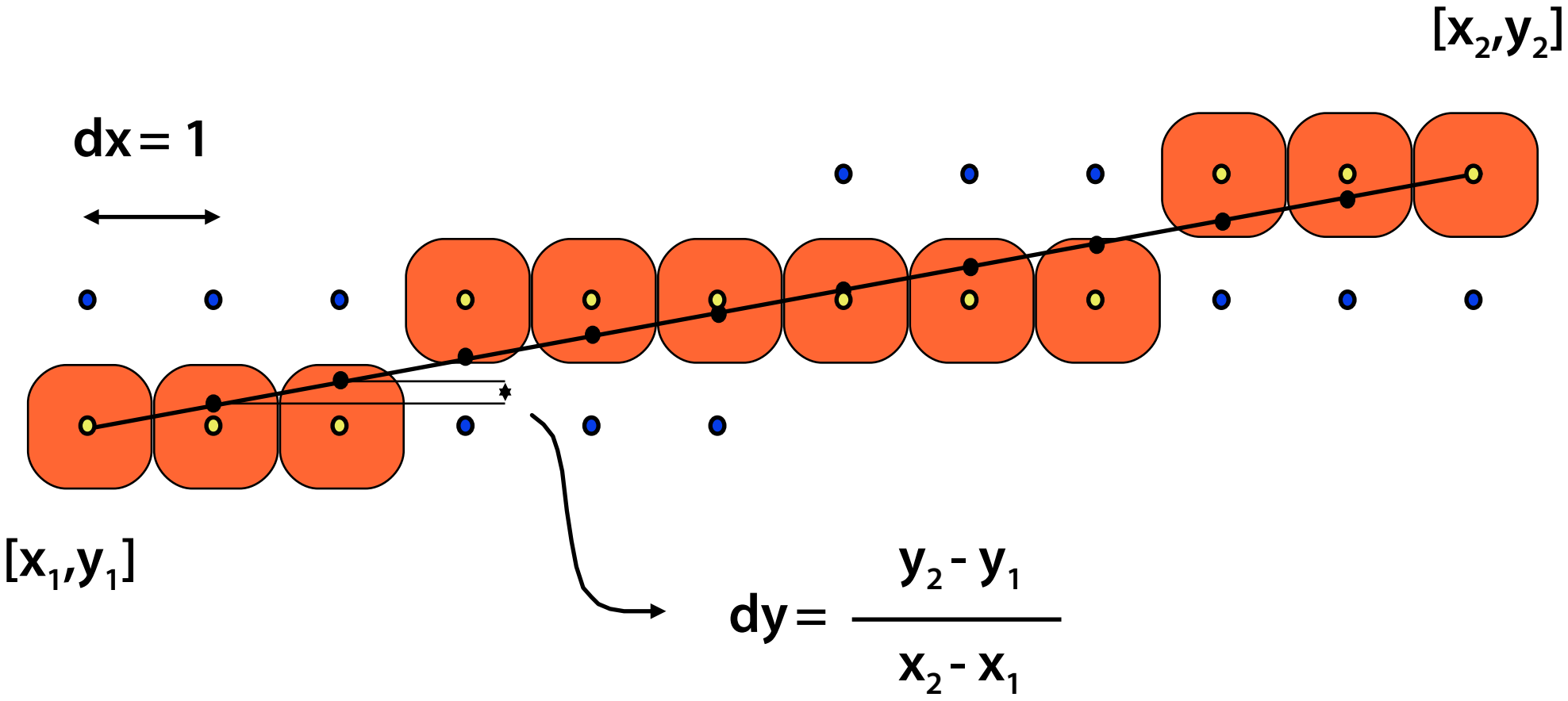


rastrové zařízení



OK

DDA algoritmus



DDA algoritmus

```
void LineDDA (int x1, int y1, int x2, int y2, Color color)
  // předpoklady: x1 < x2, |y2-y1| < |x2-x1|
{
  float y = y1;
  float dy = (y2 - y1) / (x2 - x1);
  PutPixel(x1, y1, color);
  while (x1 < x2)
  {
    x1++;
    y += dy;
    PutPixel(x1, round(y), color);
  }
}
```

DDA algoritmus

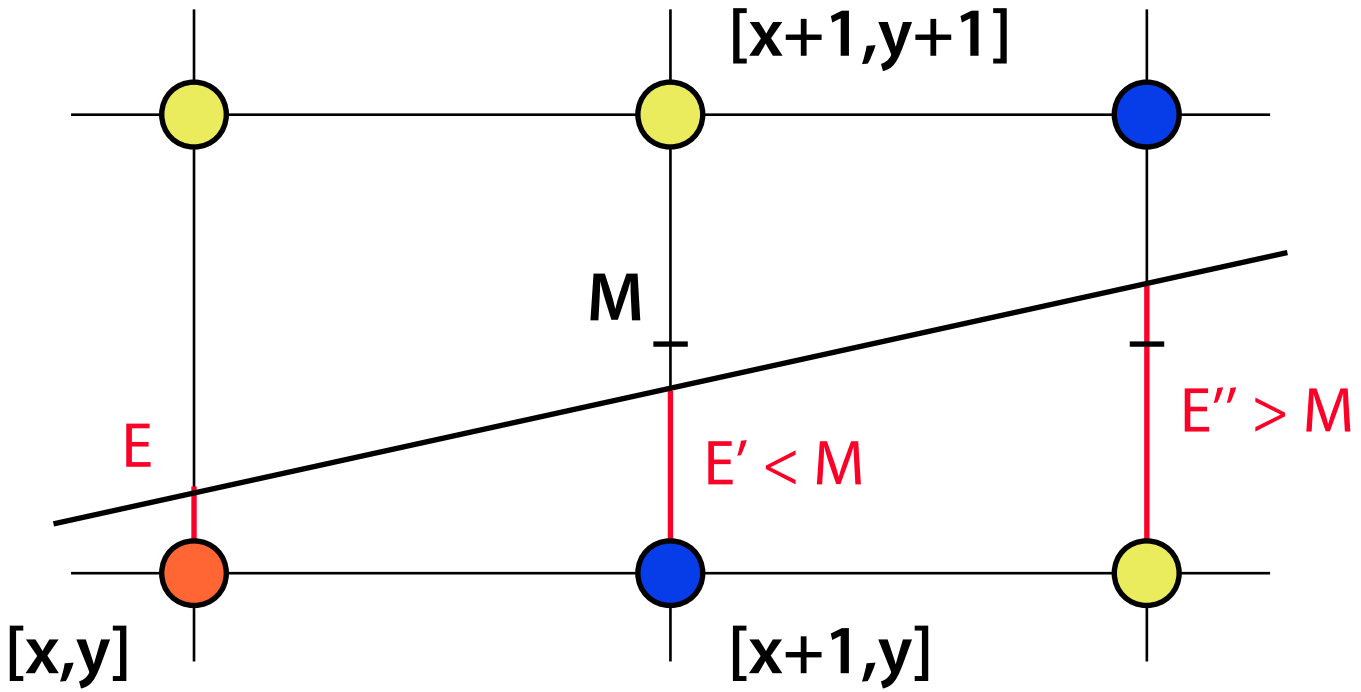
Výhody

- snadná implementace (HW)
- bez větvení kódu

Nevýhody

- nutno počítat s vyšší **přesností** (float, double, fixed point)
- jedno **dělení** a v cyklu **zaokrouhlování**

Bresenhamův algoritmus



$$dx = x_2 - x_1$$

$$dy = y_2 - y_1$$

$$E' = E + \frac{dy}{dx} \leq M = \frac{1}{2}$$

Celočíselné odvození

$$E' = E + \frac{dy}{dx} \leq \frac{1}{2} \quad / \cdot 2dx$$

$$2dx \cdot E' = 2dx \cdot E + 2dy \leq dx \quad / - dx$$

$$dx(2E' - 1) = dx(2E - 1) + 2dy \leq 0$$

$$D' = D + 2dy \leq 0$$

$$D_0 = 2dy - dx$$

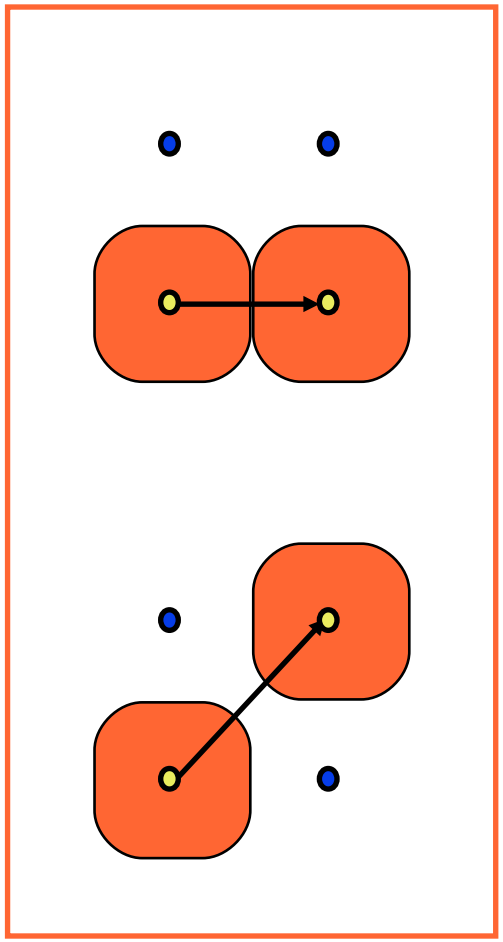
$$D \leq 0 \Rightarrow D' = D + 2dy, \qquad y' = y$$

$$D > 0 \Rightarrow D' = D + 2dy - 2dx, \qquad y' = y + 1$$

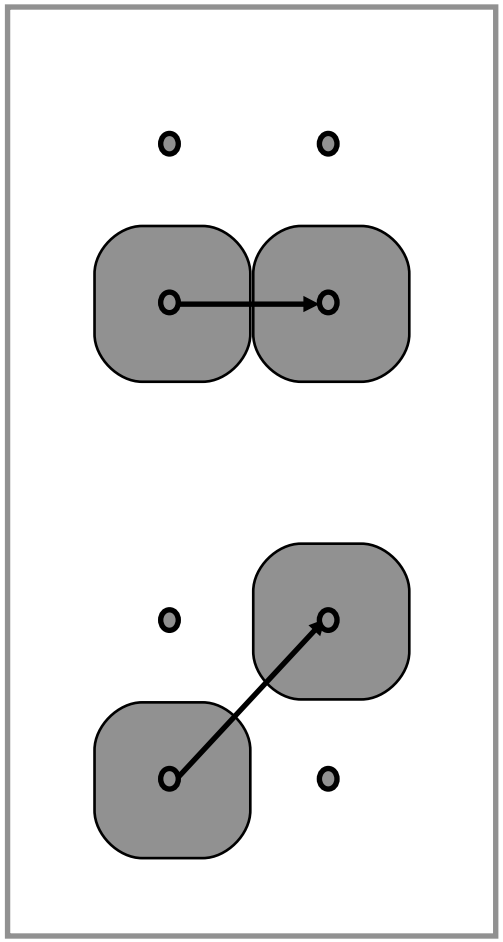
Bresenhamův algoritmus

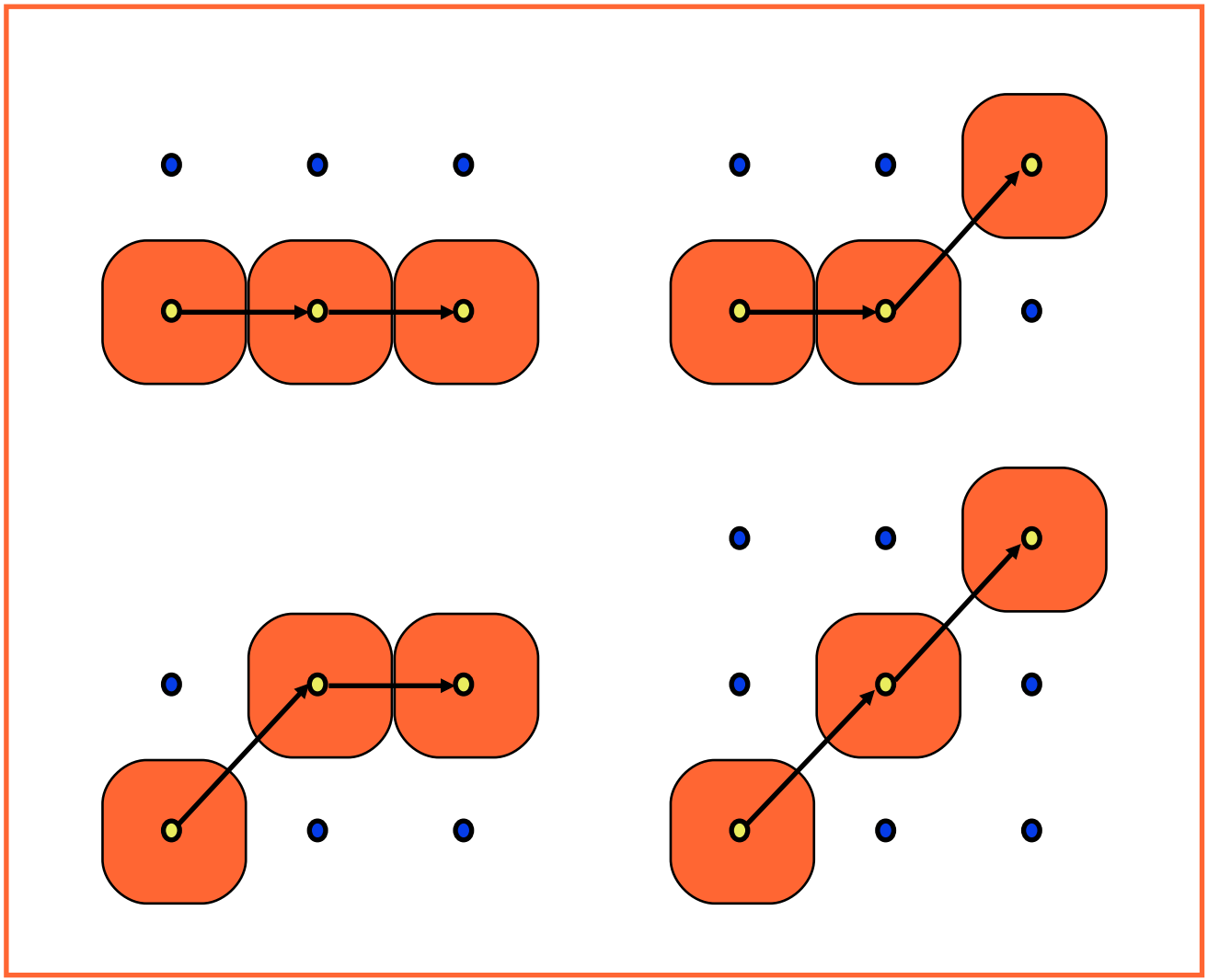
```
void LineBres (int x1, int y1, int x2, int y2, Color color)
   // předpoklady: x1 < x2, |y2-y1| < |x2-x1|
{
   int dx   = x2 - x1;
   int dy   = y2 - y1;
   int D    = 2 * dy - dx;
   int inc0 = 2 * dy;
   int inc1 = 2 * (dy - dx);
   PutPixel(x1, y1, color);
   while (x1 < x2)
   {
     if (D <= 0)
       D += inc0;
     else
     {
       D += inc1;
       y1++;
     }
     x1++;
     PutPixel(x1, y1, color);
   }
}
```

Jednokrokový algoritmus

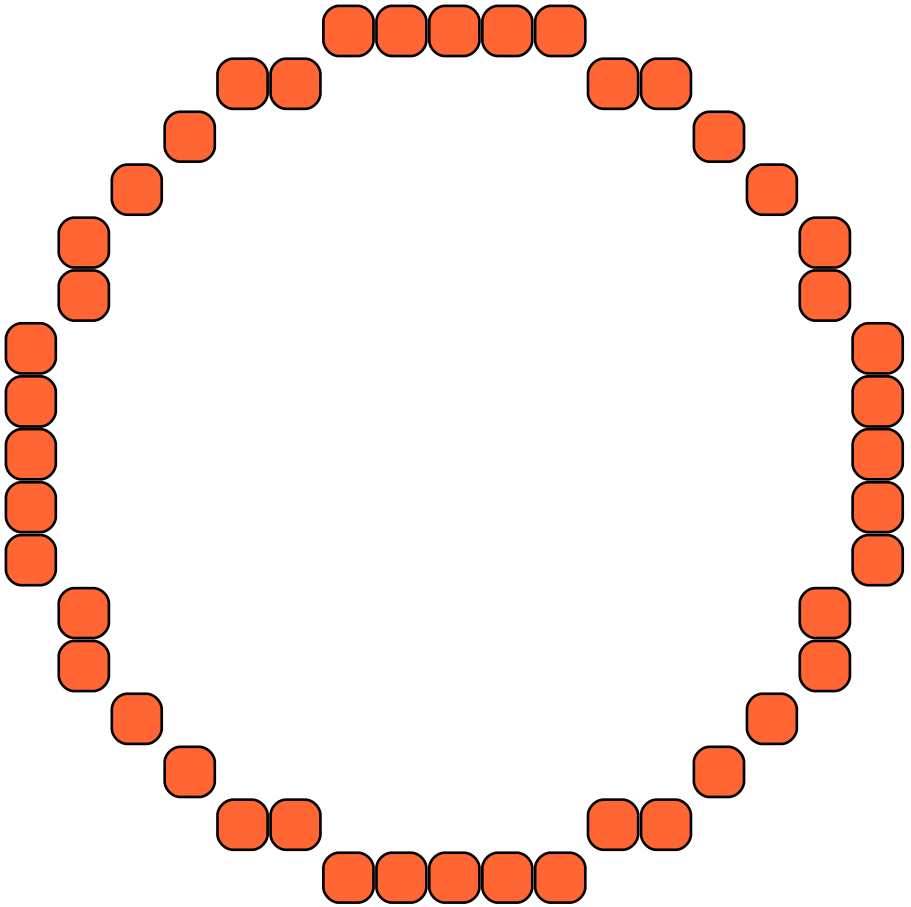


Vícekrokové algoritmy

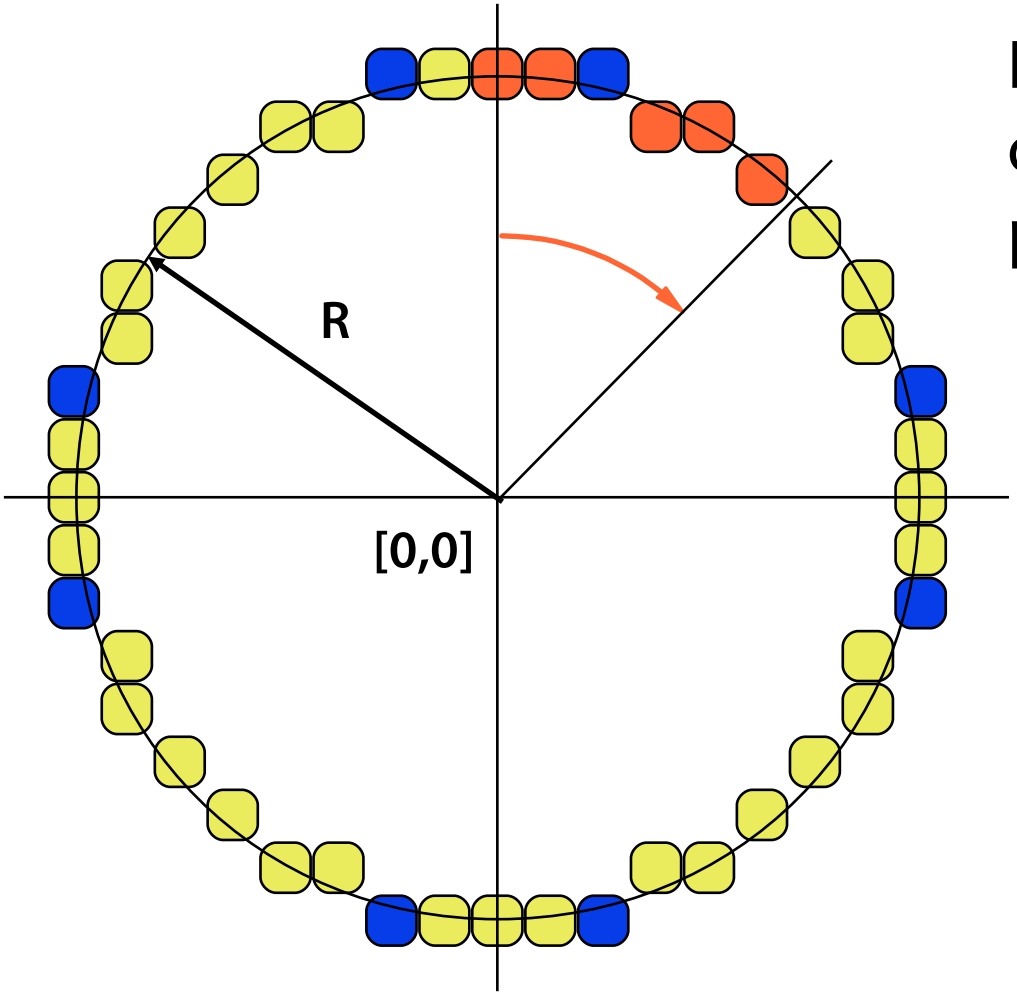




Kreslení kružnice

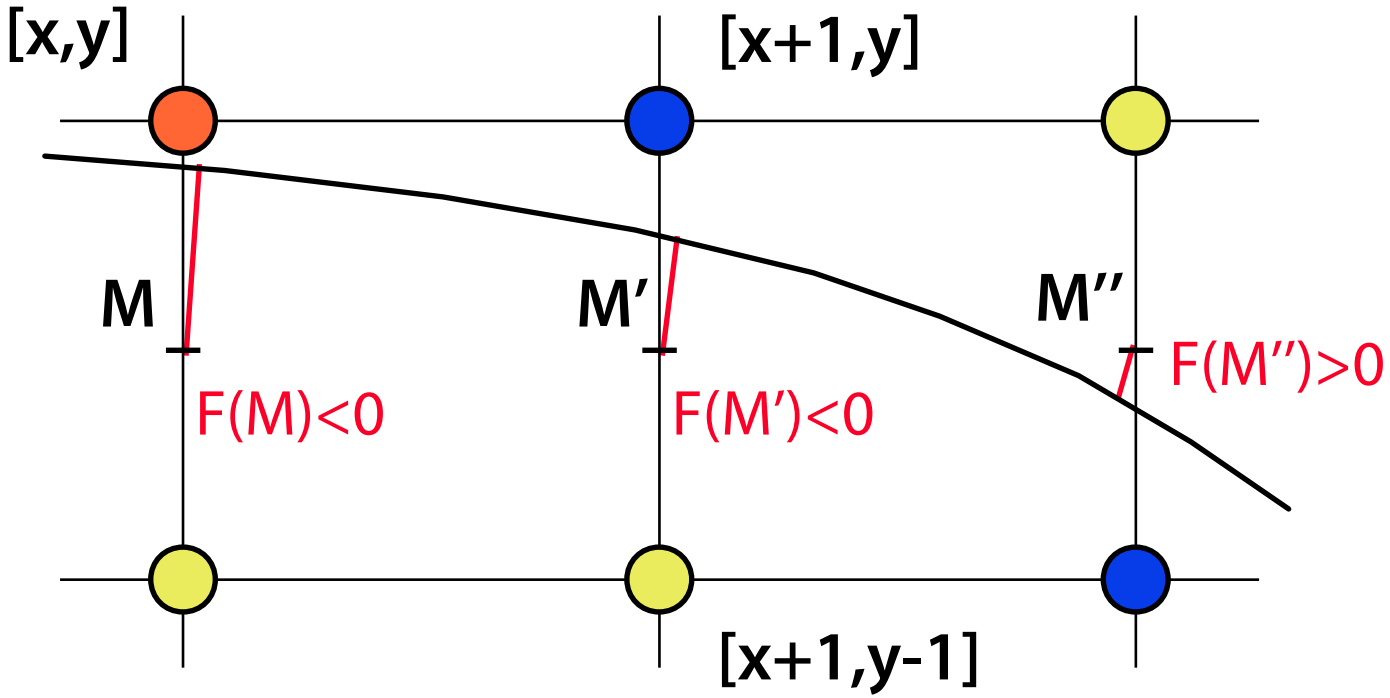


Kreslení kružnice



Kreslí se jen jedna **osmina** oblouku – zbytek se přenese pomocí symetrií

Bresenhamův algoritmus



$$F(M) = M_x^2 + M_y^2 - R^2$$

Inkrementální odvození

1) $F(M') = (x+1)^2 + (y-\frac{1}{2})^2 - R^2 \quad < \quad 0$

$F(M'') = (x+2)^2 + (y-\frac{1}{2})^2 - R^2 = F(M') + 2x + 3$

2) $F(M') \geq 0$

$F(M'') = (x+2)^2 + (y-\frac{3}{2})^2 - R^2 = F(M') + 2x - 2y + 5$

$D_0 = 1.25 - R \quad \{1 - R\}$

$D < 0 \Rightarrow D' = D + 2x + 3, \qquad y' = y$

$D \geq 0 \Rightarrow D' = D + 2x - 2y + 5, \qquad y' = y - 1$

Kreslení kružnice – pomocná funkce

```
void CirclePoints (int x0, int y0, int x, int y, Color color)
{
  PutPixel(x0 + x, y0 + y, color);
  PutPixel(x0 + y, y0 + x, color);
  PutPixel(x0 + x, y0 - y, color);
  PutPixel(x0 + y, y0 - x, color);
  PutPixel(x0 - x, y0 + y, color);
  PutPixel(x0 - y, y0 + x, color);
  PutPixel(x0 - x, y0 - y, color);
  PutPixel(x0 - y, y0 - x, color);
}
```

Kreslení kružnice

```
void CircleBres (int x0, int y0, int R, Color color)
{
  int x = 0;
  int y = R;
  int D = 1 - R;
   CirclePoints(x0, y0, 0, R, color);
   while (y > x)
   {
     if (D < 0)
        D += 2 * x + 3;
     else
     {
        D += 2 * (x - y) + 5;
        y--;
     }
     x++;
     CirclePoints(x0, y0, x, y, color);
   }
}
```

Literatura

J. Foley, A. van Dam, S. Feiner, J. Hughes: ***Computer Graphics, Principles and Practice***, 72-87

Jiří Žára a kol.: ***Počítačová grafika, principy a algoritmy***, 91-100, 106-112