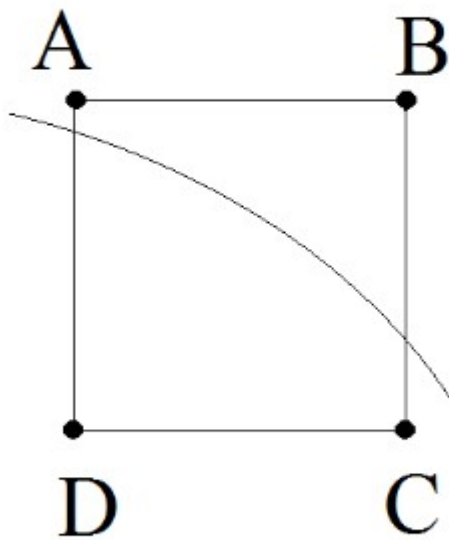


Рисование заданного эллипса алгоритмом Брезенхема.

При построении эллипса будем рассчитывать только точки 1/4 части окружности.

После расчета точки с координатами (x, y) будем рисовать 4 точки:
 (x, y) , $(x, -y)$, $(-x, y)$, $(-x, -y)$.

Шаг итерации

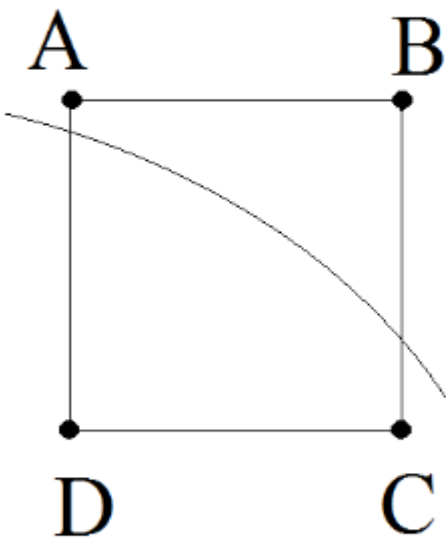


Пусть построена точка A с координатами (x, y) . Рассмотрим какую из точек B, C или D брать на следующем шаге.

Введем величину
$$\Delta = b^2 \cdot (x+1)^2 + a^2 \cdot (y+1)^2 - a^2 b^2$$

Если $\Delta < 0$, то точка C внутри эллипса, т.е. выбор между B и C, иначе выбор между D и C.

Выбор между B и C



Расстояние от точки B равно

$$b^2 \cdot (x+1)^2 + a^2 \cdot y^2 - a^2 b^2 = b^2 \cdot (x+1)^2 + a^2 \cdot (y+1)^2 - a^2 b^2 - 2a^2 y - a^2 = \Delta - 2a^2 y - a^2$$

Так как C внутри эллипса, т.е. $\Delta < 0$, то от точки C расстояние равно $-\Delta$.

Требуется сравнить
$$\text{if } (2\Delta - 2a^2 y - a^2 > 0) \text{ } ++y; \text{ //C}$$

иначе точка B и в любом из этих случаев $++x$;

Выбор между C и D

Аналогично, в случае положительного Δ , т.е. когда точка C находится вне эллипса, выбираем между C и D.

Расстояние от точки D равно

$$b^2x^2 + a^2(y+1)^2 - a^2b^2 = b^2(x+1)^2 + a^2(y+1)^2 - a^2b^2 - 2b^2x - b^2 = \Delta - 2b^2x - b^2$$

Расстояние от точки C равно Δ .

Требуется сравнить

if $(2\Delta - 2b^2x - b^2 < 0) ++x$; //C

иначе точка D и в любом из этих случаев
 $++y$;

Получаем следующий алгоритм:

```
x = 0
y = -math.sqrt(b_sq)
delta = a_sq + b_sq - 2*a_sq*b
while y <= 0:
    qp.drawPoint(QPoint(150 + self.rotate_x(dx + x, dy + y), 350 + self.rotate_y(dx + x, dy + y)))
    qp.drawPoint(QPoint(150 + self.rotate_x(dx + x, dy - y), 350 + self.rotate_y(dx + x, dy - y)))
    qp.drawPoint(QPoint(150 + self.rotate_x(dx - x, dy + y), 350 + self.rotate_y(dx - x, dy + y)))
    qp.drawPoint(QPoint(150 + self.rotate_x(dx - x, dy - y), 350 + self.rotate_y(dx - x, dy - y)))
    if delta < 0:
        if 2 * (delta - a_sq*y) > a_sq:
            y += 1
            delta += a_sq*(2*y + 1)
        x += 1
        delta += b_sq*(2*x+1)
    else:
        if 2 * (delta - b_sq * x) < b_sq:
            x += 1
            delta += b_sq*(2*x + 1)
        y += 1
        delta += a_sq * (2 * y + 1)
```