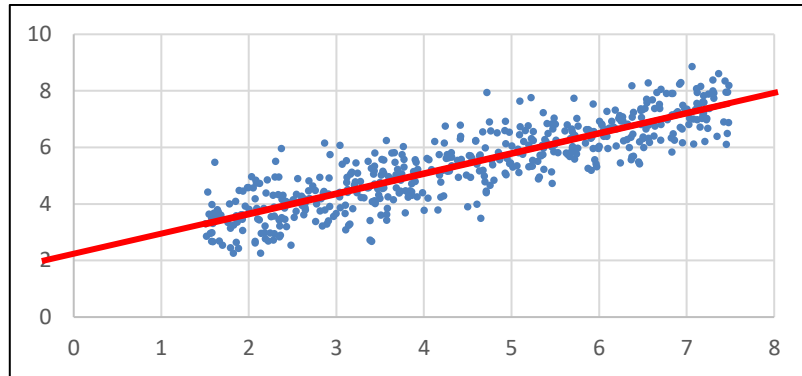


Simple Linear Regression

กำหนดให้มีชุดพิกัด $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ เราต้องการหาฟังก์ชันเชิงเส้น $\hat{y} = mx + b$ ที่ทำให้ $\sum_{i=1}^N (y_i - \hat{y}_i)^2$ มีค่าน้อยที่สุด กราฟข้างล่างนี้แสดงตัวอย่างชุดพิกัด และเส้นสีแดงที่แทนฟังก์ชัน $\hat{y} = 0.714x + 2.204$ ที่มีคุณสมบัติตามที่ต้องการ



เราสามารถใช้การวิเคราะห์ถดถอย (ขอไม่อธิบายรายละเอียด) เพื่อหาฟังก์ชันถดถอยเชิงเส้น $\hat{y} = mx + b$ ที่ต้องการจากชุดข้อมูล x_i, y_i ต่าง ๆ ได้ โดยค่า m (ซึ่งเป็นความชัน) และ b (ซึ่งเป็นจุดตัดแกน y) หาได้จากสองสมการข้างล่างนี้

$$m = \frac{(N \sum_{i=1}^N (x_i y_i)) - (\sum_{i=1}^N x_i)(\sum_{i=1}^N y_i)}{(N \sum_{i=1}^N x_i^2) - (\sum_{i=1}^N x_i)^2}$$

$$b = \frac{(\sum_{i=1}^N y_i) - (m \sum_{i=1}^N x_i)}{N}$$

ให้ใช้ **float** ในการคำนวณ
อย่าใช้ **double**

ให้เขียนโปรแกรมรับชุดข้อมูล x_i และ y_i แล้วแสดงฟังก์ชันถดถอยเชิงเส้นของชุดข้อมูลนี้

ข้อมูลนำเข้า

บรรทัดแรกเป็นจำนวนเต็ม N ตามด้วยคำว่า **mb** หรือ **func**

อีก N บรรทัดต่อมา แต่ละบรรทัดมีจำนวนจริงสองจำนวน x_i และ y_i

ข้อมูลส่งออก

ให้หาฟังก์ชันถดถอยเชิงเส้นของชุดข้อมูลที่ได้รับ

- ถ้าเป็น **mb** ให้แสดงค่า m ตามด้วย b (ค่าละบรรทัด)
- ถ้าเป็น **func** ให้แสดงตัวฟังก์ชันในรูปแบบ $y = mx + b$ โดย
 - ไม่แสดงพจน์ที่มีค่าเป็น 0 (ยกเว้นฟังก์ชัน $y = 0$)
 - ถ้า $m = 1$ ไม่ต้องแสดงค่า m ถ้า $m = -1$ แสดงเฉพาะเครื่องหมายลบ เช่น ไม่แสดง $y = -1x$ ให้แสดง $y = -x$
- ให้แสดงค่า m และ b อย่างมาก 3 หลักหลังจุดทศนิยมด้วยคำสั่ง `round(m*1e3)/1e3` (ต้อง `#include <cmath>` ด้วย)
- ชุดข้อมูลที่ได้รับจะไม่มีกรณีที่ค่า m เป็นอนันต์

ตัวอย่าง

Input (จากแป้นพิมพ์)	Output (ทางจอภาพ)
3 mb 1.0 0.0 2.0 0.0 9.4 0.0	0 0 ค่า m ค่า b
3 mb 1.0 11.0 2.0 12.0 3.0 13.0	1 10

20 mb 1.1881 1.5293 1.7655 1.1602 1.8581 1.1016 2.2834 0.8302 2.7323 0.5438 3.0490 0.3426 3.2191 0.2325 3.5325 0.0332 3.7860 -0.1221 5.8511 -1.4411 6.0823 -1.5966 6.2641 -1.7018 6.6594 -1.9658 6.9622 -2.1554 7.5696 -2.5427 7.6285 -2.5792 7.9083 -2.7581 7.9242 -2.7681 9.6531 -3.8725 9.9108 -4.0347	-0.638 2.288	<div> <p>ข้อแนะนำ</p> <p>อย่าใช้การแปลงค่า float เป็น string แล้วนำไปส่งให้ cout ในภายหลัง เช่น</p> <pre>float a = 2.0/3.0, b = 1.0; a = round(a * 1e3)/1e3; b = round(b * 1e3)/1e3; cout << "*** " << a << " " << b << endl; string s = "*** " + to_string(a) + " " + to_string(b); cout << s << endl;</pre> <p>จะแสดง</p> <pre>** 0.667 1 ** 0.667000 1.000000</pre> <p>แบบหลังไม่ตรงตามโจทย์</p> </div>
3 func 1.0 0.0 2.0 0.0 9.4 0.0	$y = 0$	
3 func 1.0 -3.0 2.0 -3.0 4.4 -3.0	$y = -3$	
3 func 1.0 1.0 2.0 2.0 3.0 3.0	$y = x$	
3 func 1.0 11.5 2.0 12.5 3.0 13.5	$y = x + 10.5$	
3 func 1.0 -1.0 2.0 -2.0 3.0 -3.0	$y = -x$	
3 func 3.0 1.0 2.0 2.0 1.0 3.0	$y = -x + 4$	
3 func -3.0 -1.0 -2.0 -2.0 -1.0 -3.0	$y = -x - 4$	
20 func -47.4275 -643.9849 -43.2994 -591.5547 -43.1769 -589.5735 -32.6425 -453.3533 -28.2735 -412.2329 -20.1902 -299.63 -6.3605 -140.4817 -5.6405 -128.4213 3.9611 -8.7905 4.7316 2.7213 4.1944 6.8667 16.9359 153.0378 19.2241 181.903 22.1265 222.7692 23.7411 227.653 34.6474 382.2777 41.1725 439.1405 39.5836 445.2155 47.5559 528.9183 47.6268 533.7816	$y = 12.365x - 55.714$	<div> <p>ใส่ช่องว่างให้ถูกต้องตามตัวอย่าง</p> <p>จากตัวอย่างนี้ หากแสดงต่างจากนี้ถือว่าผิด เช่น</p> <pre>y=12.365x -55.714 y = 12.365 x - 55.714</pre> </div>

ชุดข้อมูลทดสอบ

- กรณีทดสอบทั้งหมด 24 กรณี (10 กรณีทดสอบแบบ **mb** กับ 14 กรณีทดสอบแบบ **func**) แบ่งเป็น
 - 6 กรณีที่ slope ของฟังก์ชันที่หาได้ ขนานกับแกน x
 - 8 กรณีที่ slope ของฟังก์ชันที่หาได้ ทำมุม 45° หรือ 135° กับแกน x
 - 10 กรณีแบ่งเป็น
 - มี 2 กรณีที่ slope เป็นบวก ตัดแกน y เหนือจุดกำเนิด
 - มี 2 กรณีที่ slope เป็นบวก ตัดแกน y เหนือจุดกำเนิด
 - มี 2 กรณีที่ slope เป็นลบ ตัดแกน y ใต้จุดกำเนิด
 - มี 4 กรณีที่ slope เป็นลบ ตัดแกน y ใต้จุดกำเนิด