JONAS Audrey

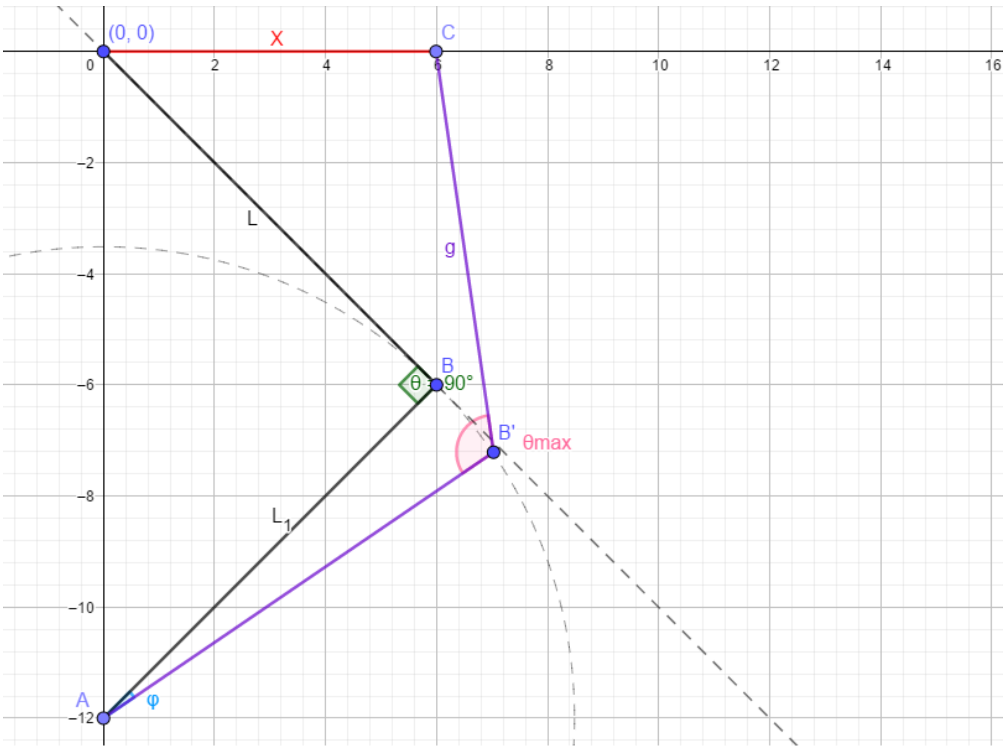
SEANCE 4: LIGNE DROITE

TRAVAIL REALISE PENDANT LES VACANCES :
Il a bien fallu avancer ! Il devenait nécessaire de finaliser le travail sur la ligne droite.

Hypothèses :
- ⇨ Les deux bras ont même longueur L
- ⇨ Les déplacements horizontaux et verticaux doivent avoir même amplitude : amplitude max=2L.

Trait horizontal :

Etape 1: Visualisation



- AO=$\sqrt{2}$L
- AC²=AB'² + B'C² - 2.AB'.B'C.cos(θ)

D'après le théorème d'Al Kashi

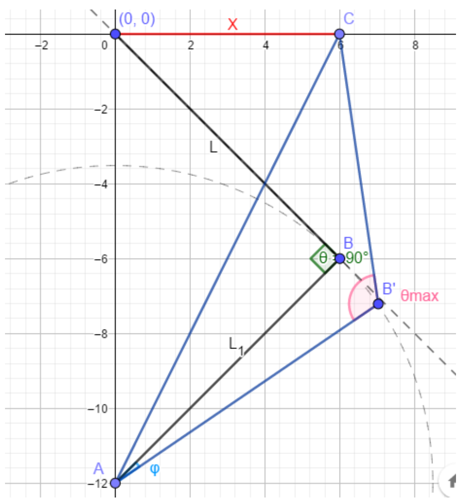
d'où : AC=$\sqrt{2}$L$\sqrt{1-\cos(\theta)}$ = AO$\sqrt{1-\cos(\theta)}$

avec θ appartenant à [π/2 ; π].

Etape 2 : Calcul de θmax

On souhaite obtenir une expression du type : **θmax = f(x).**

Al Kashi au point C :
- ⇨ triangle AB'C
- ⇨ angle θ=θmax
- ⇨ x = déplacement horizontal



(1) $AC^2 = OA^2 + OC^2$

⬄ $AC^2 = 2L^2 + x^2$

⬄ $AC = \sqrt{2L^2 + x^2}$

(2) $AC^2 = AB'^2 + B'C^2 - 2.L^2.\cos(\theta max)$

⬄ $AC^2 = L^2 + L^2 - 2.L^2.\cos(\theta max)$

⬄ $AC^2 = 2 . L^2 (1 - \cos(\theta max))$

(1) , (2) ➔ (3)

$2.L2 + x2 = 2.L2 - 2.L2 \cos(\theta max)$

⬄ $x2 = -2.L2.\cos(\theta max)$

⬄ $x = \sqrt{2}.L.\sqrt{-\cos(\theta max)}$

⬄ $\text{Cos}(\theta max) = -\frac{x^2}{2.L^2}$

Finalement :

$$\theta max = \cos^{-1}\left(-\frac{x^2}{2.L^2}\right)$$

Expression de l'angle à appliquer au coude pour effectuer une distance x : $\theta max - \pi/2$

Le résultat paraît cohérent, lorsque x augmente l'angle augmente.

Etape 3 : Calcul du second angle

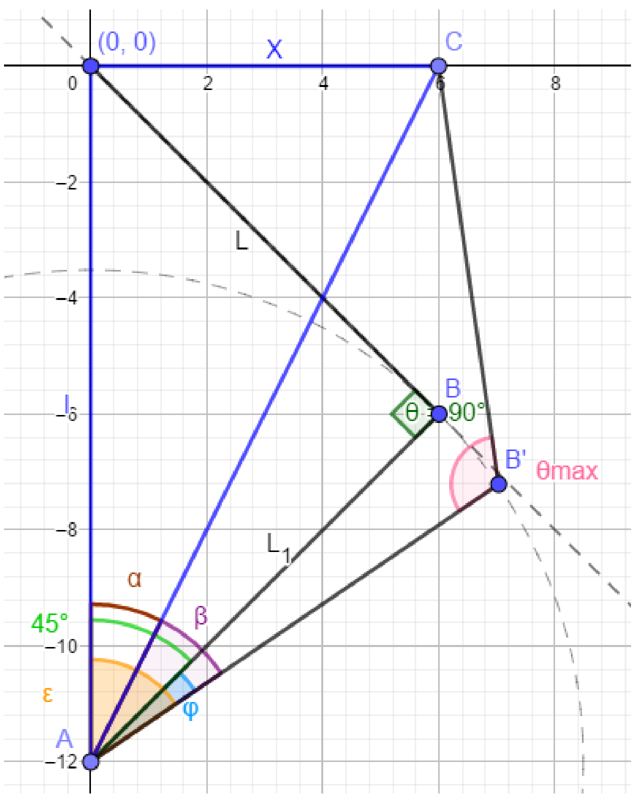
On cherche à exprimer l'angle **φ** , angle de rotation autour du point A

On peut décomposer l'angle ε :

- ⇨ $\varepsilon = 45 + \boldsymbol{\varphi}$
- ⇨ $\varepsilon = \alpha + \beta$

On a :

- ⇨ $OC = x$
- ⇨ $OA = \sqrt{2}\, L$



➔Calcul de α :

Th. Al Kashi dans le triangle AOC :

$OC^2 = OA^2 + AC^2 - 2.OA.AC.\cos(\alpha)$

⬄ $x^2 = 2L^2 + 2L^2 + x^2 - 2(\sqrt{2}L . \sqrt{2L^2 + x^2})\cos(\alpha)$

⬄ $4L^2 = 2\sqrt{2}L\,\sqrt{2L^2 + x^2}\cos(\alpha)$

⬄ $\cos(\alpha) = \frac{\sqrt{2}L}{\sqrt{2L^2+x^2}}$

⬄ $\alpha = \cos^{-1}\left(\frac{\sqrt{2}L}{\sqrt{2L^2+x^2}}\right)$ (1)

➔Calcul de β :

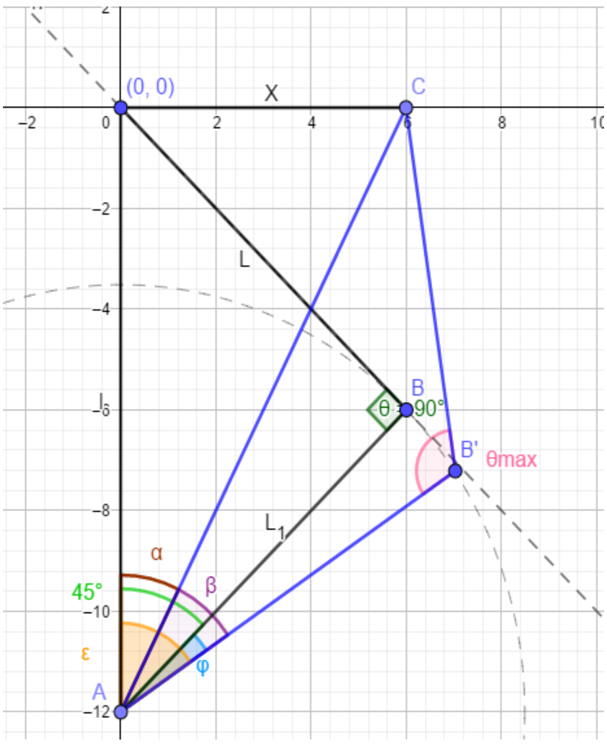
Th. Al Kashi dans le triangle ACB' :

B'C² = AB'² + AC² – 2.AB'.AC.cos(β)

⇔ $L^2 = L^2 + (2L^2 + x^2) - 2L(\sqrt{2L^2+x^2})\cos(\beta)$

⇔ $\cos(\beta) = \frac{\sqrt{2L^2+x^2}}{2L}$

⇔ $\beta = \cos^{-1}\left(\frac{\sqrt{2L^2+x^2}}{2L}\right)$ (2)



➔Calcul de φ :

On peut désormais conclure sachant que : **φ** = α + β – 45 (3)

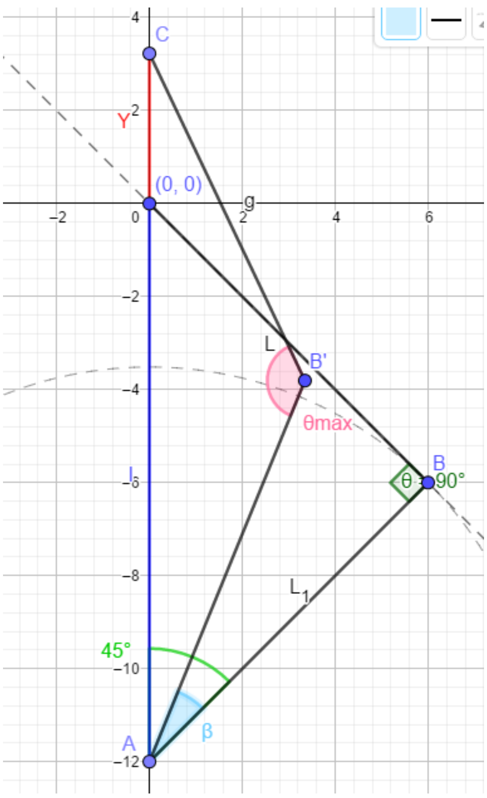
(1) , (2) ➔ (3)

$$\varphi = \cos^{-1}\left(\frac{\sqrt{2}L}{\sqrt{2L^2+x^2}}\right) + \cos^{-1}\left(\frac{\sqrt{2L^2+x^2}}{2L}\right) - 45$$

Expression de l'angle à appliquer à l'épaule pour effectuer une distance x.

Trait horizontal :

Etape 1: Visualisation



Conditions initiales :

⇨ θ0 =π/2

⇨ θmax=π

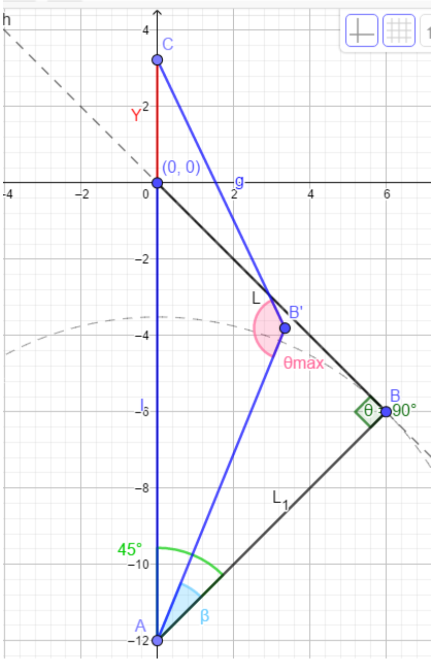
- AO=√2L
- AC=√2L+ y

Etape 2 : Calcul de θmax

On souhaite obtenir une expression du type : **θmax = f(y).**

Al Kashi au point B :

- ⇨ triangle AB'C
- ⇨ angle θ=θmax
- ⇨ y = déplacement vertical



AC² = AB'² + B'C² – 2.AB'.B'C .cos(θmax)

⇔ $(\sqrt{2}L + y)^2 = L^2 + L^2 - 2L^2 \cos(\theta max)$

⇔ $(\sqrt{2}L + y)^2 = 2L^2 - 2L^2 \cos(\theta max$

⇔ $\cos(\theta max) = 1 - \left(\frac{\sqrt{2}L+y}{\sqrt{2}L}\right)^2$

Finalement

⇔ $\theta max = \cos^{-1}\left(1 - \left(\frac{\sqrt{2}L+y}{\sqrt{2}L}\right)^2\right)$

L'angle à appliquer au coude pour effectuer une distance y est : θmax - 90

Etape 3 : Calcul du second angle

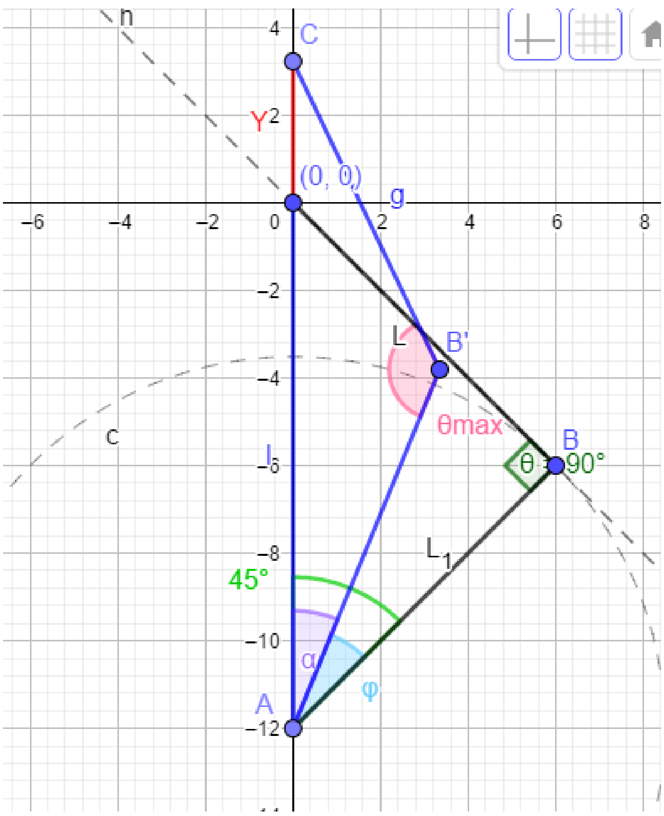
On cherche à exprimer l'angle **φ**, angle de rotation autour du point A

On peut décomposer ainsi :

- ⇨ 45= α + **φ**

On a :

- ⇨ OC = y
- ⇨ OA=$\sqrt{2}\,L$
- ⇨ AC= $\sqrt{2}\,L + y$



Il faut d'abord trouver l'expression de l'angle α.

➔Calcul de α :

B'C² = AB'² + AC² – 2.AB'.AC .cos(α)

⇔ $L^2 = L^2 + (\sqrt{2}L + y)^2$ - 2L.$(\sqrt{2}L + y)$.cos(α)

⇔ cos(α)= $\frac{y+\sqrt{2}L}{2L}$

$\Leftrightarrow \alpha = \cos^{-1}\left(\frac{y+\sqrt{2}L}{2L}\right)$

➔Calcul de φ :

On peut désormais conclure sachant que : $45 = \varphi + \alpha$

$$\varphi = 45 - \cos^{-1}\left(\frac{y+\sqrt{2}L}{2L}\right)$$

Expression de l'angle à appliquer à l'épaule pour effectuer une distance verticale y.

Objectif : L'objectif de cette séance était d'essayer de mettre en pratique la théorie.

J'ai d'abord dû créer un programme me permettant de mettre l'épaule et le bras dans la situation initiale cad perpendiculaires l'un à l'autre et l'épaule formant un angle de 45° par rapport à l'axe Y. J'ai rencontré un premier problème car je sais que les deux bras sont alignés lorsque le coude est à 155°. Je m'attendais donc à les avoir perpendiculaires lorsque je le mets à 65. Et bien non, après différents essais je me suis rendue compte qu'il fallait que je lui applique un angle de 35°.

Vous nous avez conseillé d'appliquer les angles plutôt en Millisecondes pour plus de précision.

J'ai donc crée ma fonction :

```
void initialize(){
  epaule.write(20);
  delay(1000);
  coude.write(35);
  delay(20);
}
```

J'ai ensuite commencé l'écriture de la fonction qui me permet de tracer une ligne droite horizontale.

```
void lineH(float x){
  int L=9;
  teta=conversion(acos(-pow(x,2)/(2*pow(L,2))));
  phi=conversion(acos((sqrt(2)*L)/(sqrt(2*pow(L,2)+pow(x,2))))+acos((sqrt(2*pow(L,2)+pow(x,2))/(2*L))-(PI/4));
  Serial.println(teta);
  Serial.println(phi);
  float j=0;
  for (float i=0; i<phi; i+=0.25){
      Serial.println("OK");
      if(j<teta){
      Serial.println("DAC");
      epauleDroite(i);
      coudeDroite(j);
      j+=0.25;
      Serial.println(j);
    delay(1000);
  }
}
```

La fonction arccosinus de arduino renvoie des angles en radians, j'ai donc écrit une fonction « conversion » qui me permet de les convertir en degré.

```
double conversion(double x){
  return (x*90/(PI/2));
}
```

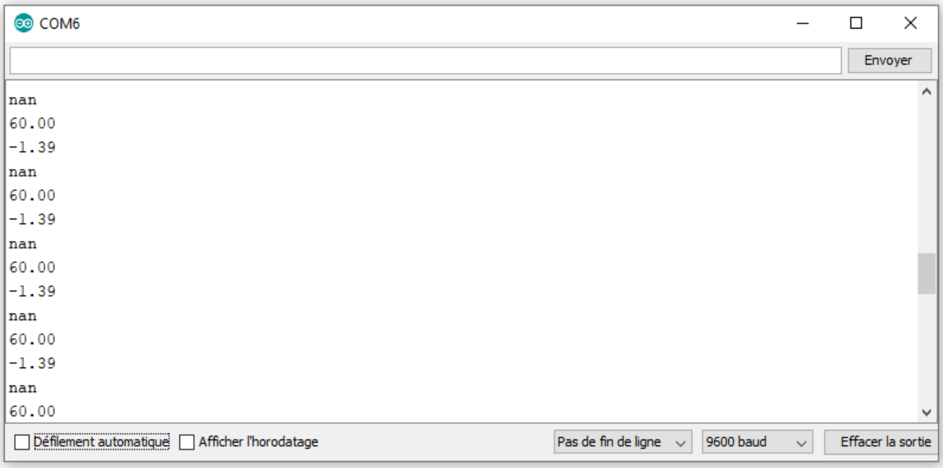
Mais j'avais un problème car sur le moniteur série, l'affichage du calcul de l'angle de l'épaule était « nan » soit Not A Number. Après recherche, je me suis rendue compte que c'était parce que dans mon calcul je n'avais pas converti les 45°en radian ! Dans la formule je ne dois pas soustraire 45 mais π/4. Le résultat n'était donc pas dans le domaine de définition de la fonction.

```
double y;
double w;

double conversion(double x){
  return (x*90/(PI/2));
}

void setup() {
  Serial.begin(9600);
}

void loop() {
  x=conversion(acos(0.5));
  Serial.println(x);
  delay(60);
  w=(-pow(5,2)/(2*9));
  Serial.println(w);
  delay(60);
  y=conversion(acos(-pow(5,2)/(2*9)));
  Serial.println(y);
  }
```




```
void lineH(float x){
  int L=9;
  teta=conversion(acos(-pow(x,2)/(2*pow(L,2))));
  phi=conversion(acos((sqrt(2)*L)/(sqrt(2*pow(L,2)+pow(x,2))))+acos((sqrt(2*pow(L,2)+pow(x,2))/(2*L)))-(PI/4));
  Serial.println(teta);
  Serial.println(phi);
  float j=0;
  for (float i=0; i<phi; i+=0.25){
      Serial.println("OK");
      if(j<teta){
      Serial.println("DAC");
      epauleDroite(i);
      coudeDroite(j);
      j+=0.25;
      Serial.println(j);
    delay(1000);
  }
}
```

L'affichage du OK et du DAC me permet de vérifier que j'entre bien dans la boucle.

Une fois avoir tout réglé le programme s'exécute en entier.

Le seul problème est que, pour une distance test de 5, l'angle à appliquer à l'épaule est plus grand que l'angle max qu'il peut effectuer vers la droite.

Problème à régler la prochaine fois, à voir si on ne définit pas une taille max (ou standard) de trait à réaliser dans un premier temps.