

Calcul approché de longueur d'une portion de courbe

Compétences visées

- **chercher**, expérimenter – en particulier à l'aide d'outils logiciels ;
- **modéliser**, faire une simulation, valider ou invalider un modèle ;
- **représenter**, choisir un cadre (numérique, algébrique, géométrique...), changer de registre ;
- **calculer**, appliquer des techniques et mettre en œuvre des algorithmes.

Ces compétences sont mises en œuvre dans le cadre de l'extrait du programme de 2^{nde} GT ci-dessous :

« Algorithme de calcul approché de longueur d'une portion de courbe représentative de fonction. »

Situation déclenchante

Estimation de la longueur d'une courbe

On sait que la trajectoire d'un ballon est une portion de parabole, représentant une fonction polynôme du second degré.

On souhaite déterminer **la longueur du trajet du ballon** entre le moment où il quitte les mains du joueur et l'arrivée au panier.

Dans un souci de simplification, on traitera dans un premier temps une fonction de référence (ici, la fonction carré sur l'intervalle $[0 ; 1]$ et on reviendra à ce problème dans un second temps).



Problématique

Soit f la fonction définie sur l'intervalle $[0 ; 1]$ par : $f(x)=x^2$ et C sa courbe représentative.
Quelle est la longueur de la courbe C ?

Fiche méthode

Proposition de résolution

Le principe est d'approcher la courbe par une ligne brisée composée de segments dont on ajoutera les longueurs.

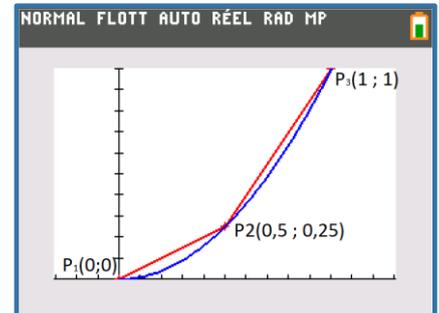
- On crée la fonction **longseg()** qui donne la longueur du segment [AB], avec $A(xA ; f(xA))$ et $B(xB ; f(xB))$; cette fonction a comme paramètres x_A et x_B .
- Ci-contre, on approche la courbe par deux segments. Ces longueurs sont données par la fonction **longseg(.,.)**, et la longueur totale par **longtot(.,.)**: $P_1P_2 = \text{longseg}(0,0.5) \sim 0,866$

$$P_2P_3 = \text{longseg}(0.5,1) \sim 0,579$$

La longueur totale est la somme de ces longueurs, donnée par **longtot(0,1,2)** qui signifie que l'on travaille entre les abscisses 0 et 1, avec 2 segments.

Ainsi, on crée trois fonctions dans ce programme :

- Une fonction **f()** qui a pour paramètre x et qui renvoie l'image de x par la fonction utilisée dans le contexte, la fonction carré dans ce cas;
- Une fonction **longseg()** qui a pour paramètres x_A et x_B et qui renvoie la longueur du segment [AB] avec $A(x_A ; f(x_A))$ et $B(x_B ; f(x_B))$;
- Une fonction **longtot()** qui a pour paramètres x_A , x_B et n ; elle renvoie la somme des longueurs des n segments qui approchent la courbe.



```
PYTHON SHELL
>>> longseg(0,0.5)
0.5590169943749475
>>> longseg(0.5,1)
0.9013878188659973
>>> longtot(0,1,2)
1.460404813240945
>>> |
```

Remarques

importation en préambule du code de la bibliothèque « math » par « **from math import *** » pour pouvoir utiliser la fonction **sqrt()** (racine carrée)

```
EDITEUR : LONGCRBE
LIGNE DU SCRIPT 0002
from math import *

```

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus



Fiche méthode

Etapes de résolution

Création de la fonction $f(x)=x^2$

Fonction `longseg(xA,xB)` donnant la longueur d'un segment [AB] avec $A(xA ; f(xA))$ et $B(xB ; f(xB))$
On notera l'utilisation de la fonction `f()` précédemment définie qui rend l'écriture de cette fonction simple et proche des notations usuelles

```
EDITEUR : LONGCRBE
LIGNE DU SCRIPT 0010
from math import *

def f(x):
    return x**2

def longseg(xA,xB):
    return sqrt((xB-xA)**2+(f(xB)-f(xA))**2)
```

C'est un principe à retenir : l'utilisation successive de fonctions en python rend le programme dans son ensemble plus lisible et plus efficace.

Fonction `longtot(xA,xB,n)` qui a pour paramètres les abscisses des extrémités de l'intervalle considéré et `n` le nombre d'intervalles en lequel l'intervalle $[xA ; xB]$ est découpé.

```
EDITEUR : LONGCRBE
LIGNE DU SCRIPT 0014
def longseg(xA,xB):
    return sqrt((xB-xA)**2+(f(xB)-f(xA))**2)

def longtot(xA,xB,n):
    pas=(xB-xA)/n
    long=0
    for i in range(n):
        long=long+longseg(xA+i*pas,xA+(i+1)*pas)
    return long
```

Le principe de cette fonction est d'utiliser la fonction précédente `longseg()` sur des intervalles de type $[xA + i * pas ; xA + (i+1) * pas]$ où le pas est égal à $\frac{x_B-x_A}{n}$.

On cumule la longueur de chaque segment dans la variable `long` pour approximer la longueur totale de la courbe, valeur retournée par la fonction.

Retour au problème

Ainsi, ce programme permet de répondre au problème de départ en donnant une valeur approchée de la longueur de la courbe représentative de la fonction carré sur l'intervalle $[0 ; 1]$.

On trouve ici une longueur d'environ 1,47894.

```
PYTHON SHELL
>>>
>>> longtot(0,1,10)
1.478197397487329
>>> longtot(0,1,100)
1.478935403974238
>>> longtot(0,1,1000)
1.478942783008997
>>> longtot(0,1,10000)
1.478942856799243
>>> longtot(0,1,50000)
1.478942857514788
```

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus



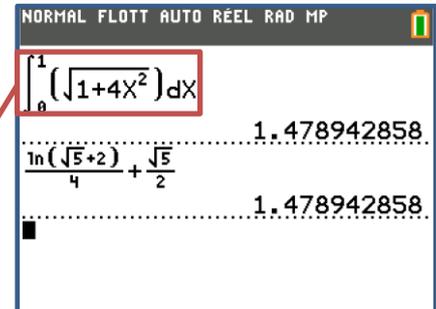
Fiche méthode

Un peu de théorie

Sur le plan théorique, la longueur de la courbe représentative d'une fonction f sur un intervalle $[a ; b]$ (sur lequel elle est dérivable) est donnée

$$\text{par : } \int_a^b \sqrt{1 + f'(x)^2} dx$$

Cela permet d'obtenir la valeur exacte par rapport au problème initial ainsi qu'une valeur approchée et d'apprécier la précision de l'algorithme selon la valeur de n .



Retour à la situation déclenchante

En important l'image du ballon de basket dans la calculatrice et en la mettant en arrière-plan, on peut approcher la trajectoire du ballon par une courbe représentant une fonction du second degré.

Remarque : on réalise cette importation par le biais de TI-Connect ; on affiche ensuite l'image en tant qu'« Arrière-plan » (passer par « Format » / « Arrière-plan » et les flèches directionnelles pour faire défiler les images enregistrées dans la calculatrice).

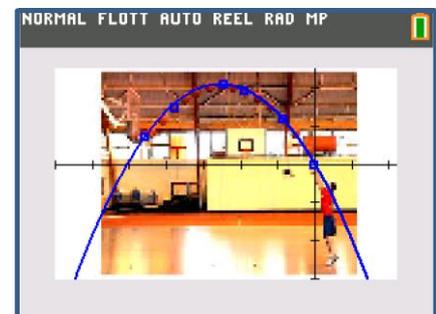
Il faudra adapter le repère pour que les coordonnées soient réalistes (taille du joueur et distance au panier).

On obtient comme équation $y = -0,3277x^2 - 1,6645x$ en ayant choisi $X_{\min}=-7 / X_{\max}=2.2 / Y_{\min}=-3$ et $Y_{\max}=2.5$

On a pour cela utilisé une fonctionnalité de la TI 83 Premium CE permettant de déterminer une fonction approchant des points (menu stats, puis CALC, puis E:TracéAjust-éq ; il faut alors placer les points qui seront interpolés. On termine par éQRég et le choix RégDeg2).

Les abscisses sont à prendre entre -4,56 (abscisse du panier) et 0 (abscisse des mains du joueur ; on a calé l'origine volontairement au niveau des mains du joueur).

On obtient une trajectoire - entre les mains du joueur et le panier - d'une longueur d'environ 5,95 m en ayant rentré l'expression obtenue ci-dessus pour la fonction $f()$ du programme.



```
PYTHON SHELL
>>> longtot(-4.56,0,1000)
5.954377643552278
>>> |
Fns... a A # Outils Éditer Script
```

```
PYTHON SHELL
>>>
>>> longtot(-1,1,4)
3.035276180410083
>>> longtot(-1,1,10)
3.115105950558346
>>> longtot(-1,1,100)
3.140760589842466
>>> longtot(-1,1,1000)
3.141566356216479
>>> longtot(-1,1,10000)
3.141591822039785
Fns... a A # Outils Éditer Script
```

Une application intéressante

En appliquant la méthode précédente à la fonction $f(x) = \sqrt{1-x^2}$ sur l'intervalle $[-1 ; 1]$, on obtient une approximation de π

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus

