

Devoir d'informatique

1 Relevé d'une course d'Ussain Bolt

Une mesure de vitesse a été réalisée lors du 100 m réalisé par Ussain Bolt lors du des championnats du monde de Berlin en 2009.

Les vitesses mesurées sont stockées dans la liste `les_v`. La figure 1 représente le tracé des vitesses en fonction de leur indice dans la liste `les_v`.

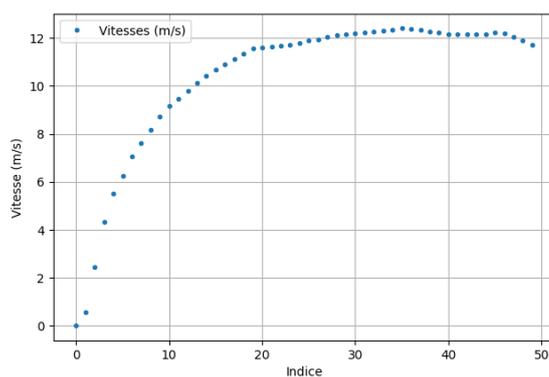


FIGURE 1 – Tracé de la vitesse lors du 100 m d'Ussain Bolt

Questions préliminaires

Question 1 Écrire une fonction `conversion(v:float) -> float` permettant de convertir une vitesse en ms^{-1} en kmh^{-1} .

Correction

```
def conversion(v:float) -> float:  
    return v*3600/1000
```

Question 2 Écrire une fonction `conversion_l(v:[float]) -> [float]` permettant de convertir une liste de vitesses en ms^{-1} en une liste de vitesses en kmh^{-1} .

Correction

```
def conversion_l(v:[float]) -> [float]:  
    les_v = []  
    for x in v :  
        les_v.append(x*3600/1000)
```

```

    les_v.append(conversion(x))
return les_v

```

Une mesure a été faite toutes les 0,2 s.

Question 3 Écrire une fonction `creer_les_t(n:int, pas:float)-> [float]` permettant de renvoyer une liste de `n` valeurs séparées par un pas `pas`. Ainsi `creer_les_t(3, 0.2)` devra renvoyer `[0,0.2,0.4]`.

Correction

```

def creer_les_t(n:int, pas: float) -> [float] :
    t = 0
    les_t = []
    for i in range(n):
        les_t.append(t)
        t = t +pas
    return les_t

```

Question 4 Donner une instruction (ou une suite d'instructions) permettant de créer la liste `les_t` associée aux vitesses mesurées `les_v`.

Correction

```

les_t = creer_les_t(len(les_v),0.2)

```

Question 5 En utilisant la bibliothèque `matplotlib.pyplot` et la fonction `plot`, donner les instructions permettant de tracer la vitesse en kmh^{-1} en fonction du temps.

Correction

```

import matplotlib.pyplot as plt
les_vv = conversion_l(les_v)
plt.plot(les_t,les_vv)
plt.show()

```

Dans les questions suivantes on interdit l'utilisation de `max`, `min`, `index`.

Question 6 Écrire une fonction `vitesse_max(les_v:[float])-> float` qui renvoie la vitesse maximale d'une liste de vitesses.

Correction

```

def vitesse_max(les_v:[float]) -> float :
    vmax = les_v[0]
    for v in les_v :
        if v>vmax :
            vmax = v
    return vmax

```

Question 7 Écrire une fonction `indice_vitesse_max(les_v:[float])-> int` qui renvoie l'indice de la vitesse maximale d'une liste de vitesses.

Correction

```

def indice_vitesse_max(les_v:[float]) -> int :
    ivmax = 0
    for i in range(len(les_v)) :
        if les_v[i]>les_v[ivmax] :

```

```

        ivmax = i
    return ivmax

```

Détermination de l'instant d'arrivée

On souhaite utiliser les vitesses mesurées pour déterminer l'instant auquel le coureur franchit la ligne d'arrivée. Pour cela on propose d'estimer les positions successives du coureur à l'aide de la méthode des trapèzes. On note $x_i = x(t_i)$ la valeur de la position estimée au i^e instant de mesure. La position initiale x_0 sera prise nulle.

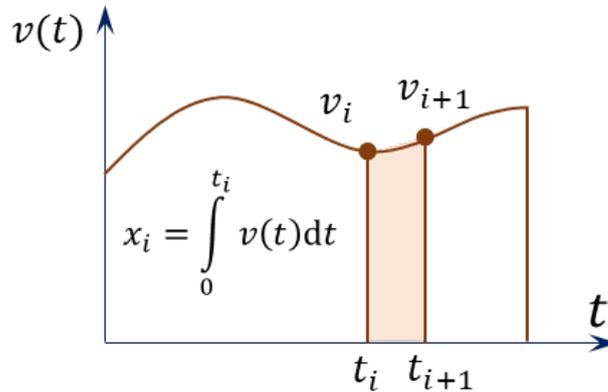


FIGURE 2 – Intégration par la méthode des trapèzes

Question 8 Donner la relation entre x_{i+1} , x_i et $\int_{t_i}^{t_{i+1}} v(t)dt$.

Correction $x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} v(t)dt$

Question 9 Donner l'expression approchée de $\int_{t_i}^{t_{i+1}} v(t)dt$ en fonction de v_i , v_{i+1} , t_i et t_{i+1} . En déduire une estimation de x_{i+1} en fonction de x_i , v_i , v_{i+1} , t_i et t_{i+1} .

Correction $x_{i+1} = x_i + \frac{v_{i+1} + v_i}{2} (t_{i+1} - t_i)$

Question 10 Écrire une fonction `integre(les_v:[float], les_t:[float]) -> [float]` permettant d'intégrer les vitesses et renvoyant la liste des positions estimées.

Correction

```

def integre(les_v:[float], les_t:[float]) -> [float]:
    les_x = [0]
    for i in range(len(les_v)-1):
        dx = 0.5*(les_t[i+1]-les_t[i])*(les_v[i+1]+les_v[i])
        les_x.append(les_x[-1]+dx)

    return les_x

```

Question 11 Écrire une fonction `chrono(les_x:[float], les_t:[float]) -> float` prenant comme argument la liste des positions et la liste des temps et permettant d'abord de déterminer l'indice pour lequel la distance de 100 m est dépassée. Cette fonction renverra alors le temps pour lequel les 100 m ont été atteints.

Correction

```
def chrono(les_x:[float],les_t:[float]) -> float:
    i = 0
    while les_x[i]<100 :
        i = i+1
    return les_t[i]
```

Question 12 Écrire une fonction `transforme(les_t:[float],les_x:[float],les_v:[float]) -> [[float]]` permettant de transformer les 3 listes en une liste de listes. Cette liste sera de la forme `[[les_t[0], les_x[0], les_v[0]], [les_t[1], les_x[1], les_v[1]], ...]`.

Correction

```
def transforme(les_t:[float],les_x:[float],les_v:[float]) -> [[float]] :
    les_res = []
    for i in range(len(les_t)):
        v = [les_t[i],les_x[i],les_v[i]]
        les_res.append(v)
    return les_res
```

2 Codage des nombres sur une carte Arduino

Les entrées analogiques de la carte permettent de coder des informations sur 10 bits.

Question 13 Combien de nombres entiers est-il possible de coder sur 10 bits ?

Correction $2^{10} = 1024$.

Question 14 Si on souhaite coder des entiers (positifs), donner le plus petit et le plus grand entier codables.

Correction Codage de 0 à 1023.

Question 15 Si on souhaite coder des entiers relatifs, donner le plus petit et le plus grand entier codables.

Correction Codage de -512 à 511.

Question 16 Convertir $(23)_{10}$ en binaire puis en hexadécimal.

Correction $(23)_{10} = (10111)_2 = (17)_{16}$

Question 17 Le nombre $(203)_{16}$ permet de coder un entier relatif. Quel est-il ?

Correction $(203)_{16} = (515)_{10} - 1024 = -509$.