

Les représentations graphiques de 2 fonctions linéaires vont être représentées pour visualiser qu'elles se coupent, puis évaluer les coordonnées du point d'intersection.

Javascript n'est pas particulièrement adapté mais avec Javascript en Ligne on peut quand même s'en sortir avec un peu d'astuce pratique.

Avec quoi écrire l'algorithme

Dans l'éditeur <http://emmanuel.ostenne.free.fr/mepirem/algo/> (en haut à droite, effacer l'exemple proposé)

Écrire l'algorithme

<pre>//fonctions function f1(x) { return -Math.PI*x/4+Math.cos(2); } function f2(x) { return 2*x+3; }</pre>	<p>On définit les 2 fonctions linéaires avec 2 fonctions javascript f1 et d2 pour les droites (D1) et (D2)</p>
<pre>//repère (à ajuster) var xmin=-3; var xmax=1; var ymin=-1; var ymax=3; var r=new repere(xmin, ymin, xmax, ymax, 200, 200); afficher("Repère:",xmin,"<x<",xmax," et ",ymin,"<y<", ymax);</pre>	<p>On prépare l'objet repère r pour la zone graphique : 200,200 est la taille en pixels (largeur, hauteur) du rectangle blanc affiché dans la « sortie du programme ». L'unité est affichée automatiquement.</p>
<pre>//tracés var x=0, y=0; //Tracé du morceau de (d1) r.couleurTrait("blue"); x=-5; y=f1(x); r.allerEn(x,y); x=5; y=f1(x); r.ligneVers(x,y); //Tracé du morceau de (d2) r.couleurTrait("green"); x=-5; y=f2(x); r.allerEn(x,y); x=5; y=f2(x); r.ligneVers(x,y);</pre>	<p>x,y pour mémoriser les coordonnées d'un point obtenu par une fonction</p> <p>En bleu, ce sera (D1) de f1 par 2 valeurs choisies pour x : -5 et 5 qui donnent 2 valeurs de y ← on met le crayon graphique au 1er point ← on trace la ligne jusqu'au 2nd point</p> <p>Même principe pour (D2) de f2 avec 2 valeurs de x : -5 et 5 qui donnent 2 valeur de y ← on met le crayon au 1er point ← on trace la ligne reliant le 2nd point</p>
<pre>//P (à ajuster) x=1; y=1; r.couleurTrait("red"); r.rectCreux(x-0.01,y-0.01,x+0.01,y+0.01);</pre>	<p>On marque un point pour évaluer les coordonnées de l'intersection ← x et y sont à régler manuellement ← le rectangle permet de mieux voir le point (un tout petit carré)</p>

Copie d'écran

```
//fonctions
function f1(x) {
return -Math.PI*x/4+Math.cos(2);
}

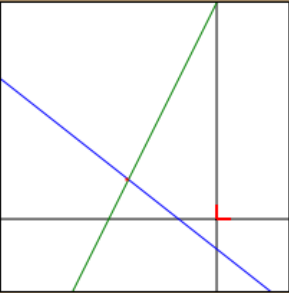
function f2(x) {
return 2*x+3;
}

//repère (à ajuster)
var xmin=-3;
var xmax=1;
var ymin=-1;
var ymax=3;

var r=new repere(xmin, ymin, xmax, ymax, 200, 200);
afficher("Repère:", xmin, "<x<", xmax, " et ", ymin, "<y<", ymax);
```

EXE Indente Efface

Sorties du programme :



le segment rouge représente 0.2 unités en abscisses et 0.2 unités en ordonnées.
Repère: -3 <x< 1 et -1 <y< 3
Exécution en 5 millisecondes

Efface

Exécution de l'algorithme

- appuyer sur le bouton jaune EXE,
- visualiser le point rouge P en (1;1)
- changer les coordonnées x et y de ce point P dans le script (à la fin)
- relancer l'exécution EXE pour visualiser sa nouvelle position
- affiner les coordonnées de P pour approcher l'intersection des 2 droites
- etc.