Projet – initiation à matlab

Voici quelques petits projets qui vous permettront de vous familiariser avec des fonctions plus ou moins évoluées de matlab. Le but est qu'à la fin de ces séances, vous ayez un bref aperçu de ce qui peut être réalisé grâce au calcul matriciel.
Une petite introduction vous permet de comprendre ce qu'on attend de vous, puis vous êtes guidés pas à pas tout au long de la séance. Afin de développer votre autonomie, les fonctions utiles vous sont données (**en gras**), tout en vous laissant le soin d'aller chercher, soit dans le document (initiation à matlab), soit dans l'help du logiciel, comment elles s'utilisent.

1. création de fichiers de données (sinusoide, signal carré, etc…), les sauvegarder (manipulation de matrice)

2. ouvrir un fichier de données, les afficher (création de fonction, ranger les données dans une matrice etc…)

3. analyse des données : - filtre mesure du gain et du déphasage entre Ve et Vs (appel de fonction), sauvegarder les données dans un fichier, diagramme de Bode
- AO : mesure du gain et de epsilon

4. Présentations des fonctions

5. création d'un jeu de tir

Projet 1 : Fondamental et harmonique

Une des caractéristiques principales des sons musicaux est leur « hauteur », notion subjective à laquelle notre oreille est sensible et que l'expérience fait correspondre à leur fréquence, grandeur physique mesurable et susceptible d'être traitée par le calcul.
Si cette grandeur physique n'est connue que depuis moins de deux cents ans, cela n'a pas empêché les théoriciens de la musique, depuis l'Antiquité, de mettre en rapport les sons et les nombres, car ils avaient remarqué que la hauteur du son émis par une corde vibrante ou un tuyau sonore dépendait directement de leur longueur. On sait aujourd'hui démontrer que la fréquence des sons émis par ces corps est en proportion inverse de ces longueurs, et par conséquent, les mathématiciens du passé avaient pu raisonner de façon correcte sur l'acoustique malgré leur méconnaissance de la théorie des phénomènes vibratoires et des ondes stationnaires.
On a depuis longtemps reconnu le principe de l'équivalence des octaves, selon lequel deux sons dont les fréquences (f_1 et f_2) sont dans un rapport de 1 à 2 (f_1=k.f_2) « sonnent » de manière tellement comparable que l'on donne à de telles notes le même nom.
L'octave étant reconnue comme l'intervalle sonore le plus simple, il reste à la diviser en intervalles plus petits car elle ne permet pas à elle seule de composer ce qu'on peut appeler de la musique. Définir une gamme musicale, c'est donc définir une méthode pour diviser l'octave en intervalles sonores plus petits.
Bien que le spectre des fréquences sonores soit continu dans l'intervalle d'octave, on évitera d'utiliser des sons de fréquence totalement arbitraire, et ceci tant pour des raisons musicales que pour des raisons techniques liées aux instruments à sons fixes.

À hauteurs (donc fréquences) identiques, les sons émis par deux instruments différents (par exemple un violon et une flûte) ne résonnent pas de la même manière. Chacun se caractérise par ce qu'on appelle son "timbre" qui permet de l'identifier, traduction du fait qu'aucun son naturel n'est réellement simple : il résulte de la combinaison d'un son principal - ou fondamental (f_0) - qui fixe la fréquence perçue par l'oreille et d'un grand nombre de ses harmoniques (k.f_0) dont les pondérations relatives déterminent, précisément, son timbre.

Toute fonction mathématique périodique, et notamment celle qui correspond à une vibration sonore, peut être décomposée en une somme de fonctions sinusoïdales élémentaires dont les périodes plus courtes, (et donc les fréquences plus hautes) sont en rapport algébrique rationnel avec sa propre période. (décomposition en « série de Fourier »)

En acoustique, on va donc distinguer les sons simples correspondant à une fonction sinusoïdale simple et les sons musicaux, comprenant un son fondamental (f_0) et des harmoniques (f_k=k.f_0), dont les rapports de fréquence avec la fondamentale sont des quotients de nombres entiers.

But de la séance :

Tracer le signal en temporel d'un accord comprenant la fondamentale (ω_0), la première (ω_1) et la seconde (ω_2) harmonique.

1. **La fondamentale**

A partir de l'environnement matlab (dans la command Window), créez la matrice t variant de -2pi à 2 pi avec un pas de 0.1, puis la matrice sinus (ωt) (**sin**), avec ω=1.2.

t et s1 doivent apparaitre dans le workspace. Vous pouvez les appeler à tout moments
Sauvegarder s1 en format matlab (**save**), son nom doit normalement apparaître dans le curent directory.

Enfin tracer le graphique correspondant de cette sinusoïde, en fonction de t (**P=plot**).
Donner un titre (**title**), une légende (**legend**) et des noms aux axes x et y (**xlabel, ylabel**)
Sauvegarder la figure P en format jpg (**saveas**)

Vider l'ensemble des variables (**clear all**), vous remarquerez que les variables t,s_1, ou ω ne sont plus attribuées.

2. Les harmoniques (première et seconde)

Cette fois on cherche à construire le signal sinusoïdal correspondant à la pulsation fondamentale (ω_0=1.2), ainsi que ces deux premières harmoniques (ω_1=2ω_0 et ω_2=3ω_0).
Créez à nouveau la matrice t, variant de -2pi à 2 pi avec un pas de 0.1, puis les trois matrices de signal s1, s2 et s3 correspondantes aux trois sinusoïdales.

Sur une même fenêtre graphique, tracer séparément les trois sinusoïdes (**subplot**), sans oublier les titres, noms des axes et légendes. Tracer ensuite les trois sinusoïdes sur une deuxième fenêtre (**figure**), (**hold on/hold off**). Attribuez à chacune des courbes une couleur différente.

3. Les harmoniques (première et seconde), introduction des boucles

Soit N le nombre de sinusoïde que l'on souhaite tracer, de valeur N*ω_0, ω_0 la valeur de la pulsation fondamentale. On souhaite cette fois ci construire le fondamental ainsi que ses harmoniques grâce à une boucle (**for**), qui va de 1 à N.

Astuce : créer la matrice S=[s1 ; s2 …]

Enfin tracer la matrice S, en fonction de t (**P=plot**).
NB : faites attention à la manière dont vous remplissez cette matrice !

4. L'accord final : fondamental et harmonique

L'accord final est représenté par la superposition du fondamental et de ses harmoniques. On souhaite maintenant le calculer.

Astuce :
Additionner les colonnes de la matrice, mettre cette valeur dans un nouveau tableau (**sum**).

$$somme = \left(s_1 + s_2\right) = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ . \quad + \quad . \\ . \quad + \quad . \\ a_m + b_m \end{pmatrix}$$

Sur une même fenêtre graphique, tracer séparément les trois sinusoïdes, sans oublier les titres, noms des axes et légendes. Tracer ensuite la somme des trois sinusoïdes sur une deuxième fenêtre.

Projet 2 : Développement d'une routine matlab
- les séries de Fourrier
- Représentation des fonctions de transfert – Analyse d'un signal sinusoïdal

But de la séance :
Se familiariser avec le développement de routine matlab et les boucles.

1. Série de Fourier

Cette fois on veut réaliser une fonction (**function**) qui permet de fabriquer les séries de Fourier suivantes :
(1) la fréquence fondamentale et ses harmoniques impaires: `ci=(1/(2*i+1))*sin((2*i+1)*w*t)`

Vous réaliserez une routine matlab qui prend en entrée :
- w la valeur de la fréquence fondamentale
- le nombre d'harmonique choisis

Dans la routine, on définira une matrices C. En utilisant le principe des boucles, on pourra construire C. Cette fonction devra vous permettre de construire C pour un nombre d'harmoniques définit (comme fonction d'entrée). Vous tracerez ensuite C.

Qu'observez-vous ? et quand on augmente le nombre d'harmonique ?

2. Fonctions de transfert – analyse d'un signal sinusoïdal

La fonction de transfert en tension d'un quadripôle est définie par le nombre complexe **$\underline{T}(j\omega) = \underline{V}_S/\underline{V}_E$** où $\underline{V}_E$ et $\underline{V}_S$ sont respectivement les amplitudes complexes associées aux tensions sinusoïdales $v_E(t)$ et $v_S(t)$:

$$v_E(t) = V_E\sqrt{2}.\cos(\omega t + \phi_E) \quad \text{avec} \quad \underline{V}_E = V_E\sqrt{2}.e^{j\phi_E}$$
$$v_S(t) = V_S\sqrt{2}.\cos(\omega t + \phi_S) \quad \text{avec} \quad \underline{V}_S = V_S\sqrt{2}.e^{j\phi_S}$$

L'étude de l'évolution de cette transmittance en fonction de la fréquence $f = \omega/2\pi$ est indispensable pour caractériser le comportement fréquentiel de ce quadripôle.

On définit le gain G et l'argument ϕ par :

$$\boxed{\begin{array}{c} G_{(\text{en dB})} = 20.\log_{10}\left(\left|\underline{T}(j\omega)\right|\right) = 20.\log_{10}\left(\dfrac{V_S}{V_E}\right) \\ \phi_{(\text{en rad})} = \mathrm{Arg}\left(\left|\underline{T}\right|\right) = \phi_S - \phi_E = \text{ dephasage de } v_S(t) \text{ par rapport à } v_E(t) \end{array}}$$

Les diagrammes de Bode sont constitués de deux courbes :
- le gain G (en dB) en fonction de $\log_{10}(f)$
- l'argument ϕ (en rad) en fonction de $\log_{10}(f)$

On veut développer une routine matlab qui calcule la fonction de transfert d'un filtre passe haut universel.
La fonction de transfert d'un tel filtre est de la forme

$$\underline{T} = \frac{\underline{V}_S}{\underline{V}_E} = \frac{\left(j\dfrac{\omega}{\omega_0}\right)^2}{1 + j\xi\dfrac{\omega}{\omega_0} + \left(j\dfrac{\omega}{\omega_0}\right)^2}$$

Cette routine prend en entrée :

la valeur de ξ: le coefficient d'amortissement
la valeur de $\omega 0$: la fréquence de coupure

En sortie on souhaite tracer les diagrammes de Bode (gain et phase) en fonction de la ω en échelle logarithmique.
Pour cela vous créerez la créez la matrice ω variant de 1Hz à 10MHz avec un pas de 100.

Fabriquez ensuite la matrice $T(\omega)$ (attention, n'oubliez pas que ω est également une matrice). $T(\omega)$ et ω doivent apparaitre dans le workspace comme deux matrices colonnes. Vous pouvez les appeler à tout moment.
Sauvegarder $T(\omega)$ et ω en format matlab (**save**), son nom doit normalement apparaître dans le curent directory.
Enfin tracer le graphique correspondant de cette fonction de transfert, en fonction de ω (**P=plot**), ainsi qu'en échelle log (**semilogx/semilogy**), le tout sur la même figure (**subplot**).

Donner un titre (**title**), une légende (**legend**) et des noms aux axes x et y (**xlabel, ylabel**)
Sauvegarder la figure P en format fig (**saveas**)

Vous vous intéresserez aux différentes fonctions de transfert pour les valeurs de ξ suivantes :

<0.7, $=0.7$ ou >0.7

Projet 3 : REPRESENTATION DES FONCTIONS

Cette fois on veut réaliser une fonction (**function**) qui permet de regrouper tout ce qui a été fait précédemment. En sortie de ce programme doit être donné le gain et le déphasage des données étudiées.

Se programme doit demander quel est le fichier à charger (**uigetfile**).
Ce fichier sera chargé (**load**) dans une variable de votre choix. Vous calculerez ensuite le gain entre Vs et Ve, de même que leur déphasage.
Ensuite vous tracerez sur une même figure les deux signaux Vs et Ve, leur légende, titre etc … Et vous ferez apparaitre la valeur du gain et du déphasage. Enfin vous sauvegarderez la figure dans un fichier jpg et les valeurs du gain et du déphasage dans un fichier texte (**save**)

Projet 4: Régression linéaire – étude des caractéristiques d'un AO

L'ampli opérationnel (AOP) ou amplificateur linéaire intégré (A.L.I.) - Présentation, symbole et fonction de transfert.

L'amplificateur opérationnel le plus couramment utilisé est un amplificateur de différence à référence commune. Il possède:
- Deux entrées: Une entrée notée (-) dite entrée inverseuse et une entrée notée (+) dite entrée non inverseuse.
- Une sortie: Vs.

Les symboles sont représentés ci-dessous.

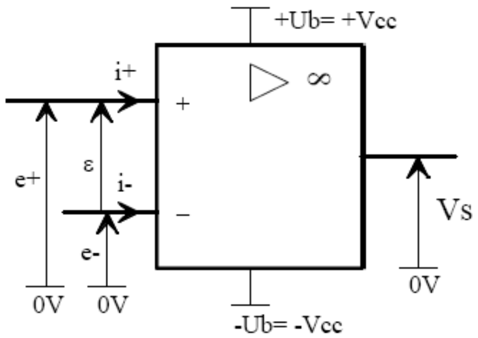


Figure 1 : symbole de l'amplificateur opérationnel (convention européenne – NF C03213)

Les potentiels des entrées sont repérées par rapport à la référence commune de toutes les tensions (masse ou 0V de référence), et sont nommés e+ et e-. La différence de potentiel entre l'entrée + et l'entrée - est appelée *tension d'entrée différentielle* et est notée e ou Ved (ou parfois e). e = (e+ -e-).
L'étage d'entrée différentiel de l'AOP permet d'assurer des courants d'entrée très faibles (i < 300nA pour un étage différentiel à transistor bipolaire, et i de qq pA à 10nA pour les étages différentiels à FET).
On supposera donc toujours: i+ = i- = 0 (sauf dans des cas très rares ou ces courants ne sont pas négligeables).
La sortie délivre la tension Vs par rapport à la référence (0V).
En statique: **Vs= Ad.(e+ - e-) + Amc.(e+ + e-)/2**
Ad représente l'amplification différentielle et Amc l'amplification de mode commun. Amc est très souvent négligeable par rapport à Ad, et Ad est noté A0. Ceci permet d'écrire: **Vs=A0.(e) = A0.(e+ - e-)**, avec A0typ > 100 000. A0 est appelé coefficient d'amplification (ou amplification statique) propre de l'ALI, ou amplification en boucle ouverte.
Attention *A0 est souvent appelé "Gain", alors que ce terme devrait être réservé à G0 = 20*Log (A0), exprimé en décibel (dB).*

Caractéristique de transfert statique:
L'amplification statique A0 étant très élevée, une très faible tension e suffit pour que Vs soit en saturation. Ex: Avec A0= 200 000, e est alors compris entre +/- (Vsat/A0) soit si Vsat est voisin de 15V, e = 75µV.

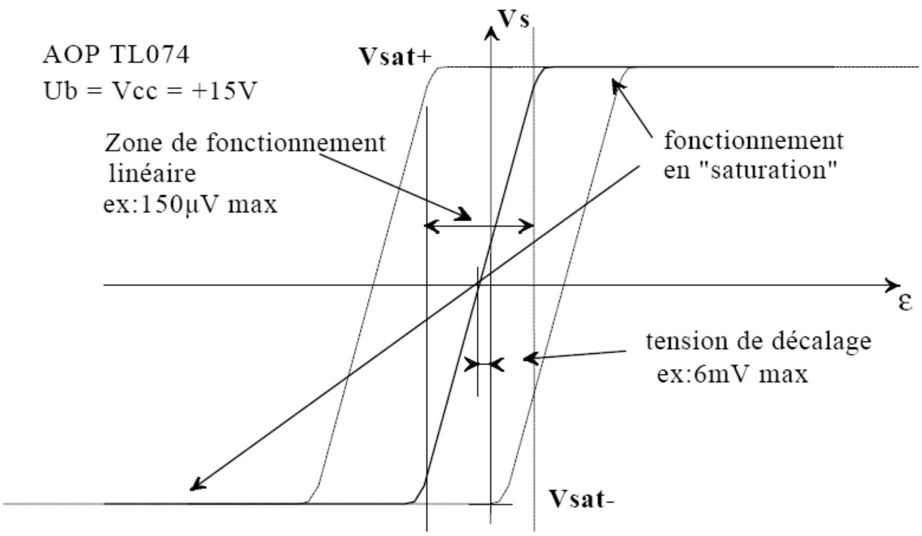


Figure 2 : caractéristique de l'amplificateur opérationnel (idéal ou avec une tension d'offset)

Pour le fonctionnement dans la zone linéaire (en dehors des plages dites de "saturation" = non saturée), on posera e = 0. Dans la réalité, il est parfois nécessaire de tenir compte de la tension de décalage ramenée à l'entrée (dite *tension d'offset Voff*), qui crée un décalage de la caractéristique de transfert de l'AOP vers la droite ou vers la gauche autour de l'origine. L'influence de cette tension sur le montage peut être étudiée plus facilement en plaçant une source de tension Voff sur l'entrée e+ ou e- de l'AOP et en considérant alors ce dernier comme idéal.

Régression linéaire - Droite des moindres carrés :

On cherche à exprimer la relation entre deux variables x (ici ε) et y (ici Vs) :

- x est la *variable indépendante* ou *explicative*. Les valeurs de x sont fixées par l'expérimentateur et sont supposées connues sans erreur (exemple : concentrations d'un produit à doser).
- y est la *variable dépendante* ou *expliquée* (exemple : réponse de l'analyseur). Les valeurs de y sont entachées d'une erreur de mesure. L'un des buts de la régression sera précisément d'estimer cette erreur.

On va chercher une relation de la forme : $y = b_0 + b_1x$. C'est l'équation d'une droite, d'où le terme de *régression linéaire* (Figure 1).

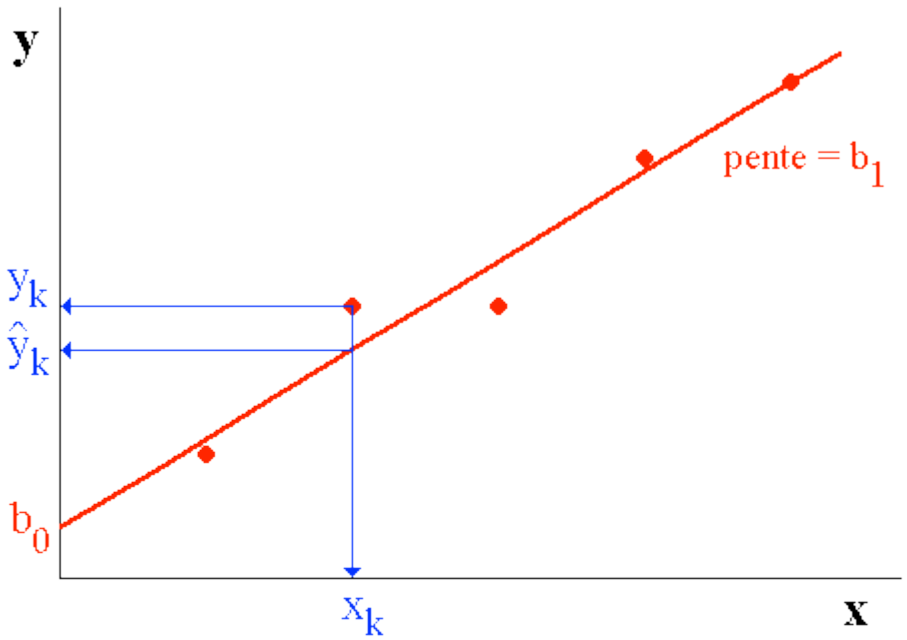


Figure 3 :principe de la régression linéaire

Du fait de l'erreur sur y, les points expérimentaux, de coordonnées (x_i, y_i), ne se situent pas exactement sur la droite. Il faut donc trouver l'équation de la droite qui passe le plus près possible de ces points.
On appelle estimateurs des moindres carrés β_1 et β_2 les valeurs minimisant la quantité

$$S(\beta_1, \beta_2) = \sum_{i=1}^{n} (y_i - \beta_1 - \beta_2 x_i)^2$$

Autrement dit, la droite des moindres carrés minimise la somme des carrés des distances verticales des points (xi, yi) du nuage de la droite ajustée y= β_1+ β_2 x .
La *méthode des moindres carrés* consiste à chercher les valeurs des paramètres β_1 et β_2 qui rendent minimale la *somme des carrés des écarts résiduelle* (*SS_r : sum of squared residuals*) entre les valeurs observées y_i et les valeurs calculées de y.
Les estimateurs des moindres carrés ont pour expressions :

$$\hat{\beta}_1 = \bar{y} - \hat{\beta}_2 \bar{x},$$

$$\hat{\beta}_2 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) y_i}{\sum_{i=1}^{n} (x_i - \bar{x})^2}. \qquad \textbf{Eq. 1}$$

n est le nombre de points de mesure.

Qualité de l'ajustement - Coefficients de corrélation

On examine ensuite le coefficient de corrélation r^2 :

$$r^2 = \frac{\left(\Sigma_{xy} - \frac{1}{n}\Sigma_x\Sigma_y\right)^2}{\left(\Sigma_{x^2} - \frac{1}{n}\left(\Sigma_x\right)^2\right)\left(\Sigma_{y^2} - \frac{1}{n}\left(\Sigma_y\right)^2\right)}$$

Eq. 1

Plus il est proche de 1, mieux les données sont décrites par une droite

r est le *coefficient de corrélation*. Il est affecté du signe + ou - selon que la pente de la droite (b_1) est positive ou négative. r est toujours compris entre -1 et 1.

Remarques :

1. Si la droite passe exactement par les points, $r^2 = 1$ et $r = \pm 1$
2. Si les variables x et y sont indépendantes, r = 0. Toutefois la réciproque n'est pas vraie : r peut être égal à 0 si la relation est non linéaire.
3. Il vaut mieux utiliser r^2 que r. En effet $r < 1$ donc $r^2 < r$. Par exemple une valeur de r égale à 0.70 peut sembler correcte, alors qu'on n'a expliqué que 50% des variations de y ($r^2 = 0.49$).

But de la séance : On cherche à caractériser un amplificateur (gain, offset) opérationnel à partir des valeurs suivantes obtenues après expérimentations. Durant l'expérience, on a bien fait attention de rester dans la zone de fonctionnement linéaire de l'AOP.

ε (µV) = (-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5)
Vs (V) = (-8.9470, -6.8380, -5.0269, -2.8264, -0.1178, 1.1823, 3.7553, 5.5080, 7.1319, 9.2801, 10.0172)

Les caractéristiques de l'AO seront déduites des données expérimentales à partir d'une régression linéaire, dont le principe est expliqué ci-dessus.
Vous rédigerez un programme matlab qui vous permettra de

- charger les fichiers de données expérimentales
- de calculer la droite s'ajustant au mieux aux données expérimentales.

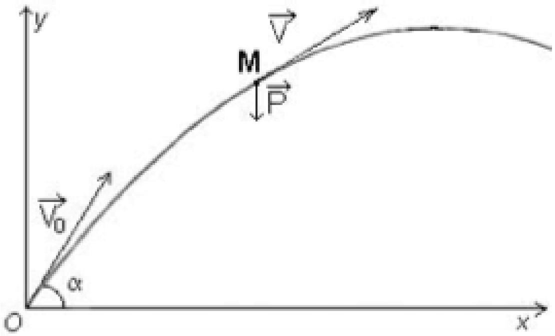
Les données expérimentales et la droite seront ensuite tracées sur une même figure. Vous y ferez également apparaître la valeur du coefficient de corrélation déterminant la qualité de votre ajustement. Enfin vous sauvegarderez les valeurs de b_0, b_1 et r_s dans un fichier txt (**fprintf**).

Principe de fprintf

```
x = 0:.1:1;
y = [x; exp(x)];
fid = fopen('exp.txt', 'wt');
fprintf(fid,'x|exp|\n');
fprintf(fid, '%6.2f |%12.8f|\n', y);
fclose(fid)
```

Projet 5: tir de missile

Les calculs « classiques » depuis Torricelli, calculs « dans le vide », sans frottement.



A t=0 le projectile est lancé à la vitesse V_0 selon un angle α (en degrés) avec l'horizontale. On considère que seul le poids s'applique à la masse M du projectile.

Dans un repère orthogonal, la décomposition sur les axes [ox) et [oy) permet d'écrire :

$$\begin{cases} v_{0x} = v_0 \cdot \cos(\alpha) \\ v_{0y} = v_0 \cdot \sin(\alpha) \end{cases} \quad \text{où } v_0 = \|\overrightarrow{V_0}\|.$$

En un point M quelconque de la trajectoire nous avons :

Horizontalement :

1. **À la main :**

$F_x \cdot \vec{i} = m \cdot a_x \cdot \vec{i} = m \cdot \frac{d^2x}{dt^2} \cdot \vec{i} = m \cdot \frac{dv_x}{dt} \cdot \vec{i} = \vec{0}$ avec $m \neq 0$ où a_x représente la valeur absolue de l'accélération horizontale.

Ce qui permet d'écrire $\frac{dv_x}{dt} = 0$. Par intégration directe : $v_x(t)=K_{1x}$ où K_{1x} est une constante.

Détermination de la constante : $v_{0x}=v_x(0)=v_0 \cdot \cos(\alpha)$. Donc $v_x(t)=v_0 \cdot \cos(\alpha)$.

Alors $\frac{dx}{dt} = v_0 \cdot \cos(\alpha)$ qui par intégration donne $x(t) = v_0 \cdot \cos(\alpha) \cdot t + K_{2x}$. Les conditions initiales permettent d'écrire $\boxed{x(t) = v_0 \cdot \cos(\alpha) \cdot t}$ (1).

Verticalement :

1. **À la main :**

$F_y \cdot \vec{j} = m \cdot a_y \cdot \vec{j} = m \cdot \frac{d^2y}{dt^2} \cdot \vec{j} = m \cdot \frac{dv_y}{dt} \cdot \vec{j} = -m \cdot g \cdot \vec{j}$ avec $m \neq 0$, où a_y représente la valeur absolue de l'accélération verticale et g=9,81 $m.s^{-2}$ une approximation de l'accélération de la pesanteur terrestre.

Ce qui permet d'écrire $\frac{dv_y}{dt} = -g$. Par intégration directe : $v_y(t)=-g \cdot t+K_{1y}$ où K_{1y} est une constante.

Détermination de la constante : $v_{0y} = v_y(0) = v_0 \cdot \sin(\alpha)$. Donc $v_y(t) = -g \cdot t + v_0 \cdot \sin(\alpha)$.

Alors $\frac{dy}{dt} = -g \cdot t + v_0 \cdot \sin(\alpha)$ qui par intégration donne $y = \left(-\frac{1}{2} g \cdot t^2 + v_0 \cdot \sin(\alpha) \cdot t\right) + K_{2y}$.

Les conditions initiales permettent d'écrire $\boxed{y(t) = -\frac{1}{2} g \cdot t^2 + v_0 \cdot \sin(\alpha) \cdot t}$ (2).

En éliminant t entre les expressions (1) et (2), on trouve $\boxed{y(x) = -\frac{1}{2} \cdot g \cdot \frac{x^2}{v_0^2 \cdot \cos^2(\alpha)} + x \cdot \tan(\alpha)}$ (3).

Avec g=9,81 (m/s^2), α=45 (en degrés) et v_0=220 (m/s).
Dans ce cas, on remarque que le résultat est indépendant de la masse, de la taille (surface frontale) du projectile.

But :
Vous réaliserez un programme qui vous permet d'essayer de tirer sur une cible fixe, et de savoir si vous l'avez atteinte ou pas.

Vous définirez la position d'une cible, à la même hauteur que celle du canon, et à une distance d choisie aléatoirement.
Vous afficherez sur un graphe, la position du canon en rouge (en 0,0) et la position de la cible en vert (en 0,d).

En fonction de la position de la cible, donnez les données angle (en radian), vitesse (en $m.s^{-1}$). Ces entrées vous seront demandées au fur et à mesure (**dialbox**).

On considère que la cible est atteinte si le boulet de canon arrive à une distance de moins d'un mètre de la cible.
Affichez sur le même graphe la trajectoire du boulet de canon.

En sortie, le programme vous dit si vous avez gagné, ou non. Si non il vous propose de rejouer en gardant la position de la cible fixe, et en changeant les paramètres angle et vitesse.