

Question 1**(6 points)****1.1)** Répondre par vrai ou faux:

- Soit A une matrice de taille M lignes et N colonnes. L'appel de la fonction `plot(A, 'linewidth', 2);` permet de tracer des courbes où les valeurs de l'axe Y sont les colonnes de la matrice A.
- Le caractère `\n` permet un retour à la ligne pour les annotations des graphiques.
- La fonction `fseek()` de MATLAB permet de se déplacer à la fin d'un fichier texte.
- La représentation en format IEEE754 simple précision du nombre réel 102 est :

| Signe | Exposant | Mantisse |
|-------|----------|------------------------------|
| 0 | 10000101 | 1001 1001 0000 0000 0000 000 |

- $(11110001110011010)_2 = (161632)_8$.
- Pour une fonction, `varargin` et `varargout` sont des matrices permettant de définir un nombre indéfini de paramètres.
- `nargin()` et `length(varargin)` peuvent donner le même résultat lors de leur utilisation dans une fonction MATLAB.
- Les fonctions `clock()` et `etime()` ne retournent que des valeurs décimales exprimées en secondes.
- L'addition de deux nombres entiers représentés en complément à deux, sur 8 bits, donne toujours le même résultat que l'addition de ces nombres de type `int8`.
- Pour modifier un fichier binaire, il est possible de l'ouvrir en lecture (avec le mode `'r'`) puis l'ouvrir en écriture (avec le mode `'w'`) sans qu'il soit nécessaire de le fermer entre ces deux opérations d'ouverture.

Solution (5 points, 0.5 points/sous question) V, F, F, F, V, F, V, F, V, F**1.2)** Soient les nombres 50 et 108 :

- Donner la représentation en binaire de ces nombres de type `int8`.
- Donner le résultat de l'addition en binaire.
- Y a-t-il une retenue et/ou débordement? Dire si la réponse est correcte. Justifiez.
- Faire la conversion dans la base 10 du résultat de l'addition en binaire.

Solution (1 point – 0.25/ étape)

$$50 = 0011\ 0010$$

$$108 = 0110\ 1100$$

résultat = 1001 1110 sans retenue, oui débordement (l'addition de deux positifs donne un négatif), réponse incorrecte

$$128+16+8+4+2 = -98$$

Question 2**(12 points)**

Vous avez commencé à faire des plans de vacances depuis plusieurs semaines, pour votre meilleur ami et pour vous-même. Étant donné que le prix est un facteur important dans votre décision, vous vous êtes abonné au site `Destination Soleil` qui vous informe des offres de vacances durant la période des Fêtes. La compagnie vous envoie périodiquement par courriel le fichier texte `Soleil.txt`.

Chaque ligne du fichier texte contient :

- un code (chaîne de 5 caractères formée par des lettres et des chiffres, unique pour une destination et une offre);
- une destination (hôtel, ville, pays), toujours 3 mots : un mot pour le nom de l'hôtel, un mot pour le nom de la ville, un mot pour le nom du pays;
- la catégorie (un nombre réel représentant le nombre d'étoiles entre 0 et 5) de l'hôtel;
- le prix en dollars canadiens (un nombre réel);
- le nombre de nuits couvertes par l'offre (un entier);
- l'appréciation de vacanciers (chaîne de caractères).

Exemple de fichier `Soleil.txt`

```
125ME El_Cano Acapulco Mexique 4 1562.00 7 1/3
126ME Oasis Cancun Mexique 4 475.00 7 1/1
128ME Bellevue Cancun Mexique 3 618.00 5 3.37/19
. . .
123JA Four_Seasons Kingston Jamaïque 3 1309.00 7 0/0
124JA Knutsford_Court Kingston Jamaïque 3.5 1930.00 7 0/0
125JA Knutsford_Court Kingston Jamaïque 3.5 1829.00 6 0/0
. . .
127JA Ocho_Rios Ocho_Rios Jamaïque 3 673.00 5 3.1/21
123CU Las_Terrazas Havana Cuba 2 288.00 5 2.65/54
. . .
```

Vous avez créé sur votre ordinateur le fichier texte `Vacances.txt` qui contient seulement les destinations qui vous intéressent. Chaque ligne contient :

- la position (c.-à-d. la ligne) de la destination du fichier `Soleil.txt`
- un vecteur ayant 1 ligne et 12 colonnes contenant l'évolution du prix des 12 (douze) dernières offres reçues du site `Destination Soleil`.

Exemple de fichier `Vacances.txt`

```
6 674.00 675.00 650.00 . . . 630.00 650.00 650.00 650.00
12 725.00 693.00 693.00 . . . 690.00 690.00 679.00 673.00
13 325.00 326.00 300.00 . . . 290.00 290.00 290.00 290.00
```

Notes :

- Le nombre de destinations reçues par courriel est différent du nombre de destinations écrites dans le fichier `Vacances.txt`.
- Les fichiers texte `Soleil.txt` et `Vacances.txt` sont sauvegardés toujours avec les mêmes noms dans le même répertoire, vous n'avez pas besoin de les demander à l'utilisateur.
- On suppose que toutes les offres de vacances transmises périodiquement par la compagnie sont toujours à la même ligne, c.-à-d. la destination ayant le code 126ME (pour Oasis Cancun Mexique ...) se trouve toujours à la ligne numéro 2.
- Pour toutes les ouvertures de fichiers échouées, affichez un message d'erreur.
- Pour la question 2, toutes les sous-questions il est interdit d'utiliser les fonctions `textscan()`, `strsplit()`, `feof()`.

Question 2.1

Vous pensez écrire un code MATLAB qui vous permettra de chercher rapidement les destinations qui vous intéressent dans le fichier reçu par courriel. Votre programme doit suivre les étapes suivantes:

- 2.1.1** Il lit le fichier reçu par courriel, `Soleil.txt` et stocke toutes ces informations dans un ensemble de cellules `Sol`. Chaque ligne de l'ensemble de cellules `Sol` contient les 8 éléments spécifiques à une destination, un élément par cellule. Donc l'ensemble de cellules `Sol` a 8 colonnes et plusieurs lignes.

Solution:

Pour les deux questions 2.1.1. et 2.1.2

0.25 ouvertures (0 si manque une ou les deux) ,

0.50 validations et message (0,25 si manque 2, 0 si plus de deux),

0.25 fermetures (0 si manque une ou les deux)

0.25 lectures préliminaires, (0 si manque une ou les deux) ,

0.25 while et condition correcte (sans feof - interdite), (0 si manque une ou les deux) ,

0.25 lecture suivante (0 si manque une ou les deux) ,

```
fil=fopen('Soleil.txt', 'rt')
```

```
if fil ~=-1
```

```
    i=1;
```

```
        code1=fscanf(fil, '%s', 1);
```

```
        while ~isempty(code1)
```

```
            sol{i,1}=code1; %code (0.75 sauvegarde correcte dans l'ensemble  
de cellules avec les bons types de données)
```

```
            sol{i,2}=fscanf(fil, '%s', 1); %hotel
```

```
            sol{i,3}=fscanf(fil, '%s', 1); %ville
```

```
            sol{i,4}=fscanf(fil, '%s', 1); %pays
```

```

        sol{i,5}=fscanf(fil, '%f', 1); %categ
        sol{i,6}=fscanf(fil, '%f', 1); %prix
        sol{i,7}=fscanf(fil, '%i', 1); %nuits
        sol{i,8}=fscanf(fil, '%s', 1); %revue
        i=i+1;
        code1=fscanf(fil, '%s', 1);
    end
    fclose(fil)
else
    disp('ouverture echouee')
end

```

2.1.2 Il lit le fichier `Vacances.txt` déjà créé sur votre ordinateur et stocke les informations du fichier dans un ensemble de cellules `Prix`. Chaque ligne de l'ensemble de cellules `Prix` contient les 2 éléments spécifiques à une destination préférée, un élément par cellule :

- la première cellule contient la position dans le fichier texte `Soleil.txt`;
- la deuxième cellule contient le vecteur des 12 derniers prix d'une destination préférée.

Donc l'ensemble de cellules `Prix` a 2 colonnes et plusieurs lignes, une ligne par destination préférée.

Solution:

```

filv=fopen('Vacances.txt', 'rt')
if filv ~=-1

    i=1;
    pos=fscanf(filv, '%i', 1)
    while ~isempty(pos)
        prix{i,1}=pos
        prix{i,2}=fscanf(filv, '%f', [1 12])%0.50 lecture correcte
        matrice de prix type et taille
        i=i+1;
        pos=fscanf(filv, '%i', 1);
    end

    fclose(filv)
else
    disp('ouverture echouee')
end

```

2.1.3 Il écrit dans le fichier `Vacances.txt`, pour chaque destination, le nouveau prix (l'ajoute à la fin du vecteur existant du fichier `Vacances.txt`) reçu avec l'offre `Soleil.txt`. La taille du vecteur ne doit pas changer à l'ajout d'un nouveau prix, le prix le plus ancien est effacé pour que la taille du vecteur de prix ne change pas.

La modification du vecteur doit être faite dans l'ensemble de cellules `Prix` avant que le vecteur soit enregistré dans le fichier texte.

Il affiche à l'écran l'hôtel, la ville, le pays et le dernier prix pour les destinations où les prix ont diminué de 5% (ou plus). Si aucune destination n'a diminué le prix de 5% ou moins, le programme affiche : Aucune destination.

Exemple de fichier `Vacances.txt` **après la modification :**

```
6   675.00 650.00 . . . 630.00 650.00 650.00 650.00 605.00
12  693.00 693.00 . . . 690.00 690.00 679.00 673.00 673.00
13  326.00 300.00 . . . 290.00 290.00 290.00 290.00 288.00
```

Exemple d'affichage à l'écran

Bellevue Cancun Mexique \$605.00

Solution (2.5 points):

```
filve=fopen('Vacances.txt', 'wt')%0.25 ouverture en écriture avec wt
if filve ~= -1
    k=0;
    for j=1:size(prix,1)%0.25 repetition parcours prix
        prix{j,2}=[prix{j,2}(2:end), sol{prix{j,1},6}]; %0.25
        fprintf(filve, '%i\t', prix{j,1});%0.25 ecriture position

        for ii=1:12 %0.25 ecriture prix
            fprintf(filve, '%.2f\t', prix{j,2}(ii));
        end
        fprintf(filve, '\n') %0.25 ecriture prix

        %0.5 trouver et afficher destinations
        if (prix{j,2}(end-1) - prix{j,2}(end))>=.05*prix{j,2}(end-1);
            fprintf('%s\t%s\t%s\t%.2f\n', sol{prix{j,1},2},
                sol{prix{j,1},3}, sol{prix{j,1},4}, prix{j,2}(end));
            k=k+1;
        end
    end
    %0.5 afficher aucune destination
    if k==0
        disp('Aucune destination')
    end
    fclose(filve)
else
    disp('error')
end
```

Question 2.2

Vous désirez envoyer les détails des destinations qui vous intéressent à votre meilleur ami. Pour cela, vous combinez les informations des ensembles de cellules `Sol` et `Prix` dans un fichier binaire encodé avec une clé de cryptage.

Méthode d'encryptage

- Générer aléatoirement une clé de cryptage: une valeur entière entre 1 et 4;
- Additionner la clé de cryptage à chaque information lue (code, catégorie etc.).

Attention : pour les chaînes de caractères, cette opération transforme les caractères selon leur équivalent en code ASCII avant d'effectuer l'addition, comme l'illustre l'exemple suivant :

Soit la variable $A = \text{'a l l o'}$. Si la clé de cryptage est 1, alors la variable encryptée B est $A + 1$. On obtient :

$$B = 98 \quad 109 \quad 109 \quad 112$$

Suite au cryptage, il est important d'enregistrer l'information en caractère et non en code ASCII équivalent, c'est-à-dire $B = \text{'b m m p'}$ et non $B = 98 \quad 109 \quad 109 \quad 112$.

2.2.1 Écrire la fonction `CrypterInfos()` (**prototype et code MATLAB**) qui :

- prend en entrée les ensembles de cellules `Sol` et `Prix`,
- encode toutes les informations pour chaque destination d'intérêt, selon la méthode d'encryptage,
- écrit l'information codée dans le fichier binaire `Vacances.bin` (vous pouvez directement utiliser le nom de ce fichier dans vos fonctions, il n'est pas nécessaire de le demander) qui contient, pour chaque destination d'intérêt, les données suivantes :
 - ✓ le code de la destination (**chaîne de caractères**);
 - ✓ la catégorie de l'hôtel (**single ou float32**);
 - ✓ le nombre de nuits associé à l'offre (**uint8**);
 - ✓ les prix contenus dans le vecteur de prix (**double ou float64**).
- la fonction retourne :
 - ✓ la clé de cryptage `Cle`;
 - ✓ le nombre de destinations d'intérêt `nbDest`;
 - ✓ ou 0 pour les deux paramètres de sortie si le fichier `Vacances.bin` n'a pas pu être créé.

Vérifier que :

- la fonction `CrypterInfos()` est appelée avec le bon nombre d'arguments;
- les arguments d'entrée sont valides (types, dimension).

Utiliser la fonction `error()` pour signaler tout problème avec les arguments d'entrée ou de sortie.

Les fonctions `char()` et `iscell()` peuvent vous être utiles.

Solution

Pour les questions 2.2.1. et 2.2.3

0.25 ouvertures (0 si manque une ou les deux) ,

0.25 validations et message (0,25 si manque 2, 0 si plus de deux),

0.25 fermetures (0 si manque une ou les deux)

```
function [Cle, nbDest] = CrypterInfos(Sol, Prix)%0.25 prototype

Cle = 0; nbDest = 0; %0.25 initialisation valeurs de retour
%0.75 verification arguments d'entree, valeurs de retour, message
d'erreur
if nargin ~= 2 || nargin ~= 2 || ~iscell(Prix) || ...
    size(Prix,2)~= 2 || ~iscell(Sol) || size(Sol,2) ~= 8
    error('Erreur');
end

fid = fopen('Vacances.bin','w');

if fid ~= -1
    Cle = round(1 + 3*rand()); %0.25 generation cle
    nbDest = size(Prix,1);
    for i = 1:nbDest %0.25 repetition parcours ensemble prix
        % Lecture des informations dans les cellule Sol et Prix
        pos = Prix{i,1};
        code = Sol{pos,1};
        cat = Sol{pos,5};
        nuit = Sol{pos,7};
        prix = Prix{i,2};
        % Ecriture dans le fichier Vacances.bin %0.5 ecriture correcte
        (les bons types de donnees)
        fwrite(fid, char(code + Cle),'char');
        fwrite(fid, cat + Cle, 'single');
        fwrite(fid, nuit + Cle, 'uint8');
        fwrite(fid, prix + Cle, 'double');
    end
    fclose(fid);
else
    disp('Erreur d'ouverture du fichier');
end
```

Question 2.3

Votre meilleur ami reçoit le fichier `Vacances.bin` et la clé de sécurité que vous lui avez envoyés. Pour mieux faire son choix, il décide d'analyser les statistiques des 3 destinations offrant les meilleurs rapports prix/nuit pour une catégorie donnée.

2.3.1 Écrire **seulement le prototype** de la fonction `Trier()`. Cette fonction reçoit en entrée un ensemble de cellules `Cell` de taille `[N X 3]`, où chaque ligne de la

cellule correspond aux informations relatives à une destination donnée comme suit :

Exemple de cellules Cell

| Code | Prix/nuit | Vecteur de prix |
|-------|-----------|--|
| 125ME | 223.14 | 675.00 650.00 . . . 630.00 650.00 650.00 650.00 1562.00 |
| 123JA | 187 | 693.00 693.00 . . . 690.00 690.00 679.00 673.00 1309.00 |
| 125JA | 304.83 | 326.00 300.00 . . . 290.00 290.00 290.00 290.00 1829.00 |
| 123CU | 57.6 | 329.00 300.00 . . . 290.00 290.00 290.00 291.00 288.00 |

Cette fonction trie les destinations par ordre croissant selon le rapport prix/nuit et retourne un ensemble de cellules CellTrie de taille [N X 3] contenant les destinations triées.

Exemple de cellules CellTrie

| Code | Prix/nuit | Vecteur de prix |
|-------|-----------|--|
| 123CU | 57.6 | 329.00 300.00 . . . 290.00 290.00 290.00 291.00 288.00 |
| 123JA | 187 | 693.00 693.00 . . . 690.00 690.00 679.00 673.00 1309.00 |
| 125ME | 223.14 | 675.00 650.00 . . . 630.00 650.00 650.00 650.00 1562.00 |
| 125JA | 304.83 | 326.00 300.00 . . . 290.00 290.00 290.00 290.00 1829.00 |

Solution (0.25 points) fonction [CellTrie] = Trier(Cell)

2.3.2 Écrire la fonction StatVacances () qui :

- reçoit en entrée :
 - ✓ la clé de cryptage, Cle;
 - ✓ le nombre de destination d'intérêt nbDest;
 - ✓ la catégorie **minimale** d'hôtel désirée, Cat.
- retourne un enregistrement Stats contenant les informations pour les 3 meilleures destinations trouvées ou -1 si le fichier binaire n'a pas pu être ouvert (on suppose qu'il existe toujours au moins 3 destinations offrant un hôtel de catégorie Cat étoiles ou plus). Les informations dans l'enregistrement Stats doivent être regroupées selon les champs suivants :
 - ✓ Code: le code correspondant à une destination donnée;

- ✓ **Prix**: un vecteur-ligne contenant l'évolution du prix pour la destination correspondante.

Pour réaliser cette tâche, vous ne devez lire et ne décoder **que les informations utiles**, (en vous déplaçant d'une position à l'autre dans le fichier) en procédant comme suit :

- pour chaque destination contenue dans le fichier `Vacances.bin`, vous allez d'abord lire et décoder la catégorie de l'hôtel;
- si la catégorie de l'hôtel est plus petite que la catégorie minimale `Cat`, vous ignorez le reste des informations relatives à cette destination;
- si la catégorie est plus grande ou égale à `Cat`, vous lisez et décidez, le code de destination, le nombre de nuits couverts par l'offre, et le vecteur de prix. Par la suite, vous ajoutez ces informations dans un ensemble de cellules, `Cell`, selon le format spécifié pour la fonction `Trier()`.

Une fois que le fichier `Vacances.bin` a été entièrement lu et décodé, votre fonction fait appel à la fonction `Trier()`. Les 3 meilleures destinations trouvées par la fonction `Trier()` sont alors retournées par la fonction `StatVacances()`, via l'enregistrement `Stats`.

Note: Il n'est pas nécessaire de vérifier les entrées et sorties de la fonction `StatVacances()`.

Solution

```
function[Stats] = StatVacances(Cle, nbDest, Cat)%0.25 prototype

fid = fopen('Vacances.bin','r');
if fid ~= -1
    Cell = {};
    for i = 1:nbDest
        if i == 1 % 0.25 lere destination
            fseek(fid,5,'bof'); %0.25 fseek correct
        end
        categ = fread(fid,1,'single') - Cle;
        if categ >= Cat %0.25 condition
            % 0.75 Lecture des informations
            fseek(fid,-9,'cof');
            code = char(fread(fid,[1 5],'char') - Cle);
            fseek(fid,4,'cof');
            nuit = fread(fid,1,'uint8')- Cle;
            prix = fread(fid,[1 12], 'double') - Cle;

            % 0.50 Ecriture des informations dans la cellule Cell
            Cell{end+1,1} = code;
            Cell{end,2} = prix(12)/nuit;
            Cell{end,3} = prix;

            fseek(fid,5,'cof');
        end
    end
end
```

```
        if i ~= nbDest
            fseek(fid,102,'cof');
        end
    end
end
fclose(fid);

CellTrie = Trier(Cell);%0.25 appel fonction Trier

for i = 1:3 %0.5 sauvegarde dans l'enregistrement stats
    Stats(i).Code = CellTrie{i,1};
    Stats(i).Prix = CellTrie{i,3};
end
else
    disp('Erreur d'ouverture du fichier');
    Stats = -1; %0.25 valeur de retour si ouverture incorrecte
end
```

Question 3

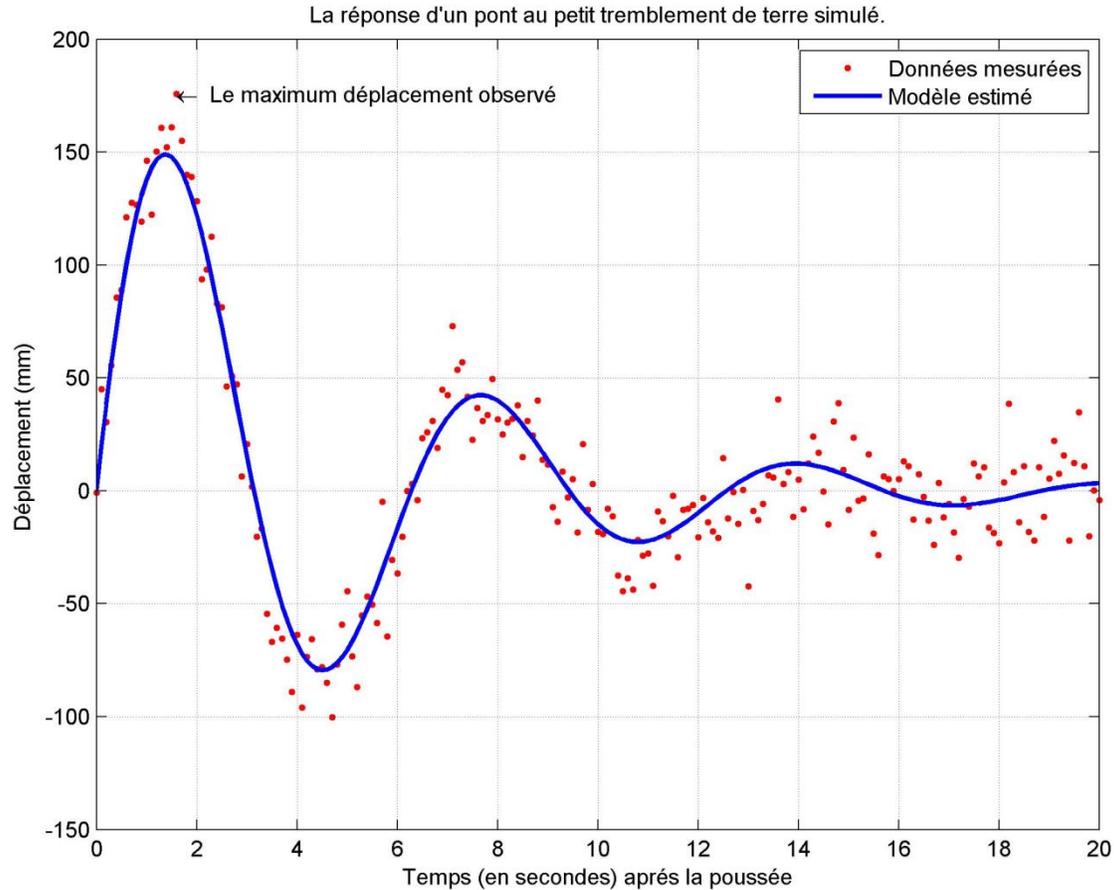
(2 points)

Soient :

- le vecteur $\mathbf{x}=0:0.1:20$;
- le vecteur \mathbf{d} ayant la même taille que \mathbf{x} et contenant des données mesurées,
- le vecteur \mathbf{y} contenant les valeurs d'un modèle estimé en fonction de \mathbf{x} ,

Les vecteurs \mathbf{x} , \mathbf{d} , \mathbf{y} sont donnés et ont la même longueur. Écrire le programme qui trace le graphique ci-dessous (voir la figure de la page suivante). Le programme doit :

1. tracer sur le même graphique \mathbf{d} et \mathbf{y} en fonction de \mathbf{x} . \mathbf{d} a la **couleur rouge** et \mathbf{y} a la **couleur bleue**. Pour tracer \mathbf{d} vous pouvez utiliser le marqueur point.
2. ajouter une **grille** au graphique;
3. ajouter le **titre** du graphique, **les titres des axes** et **la légende**; la légende doit être placée dans le coin haut à droite;
4. ajouter **un texte qui montre** le maximum des données mesurées ;
5. exporter le graphique dans le fichier eps graphique; la résolution est 300dpi.



Solution (2 points)

```
% Les choses 'connues' déjà
% x=0:0.1:20;
% d=y+15*randn(size(y));
% y=200*exp(-0.2*x).*sin(x);

figure; plot(x,d,'r.','markersize',10); %aussi possible un seul
plot 0.25 plot
hold on; %0.25 hold on / hold off
plot(x,y,'b','linewidth',2);
xlabel('Temps (en secondes) après la poussée'); %0.5 axes, titre
et legende
ylabel('Déplacement (mm)');
title('La réponse d\'un pont au petit tremblement de terre
simulé.');
```

```
grid on %0.25 grid on
legend('Données mesurées', 'Modèle estimé')
[d_max, pos_max]= max(d); %0.5 point et texte
text(x(pos_max),d_max,'\leftarrow Le max déplacement observé');
hold off ;
print -depsc -r300 examen_final %0.25 exportation graphique
```

