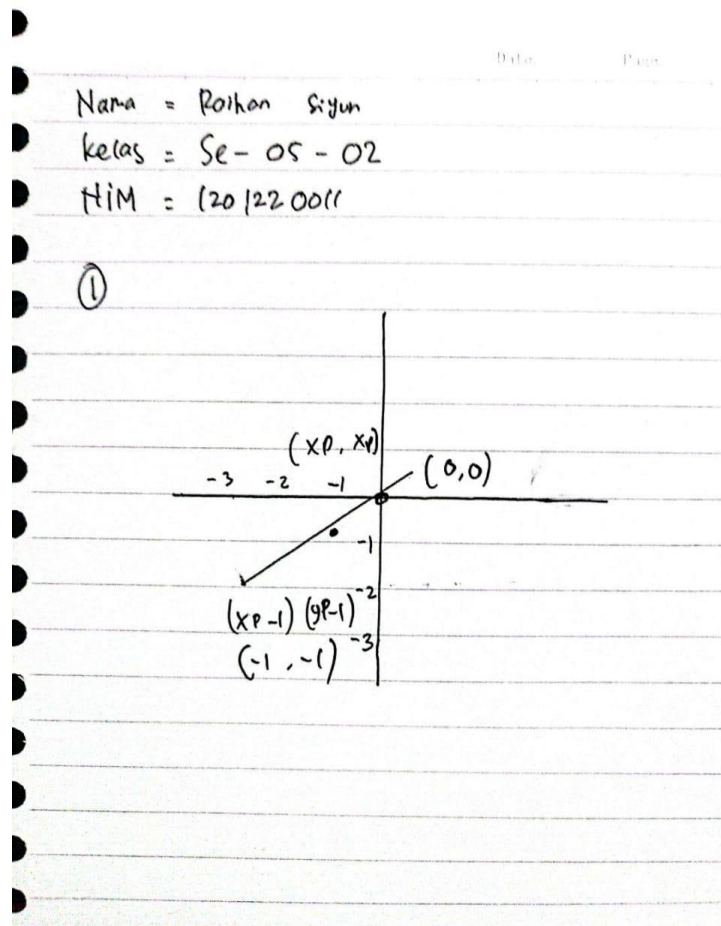


3.1 Derive the midpoint algorithm for drawing lines with a slope between -1 and 0 .



3.2 Apply the structural algorithm in section 3.3 to draw the line in figure 3.6.

1. Tentukan dua titik awal dan akhir garis, yang disimbolkan sebagai (x_0, y_0) dan (x_1, y_1)
2. Hitung perbedaan nilai antara koordinat x dan y dari kedua titik, disebut $dx = x_1 - x_0$ dan $dy = y_1 - y_0$
3. Hitung kemiringan garis dengan cara membagi perubahan nilai y dengan perubahan nilai x, yaitu dy/dx
4. Gunakan persamaan garis umum $y = m \cdot x + b$ untuk menghitung nilai b
5. Mulai dari titik awal (x_0, y_0) lakukan langkah berikut:
 - 5.1. Hitung nilai decision variable (D) menggunakan algoritma midpoint
 - 5.2. Pilih piksel berikutnya yang akan digambar berdasarkan nilai D
 - 5.3. Perbarui posisi piksel dan nilai D dan ulangi langkah hingga mencapai titik akhir (x_1, y_1)

Dengan menggunakan teknik pengurangan alternatif untuk menentukan piksel yang akan digambar dan menghindari penggunaan operasi pecahan, komputasi bisa dilakukan menggunakan operasi bilangan bulat saja. Dalam menggambarkan garis dengan menggunakan algoritma struktural pada gambar 3.6, langkah-langkahnya adalah menghitung perubahan nilai x

(dx), perubahan nilai y (dy), serta kemiringan garis, kemudian menggunakan persamaan garis untuk menentukan piksel berikutnya berdasarkan nilai variabel keputusan.

3.3 Extend the program GeneralPathCar.java for drawing the car of figure 2.10. Show the control points for the quadratic and cubic curves and connect the endings of the curves with their corresponding control points by dashed lines.

```

ar
urveTo(x1:195,y1:100,x2:200,y2:80,x3:160,y3:80);
of
ineTo(x:110,y:80);
ndscreen
ineTo(x:90,y:100);
nnet
ineTo(x:60,y:100);
ont
ineTo(x:60,y:120);

aw the car.
draw(gp);

setStroke(new BasicStroke(width:1.0f)),
// Draw a coordinate system.
drawSimpleCoordinateSystem(xmax:200, ymax:150, g2d);
```

