

# **Projet 2025 addendum**

17.12.2025 - rev.2

# modifications

- 1.7.12.2025, initial version
- 19.12.2025, ajouter Slide 13 notes, bad sequences et exécutable

# Rendu sur Moodle

- Déposer l'archive .zip (avec vos noms) via la page moodle du cours
- Nombre illimité de soumissions, uniquement la dernière est retenue


## Soumission projet 2025

Soumission du projet 2025 sous la forme d'une archive .zip contenant l'entier de vos fichiers.  
La taille maximum est de 50 MB.  
Nommez votre archive avec les noms des membres du groupe.

**File submissions**

Maximum size for new files: 50MB, maximum attachments: 1

Files



You can drag and drop files here to add them.

Save changes Cancel

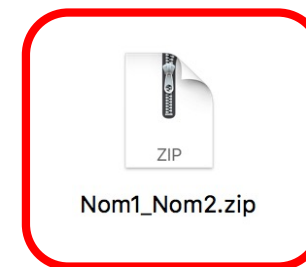
# Contenu de l'archive .zip

**Rendu sur Moodle de vos fichiers dans une archive .zip aux noms des membres du groupe, avec:**

- Brève documentation au format pdf
- Le(s) fichier(s) source(s) c et fichier(s) projet relatif
- Le programme/exécutable (Pix2Pos) compilé pour mac ou windows
- Le(s) VI(s) LabVIEW
- Le(s) script(s) Matlab
- Un jeu de fichiers générés par vos différents programmes: *pixmap.bin*, *pos.txt* et les fichiers *AnalyseTx.m*, *ScoreSheetTx.pdf* pour toutes les séquences fournies
- Ne **PAS** mettre les séquences dans l'archive

**Contrôlez que **\*\*TOUS\*\*** les fichiers se trouvent dans l'archive avant de la soumettre. Je préfère 100 fichiers de plus que 1 de pas assez!**


**Soumettre l'archive et vérifier ce que vous avez uploadé en le téléchargeant sur une autre machine/autre dossier!**



**Fichier  
à soumettre**

# Procédure d'évaluation (1)

*(A tester dans votre environnement avant la soumission!)*

- Décompression de l'archive .zip
- Ouverture du fichier LabVIEW
- Choix de paramètres dans l'interface LabVIEW
- Exécution du VI en cliquant sur la flèche 

*Après exécution de la partie (C et LabVIEW) de manière correcte, est-ce que Matlab génère correctement le fichier pdf score sheet avec les informations demandées?*

**oui** => vous êtes assuré/e d'avoir la moyenne pour la partie projet  
*(si vous n'avez pas triché avec votre code!)*

**non** => analyse de votre code et des documents fournis

# Procédure d'évaluation (2)

## C

- Votre code C est analysé en premier via le VI LabVIEW fourni et ensuite de manière indépendante en entrant automatiquement des valeurs pour pixmap.bin et en observant le fichier généré *pos.txt* ainsi que les messages éventuels dans *stderr* et *stdout*
- **Le comportement de votre programme C est comparé avec les spécifications fournies durant le cours ainsi qu'avec votre documentation!**
- **Attention aux chemins relatifs, noms de fichiers, etc.**
- Analyse du(des) fichier(s) source C

## LabVIEW

- Test du bon déroulement du programme LabVIEW, test avec plusieurs séquences et paramètres (valides et invalide), affichage du résultat dans matlab, **gestion des erreurs**, etc.
- Analyse de la gestion des erreurs en cas de problème avec la partie C ou LabVIEW (message(s) d'erreur et/ou autres mécanismes)
- Test avec exécutable C manquant
- **Comment gérez vous le cas d'une image invalide (ex. fichier Word renommé avec .PNG) ?**
- Analyse du(des) fichier(s) source LabVIEW
- **Toutes les séquences sont analysées automatiquement, pour cela votre VI principal doit avoir **strictement** le même *connector pane* que celui fourni.**

## Matlab

- Générations correctes des fichiers
- Analyse du(des) fichier(s) source matlab
- Utilisation minimale (~ absence) de boucles

# Procédure d'évaluation (3)

- Les tests se feront sur la machines virtuelles Windows ou sur mac
- Les spécifications et commentaires peuvent être en français ou en anglais
- Mettez des commentaires dans le code C, LabVIEW et matlab
- Relisez les spécifications, est-ce que vos programmes les suivent, ex. noms de fichiers, format, etc.

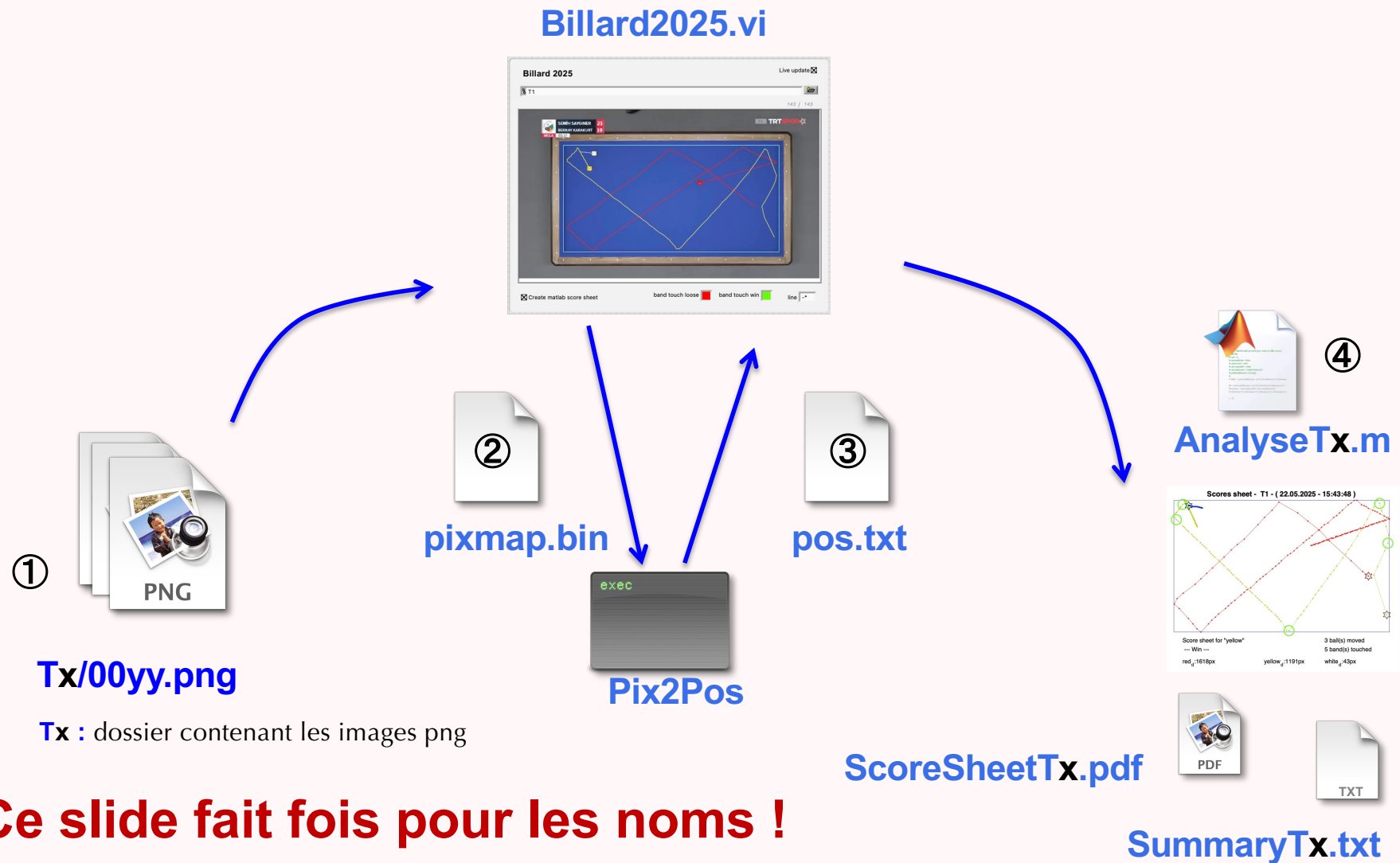
## **Pénalités assurées si:**

- **vous mettez des chemins absolus dans vos codes C/LabVIEW/matlab!**
- **vous ne respectez pas le *connector pane* (VI principal) fourni**
- **vous n'avez pas au minimum une fonction en C, un en matlab et un sousVI en LabVIEW**
- **vous omettez un/des fichiers dans l'archive zip! Testez que votre archive fonctionne correctement sur une autre machine**
- **vous vous êtes inspiré de codes trouvés sur internet sans mettre les références**
- **vous avez travaillé avec un autre groupe de manière significative sans l'indiquez**
- **vous avez employé ChatGPT ou équivalent, sans joindre le *transcript* de vos conversations en PDF**
- **vous employez des VLA**

# Procédure d'évaluation

Respectez impérativement les **noms** des fichiers, sinon mes outils automatiques ne fonctionnent pas!

Le nom du dossier contenant la séquence d'images (ex. XX) définit les noms des fichiers AnalyseXX.m, ScoreSheetXX.pdf et SummaryXX.txt



**Ce slide fait fois pour les noms !**

# Gestion des erreurs & warnings C

## **Erreurs** (impossible de continuer)

- Largeur et/ou hauteur en dehors de bornes [100..1000]
- Pas assez de pixels (1 à  $N$  pixels manquants)
- Impossible de lire dans le fichier d'entrée *pixmap.bin*
  - fichier manquant, mauvais nom, droit d'accès
- Impossible d'écrire dans le fichier de sortie *pos.txt*
  - lecture seule, droit d'accès
- Pas le bon nombre de paramètres dans la ligne de commande
- Diamètre de la boule (BallSize) en dehors de bornes [10..15] pixel

## **Warnings** (vous devez informer et continuer)

- Trop de pixels
- Moins que 3 boules -> retourner -1,-1,0 pour la (les) boules manquantes

**Ce slide fait fois pour les erreurs C !**

# Constantes/ranges pour les programmes

- **BallminScore: 15** (pixel) nombre minimum de pixels dans le bon range de couleur pour avoir une boule (C)
- **nbParams: 29 (+1)** nombre de paramètre passé en ligne de commande depuis LabVIEW (C)
- **BallBorderDist: 9** (pixel) définit si la boule touche le bord ou non (matlab)
- **MoveDistPx: 9** (pixel) distance minimum (1 segment) pour considérer que la boule a bougé (matlab)
- Le slide *Pix2Pos* – détection des balles, contient les ranges de couleurs

**Ce slide fait fois pour les constantes !**

# Slide supplémentaires

# Matlab – SummaryXX.txt

Le fichier *SummaryXX.txt* au format text contient le résumer du fichier *ScoreSheetXX.pdf*, son format est le suivant:

ou par ex. pour ScoreSheetT1

```
f:w; s:l; n:3; b:5; rb:1618; yb:1191; wb:43;
```

**f**(irst ball): **w** couleur de la 1<sup>e</sup> boule, char, valeurs possibles [w,y,r] (white, yellow, red)

**s**(core): **l** gagné/perdu, char, valeurs possibles [w,l] (win/lose)

**n**(ombre): **3** nombre de boules ayant bougé, entier

**b**(ords): **5** nombre de bords touché par la 1<sup>e</sup> boule, entier

**rb**(red ball): **1618** distance parcourue pas la boule rouge, entier

**yb**(yellow ball): **1191** distance parcourue pas la boule jaune, entier

**wb**(white ball): **43** distance parcourue pas la boule blanche, entier

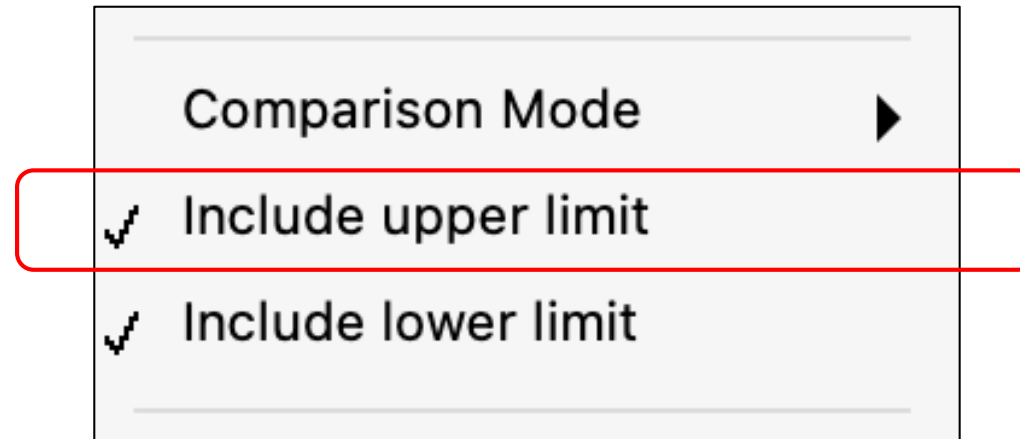
Ces valeurs sont lues par le VI principal Billard2025.vi, le format est important car l'analyse du résultat est automatisée.

A la fin de l'exécution du VI principal le champs **summary** contiendra soit le contenu du fichier *SummaryXX.txt* soit une erreur

# Dark Blue frame

- In FindBox.vi, ranges limits **are included!**

Also filled



*Ou code équivalent*

# Notes

- Les tests pour les projets Windows se feront sur la VM
  - Merci de vous assurer que votre code fonctionne sur la VM (si vous travaillez sur Windows)
- Tester la séquence [BadSeq.zip](#) (page moodle) c'est un dossier contenant une séquence d'une seule image PNG **qui n'est pas lisible** car c'est un fichier word renommé, votre programme doit reporter une erreur.
- Tester aussi votre code en renommant votre exécutable, votre programme doit reporter une erreur.
- Le rectangle/frame dans le PDF est à calculer via min/max des traces, ce n'est pas le rectangle bleu foncé (dakk blue), ils sont souvent proches mais pas dans tous les cas
- Vous pouvez avoir des variations de quelques pixels pour les distances parcourues par les boules.
- C'est **lose** et pas **loose** (my bad)
- Dans LabVIEW, définissez et sauvegardez des valeurs par défaut
  - Voir slide suivant

# Open a pdf file

From Matlab

```
uiopen('Full/Path/to/myfile.pdf',1)
```

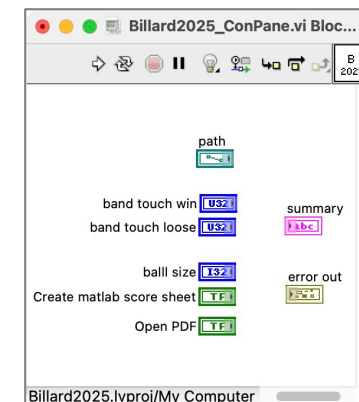
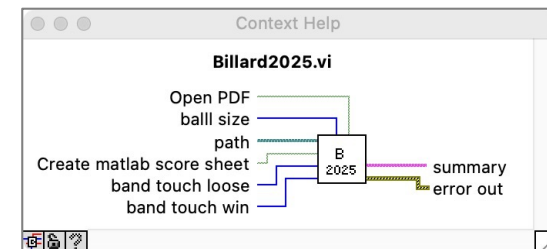
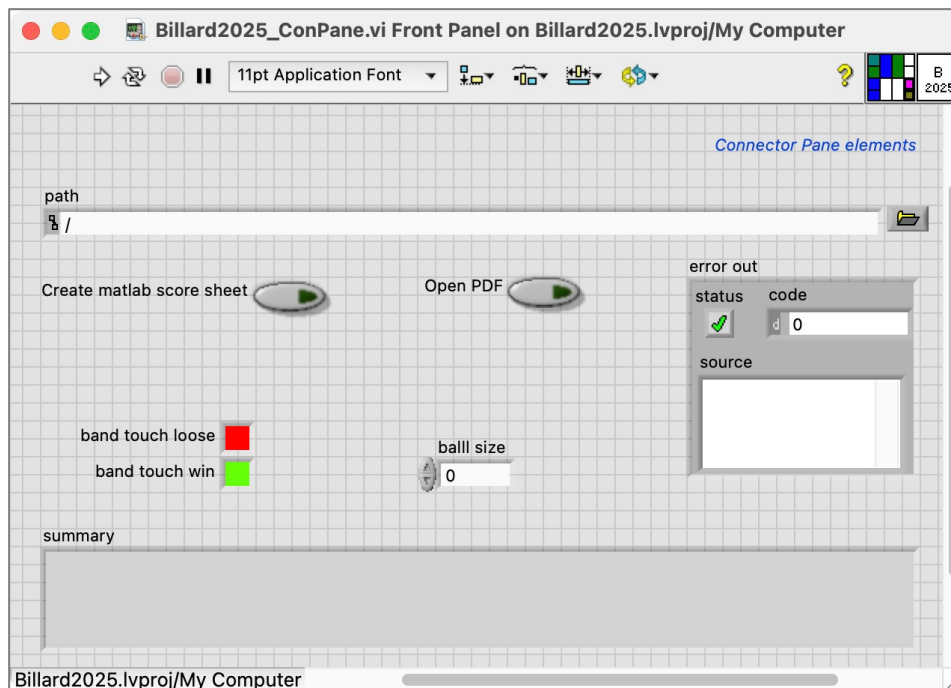
**1:** pas de dialog

From LabVIEW via **System Exec**

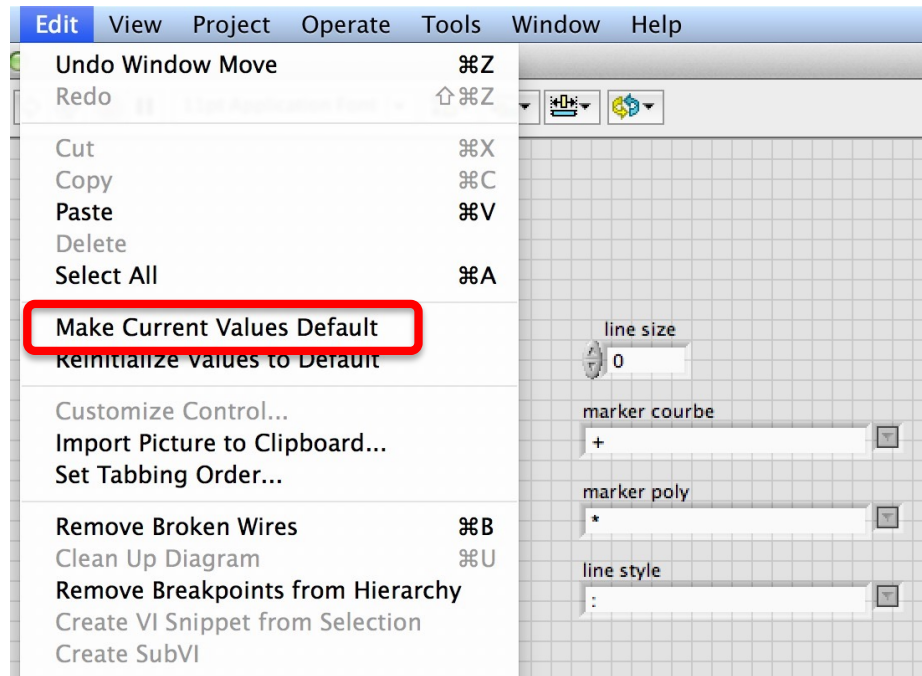
```
./open "Full/Path/to/myfile.pdf"
```

# Appel automatique de votre VI principal

- Afin de faciliter l'évaluation de votre projet le VI "CallBillard2025.vi" va appeler votre VI principal "Billard2025.vi"
- Pour cela le *connector pane* de votre VI principal doit suivre scrupuleusement le modèle fourni "Billard2025\_ConPane.vi" sur la page moodle du cours
- Les attributs pour le format de la ligne sont manquants, ne **pas** les ajouter dans le *connector pane*



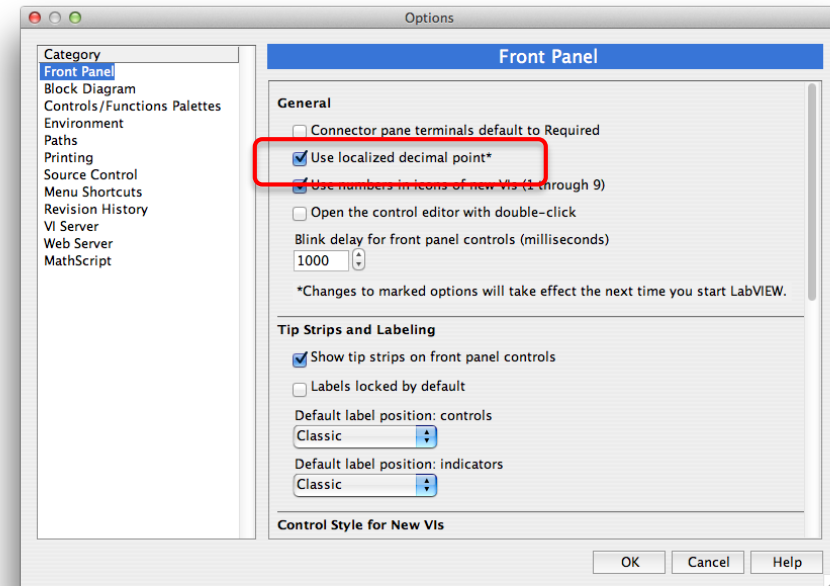
# Valeurs par défaut dans LabVIEW



L'option *Make Current Values Default* (ou *Make selected Values default*) permet de sauver les valeurs des controles. Ne pas oublier de sauvez le VI une fois les valeurs définies.

# Séparateur décimal

- Par défaut LabVIEW respecte le choix du séparateur décimal défini dans votre OS.
- Ce séparateur est souvent la virgule ',' et non pas le point '.'
- Matlab ne connaît que le point '.' comme séparateur décimal, la virgule servant à séparer des éléments.
- Soit vous changez le séparateur décimal dans l'OS, ce qui peut entraîner d'autres problèmes, soit vous dites à LabVIEW de ne plus respecter le choix de l'OS et de prendre la solution US qui est le '.'



- **Pour cela allez dans le menu Tools-Option de LabVIEW et décochez l'option "use Localized decimal point\*"**
- **Attention vous devez redémarrer LabVIEW pour que la modification soit prise en compte**

**Note:** LabVIEW permet de spécifier le séparateur décimal lors de la définition du formatage d'une string en utilisant '%,;' pour un séparateur ',' ou '%.;' pour le '.' par ex: %.,;%f. Cette manière de faire peut vite devenir compliquée, la solution globale est préférable