**DO NOT USE A RULER**

TASK ONE Draw these straight lines (2, 17) to (10, 17), (18, 2) to (18, 14), (1, 5) to (8, 12)



**TASK TWO** Can you draw and join the lines together?



**TASK 3** **Bresenham's Line Algorithm**

A = 2\*change in Y value

B = A-2\*change in X value

P = A- change in X value

**TASKS ONE WORK OUT values for A B and P**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **CHANGE IN Y VALUE** | **CHANGE IN X VALUE** | **X2** | **A -** | **ANSWER** |
| **A** | 5 |  | 10 |  | 10 |
| **B** |  |  |  |  | -16 |
| **P** |  |  |  |  | -3 |
|  |  |  |  |  |  |
| **p1** | **A =10** | **B =** | **P =-3** | P less 0? | Y / N |
| **P2** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P3** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P4** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P5** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P6** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P7** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P8** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P9** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **P10** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **p11** | **A =** | **B =** | **P =** | P less 0? | Y / N |
| **p12** | **A =** | **B =** | **P =** | P less 0? | Y / N |

**Fill the starting pixel. Then for every position along the X axis**:

* if P is less than 0, draw the new pixel on the same line as the last pixel, and add A to P.
* if P was 0 or greater, draw the new pixel one line higher than the last pixel, and add B to P.
* Continue this process until we reach the end of the line.



**EXTENSION**

**Five** Circles Midpoint Circle Algorithm

E = -Radius

X = Radius

Y = 0

Until Y becomes greater than X, repeat the following rules in order:

* Fill the pixel at coordinate (X + Centre X, Y + Centre Y)
* Increase E by 2 \* Y + 1
* Increase Y by 1
* If E is greater than or equal to 0, subtract (2X - 1) from E, and then subtract 1 from X.

Follow the rules to draw a circle on the grid, using C as the centre of the circle, and R as a point at radius distance from the centre:

