


Rallye Sciences 2024

La science olympique

Sujet Sciences Numériques

Fichiers mis à disposition par l'enseignant :

- cet énoncé au format pdf ;
- les fichiers *figure.sb3* et *figure.py* nécessaires à l'activité numéro 1 ;
- le fichier *seance_tapis.csv* qui contient les données pour l'activité 3 ;
- le dossier *RS_2024_SN* qui contient notamment
 - o la page web *rallye_sciences.html*, à ouvrir dans un navigateur web, dans laquelle sera saisie toute réponse à une question repérée par l'icône . L'appui sur le bouton *Télécharger les résultats* en bas de page génère automatiquement un fichier au format json ;
 - o la page web *aide_tableur.html*, à ouvrir dans un navigateur web, utile pour l'activité 3.

Matériel nécessaire :

- un ordinateur avec accès internet ;
- un téléphone avec scanner de QR-code (celui de l'enseignant, par exemple).

Production finale

Vous devrez remettre à votre enseignant les trois fichiers suivants.

- 1) Un fichier *figure.sb3* ou *figure.py* complété à l'issue de l'activité 1.
- 2) Un fichier au format *json* généré par l'appui sur le bouton *Télécharger les résultats* de la page web *rallye_sciences.html*.
- 3) Une image au format *jpg*, ci-après désignée par l'expression « e-ticket » : la page 2 est consacrée à sa description.

Remarques :

- Bien penser à enregistrer vos productions !
- Les trois activités contribuent à la production finale mais peuvent être traitées indépendamment.

e-ticket : création graphique


Objectif : réalisation d'un e-ticket pour les Jeux Olympiques.

Format attendu : image au format jpg.

Contenu du e-ticket :

- une photo d'athlète : intégrer une photo d'un athlète, libre de droits, en arrière-plan. Cette photo doit représenter l'esprit olympique et mettre en valeur le sport concerné ;
- les anneaux Olympiques : insérer une capture d'écran des anneaux olympiques que vous aurez créés lors de l'activité 1.
- le QR Code : ajouter une capture d'écran du QR code obtenu dans l'activité 2 ; ce code doit être clairement visible sur le e-ticket et un scan de ce code par un téléphone doit permettre d'afficher le message secret.
- la date de l'événement : la date de l'événement, récupérée dans l'activité 3, doit être clairement affichée sur le e-ticket.

Références aux activités :

Chaque élément à intégrer est signalé par une icône de ticket  dans les activités.

Il faut se reporter à ces icônes pour savoir précisément ce qui doit être intégré dans le e-ticket.

Logiciels suggérés pour la réalisation :

- Microsoft *Paint* sur Windows : un outil simple pour la retouche d'images.
- *GIMP* (<https://www.gimp.org/downloads/>) : un outil plus complexe mais avec davantage de fonctionnalités.
- *Minipaint* (<https://viliusle.github.io/miniPaint/>) : une alternative libre, en ligne, pour ceux qui n'ont pas accès à un logiciel local satisfaisant.
Pour générer l'image à la fin de la création : choisir *Export...* dans l'onglet *File*, nommer l'image *ticket* et sélectionner *JPG* comme format d'image, avant de cliquer sur le bouton *OK*.

Consignes supplémentaires : la créativité et l'esthétisme sont des éléments importants pris en compte dans l'évaluation de la production finale. Ainsi,

- veiller à la lisibilité, la qualité et la résolution de l'image finale ;
- l'ensemble des éléments doit être harmonieux et esthétique ;
- s'assurer que tous les éléments sont bien lisibles et que le e-ticket donne envie d'être utilisé.

Activité 1

L'activité consiste à appliquer un algorithme pour créer les cinq anneaux, symboles des Jeux Olympiques, soit avec le logiciel Scratch, soit en langage Python.

Le langage de programmation est au choix des élèves. Enregistrer régulièrement les productions.

Partie A

Version Scratch :

Si le logiciel n'est pas installé dans l'établissement, on pourra utiliser l'application en ligne accessible à ce lien : <https://scratch.mit.edu/projects/editor/?tutorial=getStarted>

Ouvrir le fichier *figure.sb3*.

À partir du bloc `pentagone`, coder un bloc `anneau` qui permet de dessiner une figure ressemblant à un cercle.

Version Python

Ouvrir le fichier *figure.py*

La fonction `figure` définie permet de tracer une figure géométrique à n côtés, de la couleur et aux coordonnées de départ fournies en entrée.

Compléter le code pour avoir un programme fonctionnel.

Le tester pour obtenir un pentagone de couleur « purple » aux coordonnées (0 ; 0).

Utiliser le code pour avoir un programme qui trace une figure ressemblant à un cercle à partir de la fonction `figure` précédente.

Parties B et C communes aux deux langages de programmation :

Partie B

Dans le fichier ouvert précédemment, ajouter un script utilisant le bloc `anneau` (Scratch) ou la fonction `figure` (Python) qui permet de tracer les 5 anneaux olympiques.

Pour rappel, les couleurs des anneaux de gauche à droite sont bleu, noir, rouge pour ceux du haut, jaune et vert pour ceux du bas.



Faire une capture d'écran de votre logo et l'insérer sur le e-ticket.



Le dessin obtenu avec votre script est-il conforme au logo officiel des anneaux olympiques ?
Si ce n'est pas le cas, décrire les différences observées.

Partie C : prolongement

S'il reste du temps en fin d'épreuve et si cela n'a pas été fait dans la partie B, effectuer les modifications pour obtenir une image qui ressemble au logo officiel.



Faire une nouvelle capture d'écran et remplacer sur le e-ticket votre ancienne image par cette nouvelle image.

Ne pas oublier d'enregistrer le fichier *figure.sb3* ou *figure.py* complété pour le remettre à votre enseignant.

Activité 2

Les QR-codes sont des images, composées de pixels de couleur noire ou blanche, qui ont envahi notre quotidien. Ils permettent de stocker des adresses de sites, des messages, des images. Ils utilisent la notion de masques très utiles en informatique, qui permettent, entre autres, de cacher ou révéler des images.

L'opération **xor**

L'opération **xor** est une opération logique entre 2 pixels de couleur noire ou blanche.

$$\begin{array}{l} \square \text{ xor } \square = \square \\ \square \text{ xor } \blacksquare = \blacksquare \\ \blacksquare \text{ xor } \square = \blacksquare \\ \blacksquare \text{ xor } \blacksquare = \square \end{array}$$

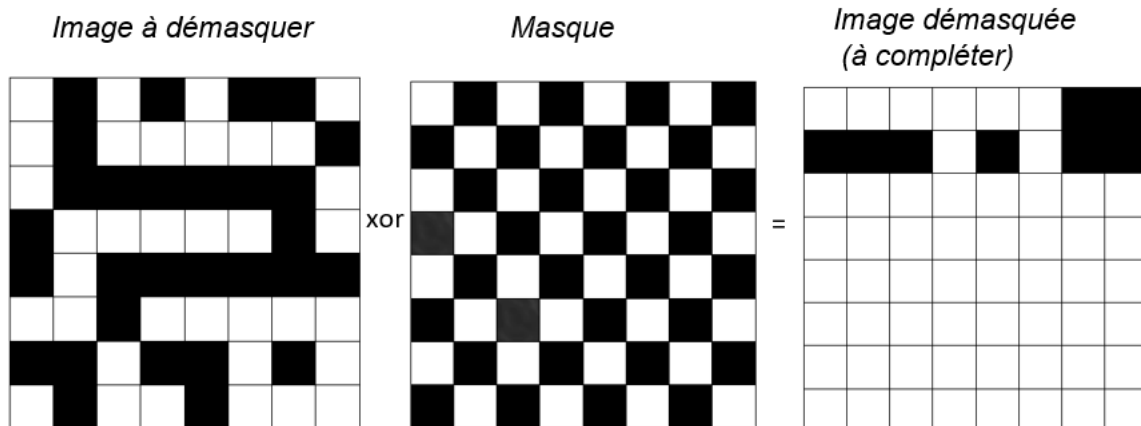
Figure 1 : opération **xor** entre 2 pixels

Question 1

Appliquer l'opération **xor** entre l'image à démasquer et le masque ci-dessous pour obtenir l'image démasquée.



Indiquer le nombre qui apparaît dans l'image démasquée.



Question 2

En page 5, on fournit une nouvelle image et un masque de taille 21 x 21 : en appliquant le masque sur l'image, on obtient un QR-code qui contient un message secret.



Appliquer l'opération **xor** entre l'image (figure 2) et le masque (figure 3) afin de reconstituer un QR-code ;

Mode d'emploi :

Dans la page [rallye_sciences.html](#), la modification de la couleur d'une case de l'image est possible soit en cliquant sur la case, soit en utilisant la barre d'espace en combinaison avec les flèches directionnelles du clavier.

Pour annuler la dernière action effectuée, vous pouvez cliquer sur le bouton **Annuler** situé en bas de la grille. Il est également possible de **Réinitialiser** la grille.



Scanner ce QR-code pour retrouver le message secret et saisir la réponse sur le fichier html.



Appuyer sur le bouton **Afficher/masquer la grille**, effectuer une capture du QR-code obtenu et insérer l'image dans le e-ticket de la production finale. Cette opération est possible, même si le QR-code n'est pas finalisé.

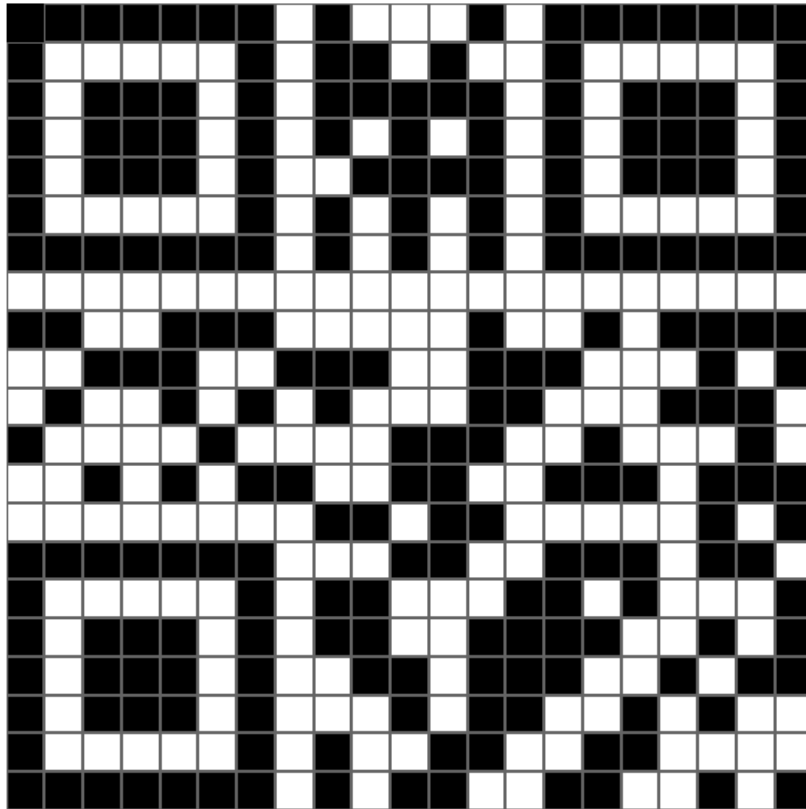


Figure 2 : image à démasquer, de taille 21 x 21

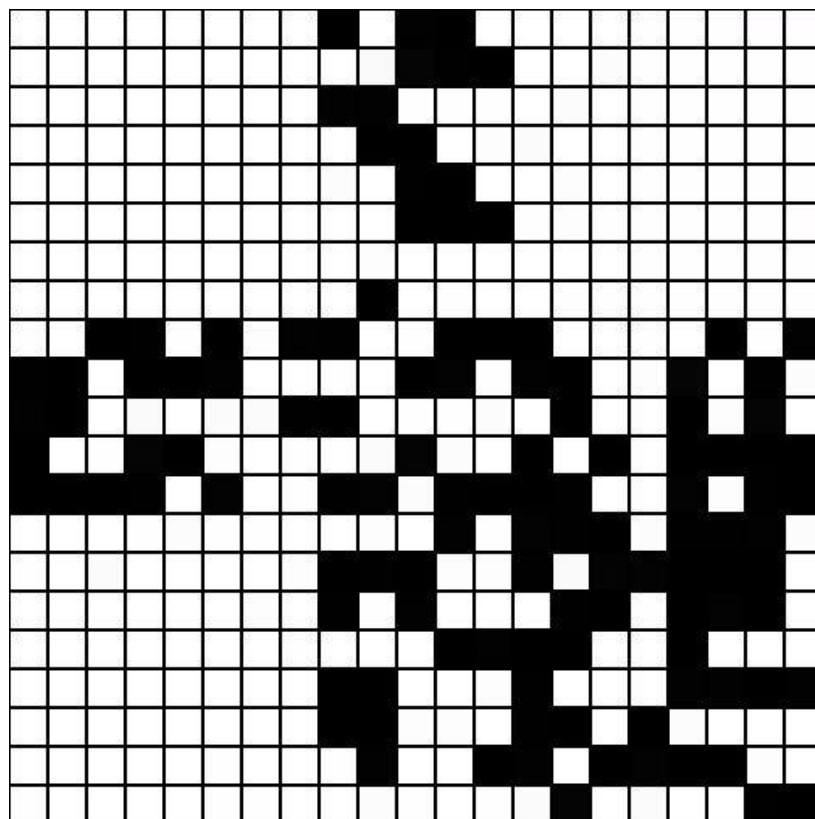


Figure 3 : masque, de taille 21 x 21

Au cœur d'un QR-code

De nombreuses applications proposent de fabriquer un QR-code automatiquement car il s'agit d'une construction complexe. Un QR-code est découpé en plusieurs zones qui contiennent des informations codées en binaire, le seul langage compris par un ordinateur.

Voici un extrait de la page Wikipedia :

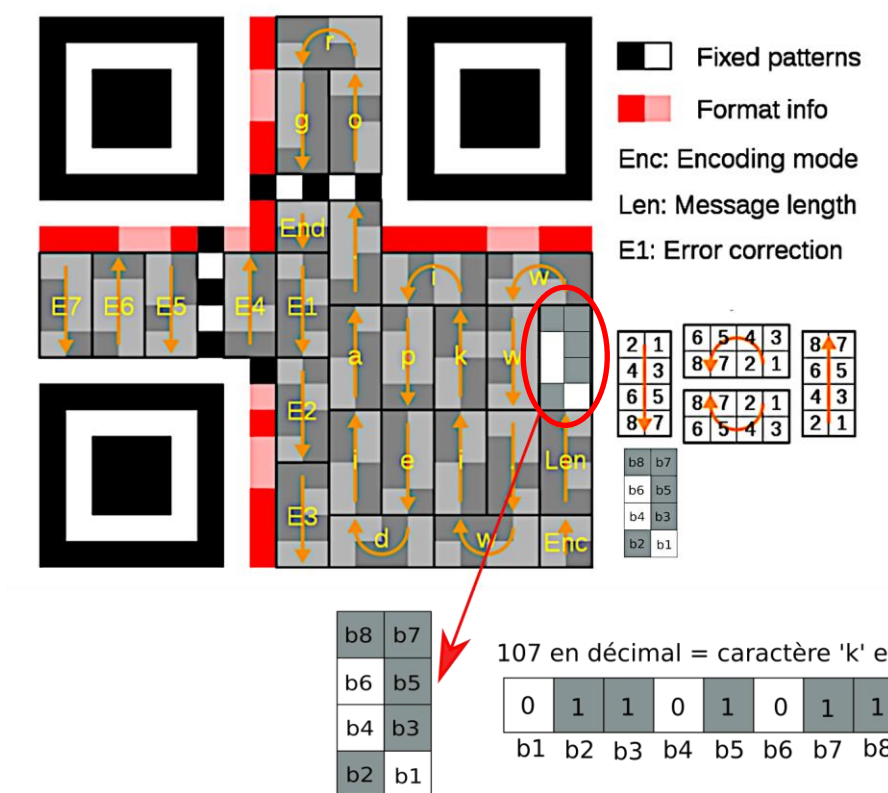


Figure 4 : QR-code de 21 x 21 pixels

Dans un QR-code, un pixel noir représente la valeur binaire 1 et un pixel blanc représente la valeur binaire 0.

Remarque : on pourra rechercher des informations concernant le codage ASCII pour répondre à la question suivante.

Question 3

La zone rectangulaire de huit pixels, identifiée dans l'image ci-dessus, contient le code binaire du caractère 'k', selon la norme de codage ASCII.

De la même façon, un caractère est codé dans la même zone rectangulaire sur le QR-code que vous avez obtenu dans la question 2.



Identifier cette zone sur votre QR-code puis indiquer sur le fichier html quel est ce caractère codé. Expliquer votre démarche.

Activité 3

Pour progresser dans une pratique sportive, il faut s'entraîner.

Pour mesurer les progrès, on se sert d'indicateurs comme par exemple la fréquence cardiaque.

On peut mesurer cette fréquence en plaçant deux doigts contre la gorge et en comptant les battements du cœur.



Quelle est la fréquence cardiaque en battement par minute si vous comptez 12 battements en 6 secondes ?

Les technologies numériques mettent à disposition des athlètes et des entraîneurs des informations plus nombreuses et précises pour mesurer les performances :

- des capteurs permettent de faire des mesures précises et massives, sans trop gêner l'athlète dans sa performance ;
- des plateformes numériques, comme Nöliö ou Strava, permettent d'analyser les grandes quantités de données produites par les capteurs.

Capteurs

Un bon capteur renvoie rarement des données erronées, mais cela peut toujours arriver. Il nous faut détecter ces erreurs dans la mesure du possible, pour les corriger et évaluer la fiabilité des données.

Une mesure en provenance du capteur peut sembler :


- *valide* : elle n'a sans doute pas été issue d'une erreur de mesure.
- *erronée* : elle résulte sans aucun doute d'une erreur de mesure, car elle est impossible à mesurer chez un humain.
- *suspecte* : à mi-chemin entre *erronée* et *valide*, c'est une donnée étonnante chez un humain, mais pas impossible pour autant.



À l'aide d'une recherche documentaire, préciser, pour chaque mesure de fréquence cardiaque ci-dessous, exprimée en battement par minute (bpm) si elle semble valide, erronée ou suspecte.

N° mesure	Fréquence cardiaque (bpm)
1	13
2	30
3	70
4	180
5	230
6	280
7	1000

Afficher les données


Pour traiter cette partie, des aides concernant l'utilisation du **tableur LibreOffice Calc** sont disponibles dans la page [web aide_tableur.html](http://web.aide_tableur.html). Ces aides sont repérées par l'icône .

L'utilisation de tout autre tableur est possible, mais les aides ne sont pas fournies.

Dans un tableur, ouvrir le fichier `seances_tapis.csv`. 

Dans ce fichier, se trouvent les données relatives à deux séances d'entraînement où l'athlète a couru sur un tapis roulant, en suivant le même programme, à 6 mois d'intervalle : la colonne A contient les données relatives à la première séance (séance 1) et la colonne B celles de la deuxième séance (séance 2).

Les données correspondent à la fréquence cardiaque de l'athlète mesurée chaque seconde, à partir de la première, pendant 5 minutes.

Représenter ces deux séries de données sur un même graphique. 



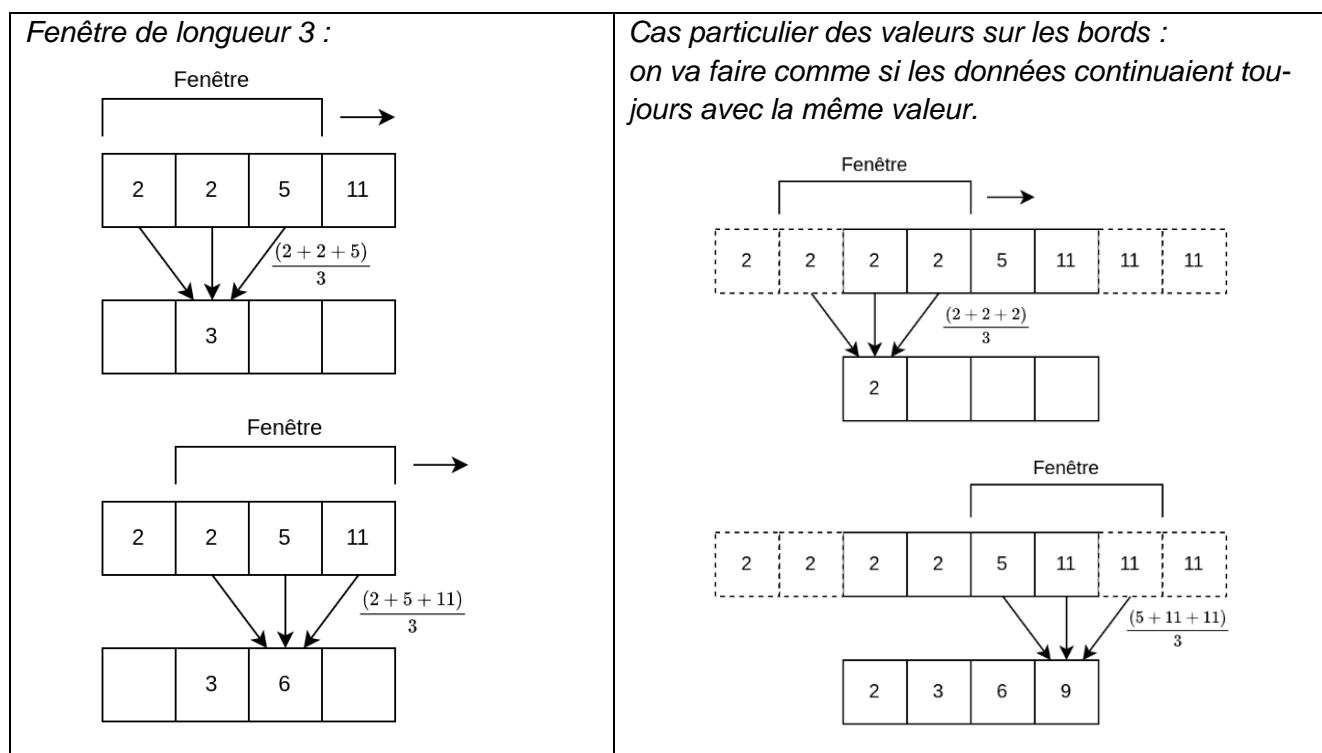
Parmi les trois périodes, entre 61 s et 79 s, entre 139 s et 169 s, entre 199 s et 223 s, déterminer laquelle correspond :

- Au plus gros effort ;
- Au plus petit effort ;
- À un effort d'intensité moyenne.

Les mesures, issues d'une montre connectée, sont relativement précises mais comportent du *bruit* : il s'agit de petites variations de quelques battements par minute à chaque mesure. Ce sont les "pointes" que l'on voit partout sur les courbes.

Lissage de données

Pour rendre les courbes plus simples à lire, on va aplatir les pointes (c'est ce qu'on appelle *un lissage*) en utilisant une *moyenne glissante* : chaque donnée est remplacée par une moyenne entre elle et ses voisines. La méthode est illustrée ci-dessous sur une série de 4 données : 2 ; 2 ; 5 ; 11.





On prend, comme dans les exemples précédents, une fenêtre de longueur 3.
Calculer les valeurs A, B et C :

Données avant lissage	8	8	2	2	5	11
Données après lissage	A	B	C	3	6	9

Pour appliquer cette méthode sur les 300 valeurs des deux entraînements, nous allons utiliser un tableur en considérant **une fenêtre de longueur 9**.



- Insérer une nouvelle colonne à droite des données de la séance 1 (entre les colonnes A et B).
- Saisir, dans la cellule B6, la formule permettant de calculer la moyenne des cellules de la plage A2:A10.
- Recopier cette cellule vers le bas, jusqu'à la cellule B297, pour obtenir les données lissées de la séance 1.
- Recommencer ces opérations pour obtenir, dans la colonne D, les données lissées de la séance 2.
- Afficher sur le même graphique les données lissées de la séance 1 et de la séance 2.



En comparant les données lissées obtenues à l'aide du tableur et sachant que la séance 1 et la séance 2 d'entraînement ont été réalisées dans les mêmes conditions, pensez-vous que l'athlète ait amélioré ses performances ? Expliquer votre réponse.

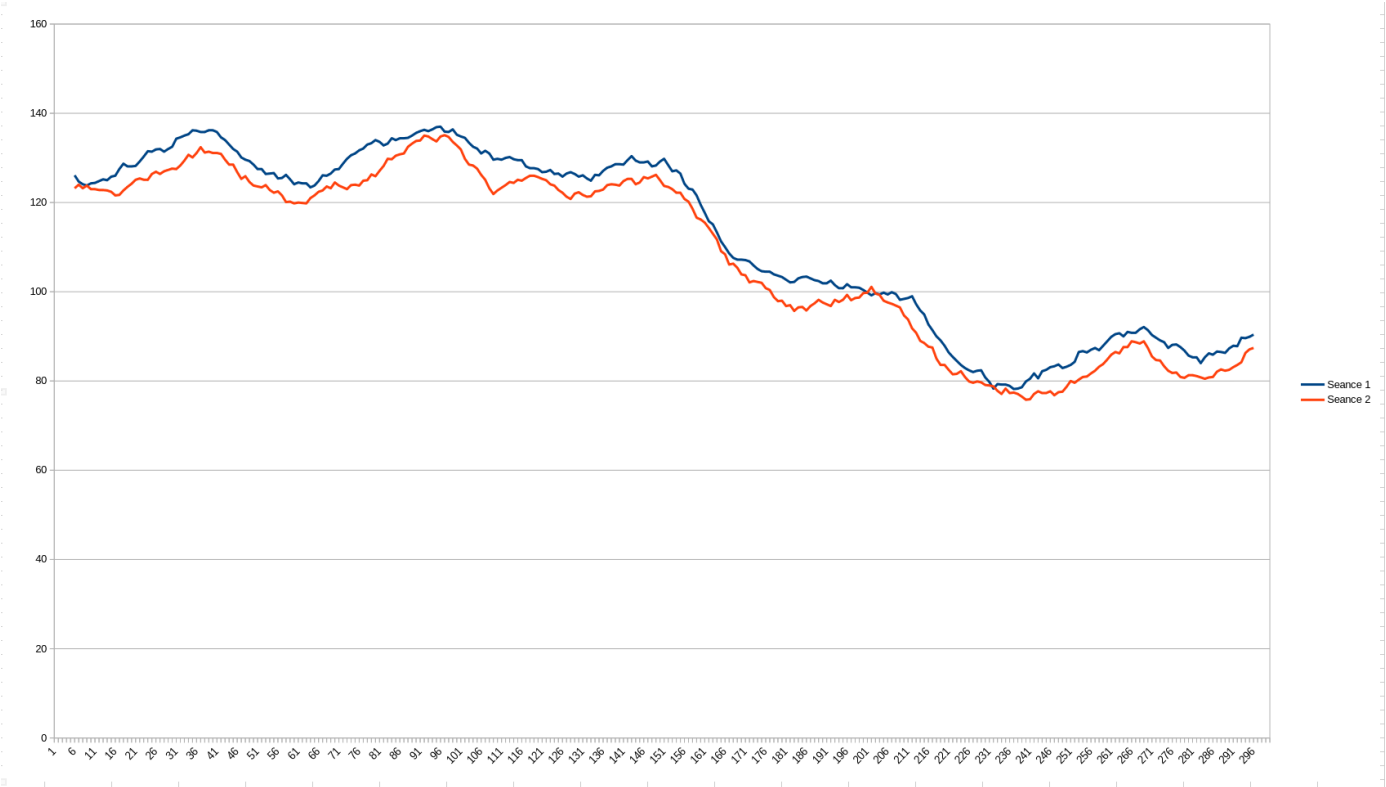
Remarque : sur certaines plateformes, l'analyse que nous avons faite avec Libre Office est automatique, et les entraîneurs et athlètes peuvent directement visualiser les courbes de performances.

Météo

Les athlètes s'entraînent en général sur des parcours et dans des conditions variées. Certaines plateformes permettent d'enregistrer de nombreuses autres données qui servent ensuite aux entraîneurs.

- Vitesse ajustée à la pente, qui est l'effort équivalent à une course sur du plat, pour les courses en montée ou en descente ;
- Altitude ;
- Météo ;
- ...

Voici deux courbes qui correspondent à deux séances réalisées par un athlète, sur le même parcours, en extérieur, à quelques jours d'intervalle. Il n'a donc pas eu le temps de progresser entre les deux, ni de régresser. Pourtant, les deux courbes montrent une fréquence cardiaque différente.



Voici maintenant quatre paires de bulletins météo.

La séance 1 est la première à avoir été réalisée.



Parmi les quatre paires de bulletins météo suivantes, déterminer celle qui correspond à ces deux séances d'entraînement : ajouter, sur votre e-ticket, la date de la première séance d'entraînement.

