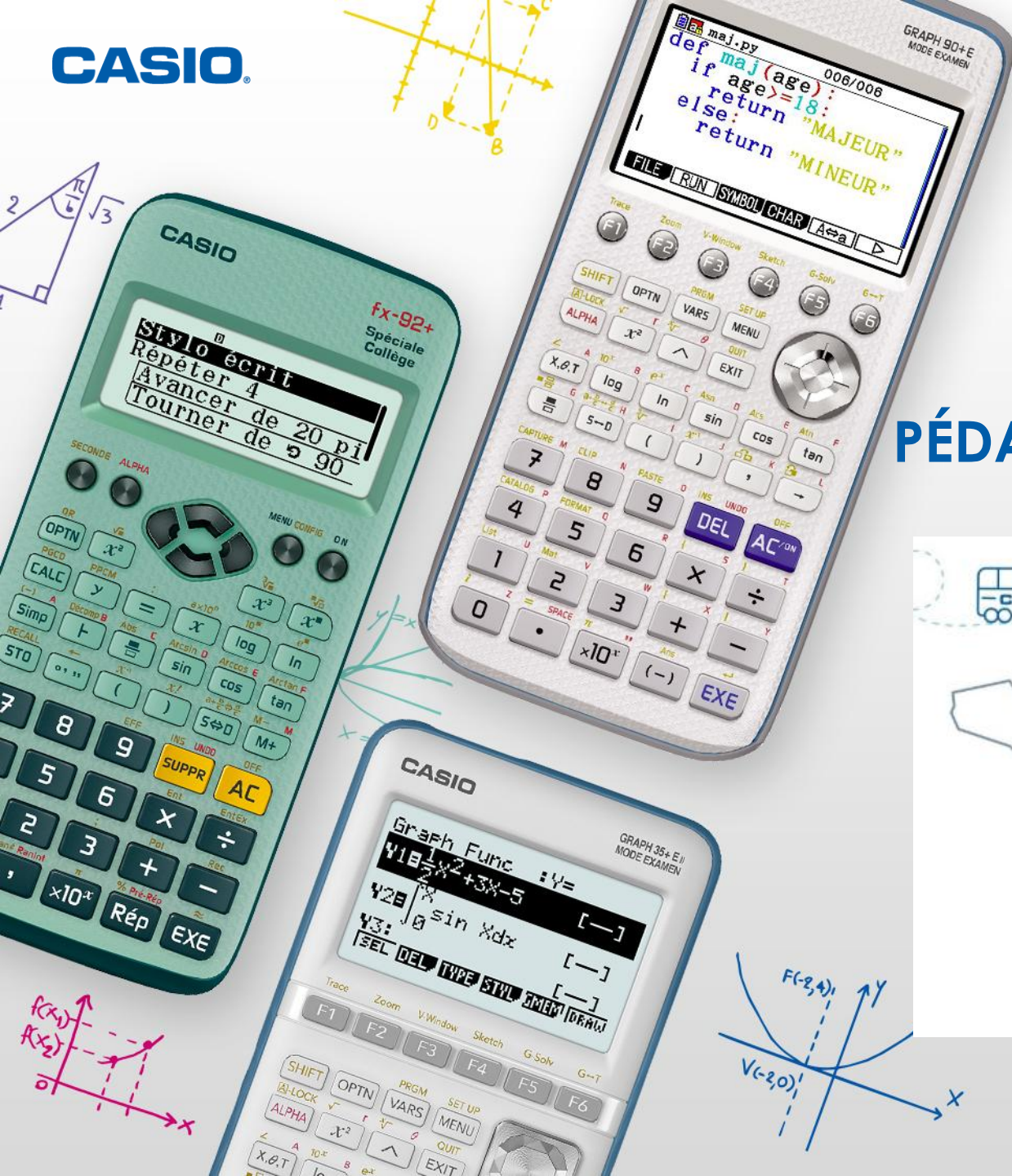
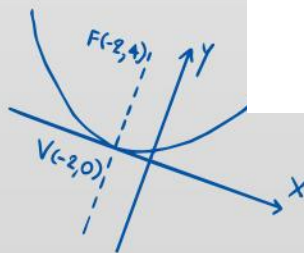
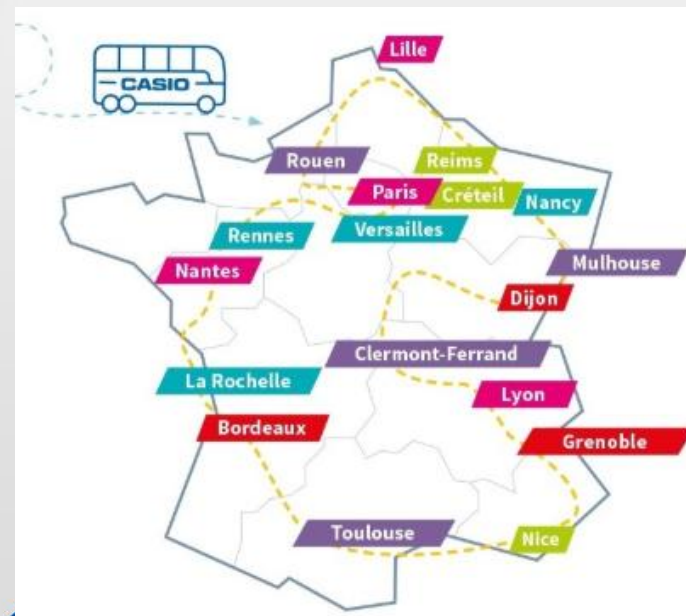


CASIO



TOURNÉE PÉDAGOGIQUE CASIO



PRIMAIRE

COLLÈGE

LYCÉE

ÉTUDES SUPÉRIEURES



Petites fx



FX JUNIOR +



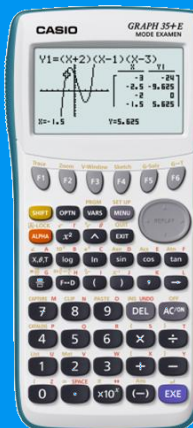
fx-92+
Spéciale
Collège



Création de QR codes pour
visualiser des graphiques /
programmes sur l'application
mobile CASIO EDU+



Graph 25+E



Graph 35+E

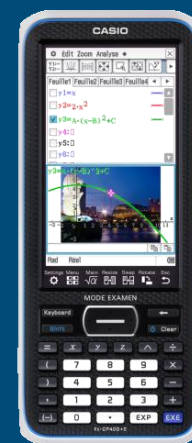


Graph 35+E II



Graph 90+E

LANGUAGE PYTHON
INTÉGRÉ



fx-CP400+E

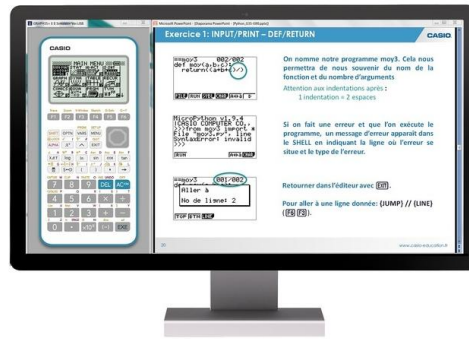
MODE EXAMEN INTÉGRÉ

fx-92+ Spéciale Collège



fx-92+ Spéciale Collège Emulator
(Windows)

Graph 35+E II



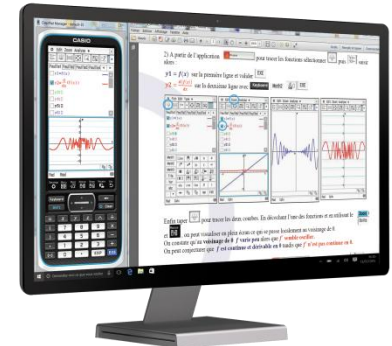
fx-Manager Plus
(Windows)
Graph 35+E II

Graph 90+E



fx-CG Manager PLUS
(Windows / Mac)

fx-CP400+E



ClassPad Manager
(Windows / Mac)

Une licence gratuite pour l'émulateur de votre choix sur casio-education.fr

NOUVEAU EMULATEUR SUR CLE USB (Windows):

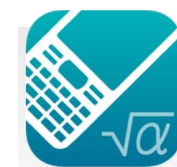
fx-92+ Spéciale Collège, Graph 35+E II et Graph 90+E

→ Offerte pour toute participation à une formation ou offre CASIO

APPLICATIONS CASIO



CASIO EDU+



CASIO CLASSPAD

Site casio-education.fr

Chaîne youtube [CASIO EDUCATION](https://www.youtube.com/CASIOEDUCATION)



RESSOURCES PÉDAGOGIQUES

Retrouvez des manuels et des exercices pour
prendre en main vos outils CASIO



OFFRES ENSEIGNANTS

Bénéficiez d'offres exclusives enseignants sur
les outils pédagogiques CASIO



FORMATIONS ENSEIGNANTS

Inscrivez-vous à des formations
personnalisées réalisées par notre équipe
pédagogique

EN SAVOIR +

ACHAT GROUPÉ PAR VOTRE ÉTABLISSEMENT :



→ Equiper votre établissement de calculatrices pour mise à disposition en classe

→ Centraliser l'achat de vos élèves

Mettez en place un achat groupé, **payé par l'établissement scolaire, et réceptionnez les calculatrices pour distribution en classe.**

CAGNOTTE EN LIGNE POUR VOS ÉLÈVES :

→ Garantie que les familles achèteront le modèle choisi par l'équipe professorale (Mode Examen, Python...)

→ Pas de gestion d'argent pour l'établissement

→ Réception facilitée, par l'établissement.

La cagnotte est établie spécialement pour l'établissement, conformément aux attentes du professeur, **permettant ainsi aux familles d'acheter sur Internet la calculatrice choisie et de la récupérer à la rentrée.**

Si vous souhaitez mettre en place un achat groupé ou une cagnotte pour votre établissement, contactez-nous à education-france@casio.fr
Nous vous mettrons en relation avec nos revendeurs scolaires

Vous utilisez les calculatrices CASIO en classe pour
l'apprentissage des sciences ?

Recevez des calculatrices et émulateurs supplémentaires
pour votre établissement !

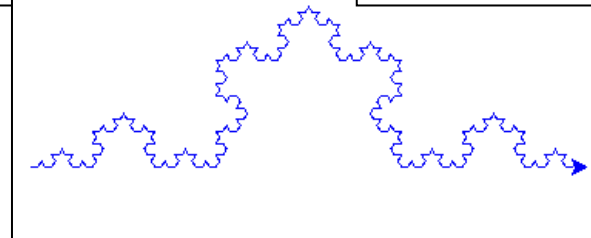
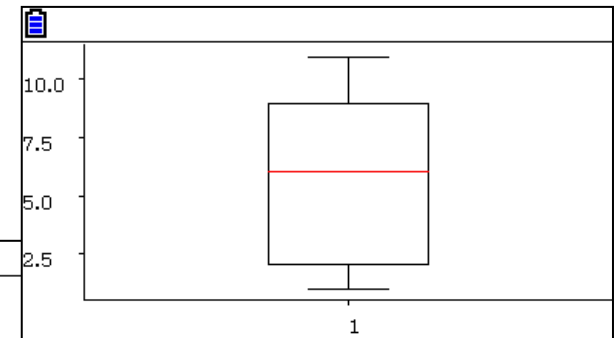
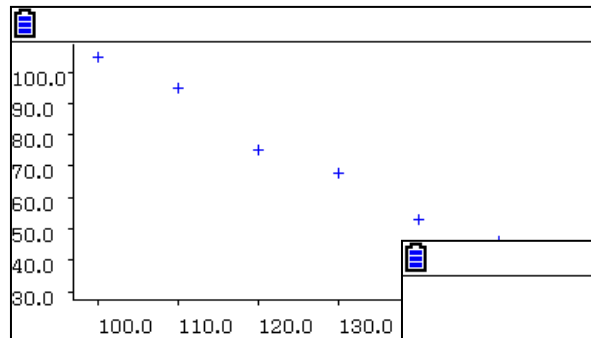
Ouverture de l'offre en mai !

Plus d'informations sur www.casio-education.fr



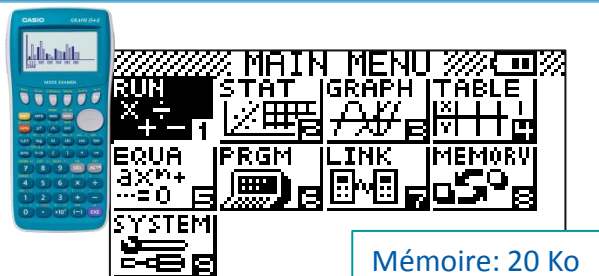
MODE
EXAMEN
INTÉGRÉ

NOUVELLES BIBLIOTHÈQUES GRAPHIQUES
MATPLOTLIB ET TURTLE DISPONIBLES
SUR LA GRAPH 35+E II ET GRAPH 90+E

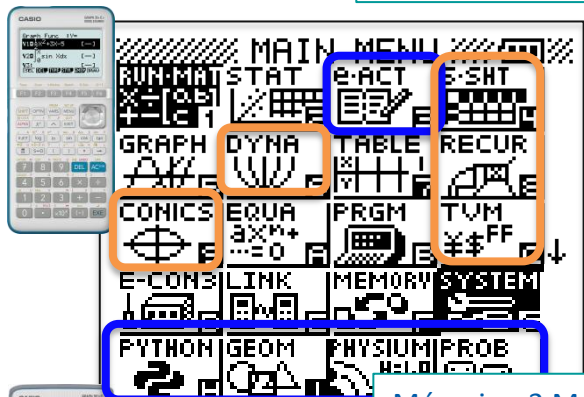


Trois modèles avec le même fonctionnement:

- Menus à icônes
- **CATALOG** avec **SHIFT** **4**
- **SET UP** avec **SHIFT** **MENU**
- Onglets aux mêmes endroits



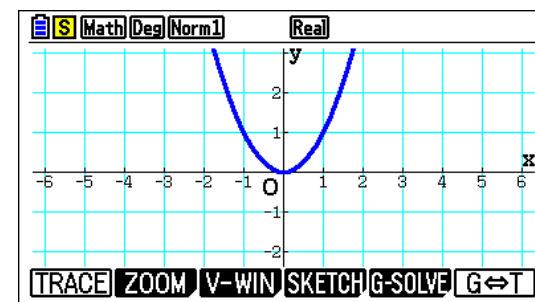
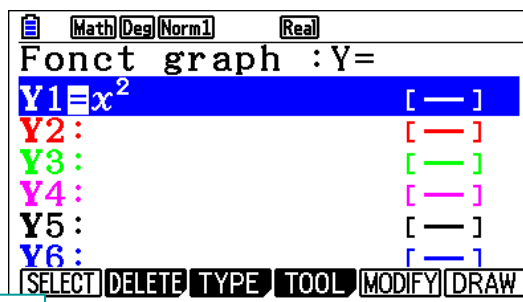
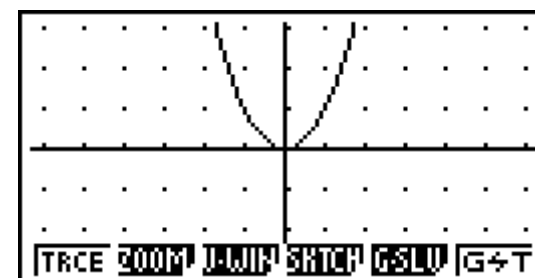
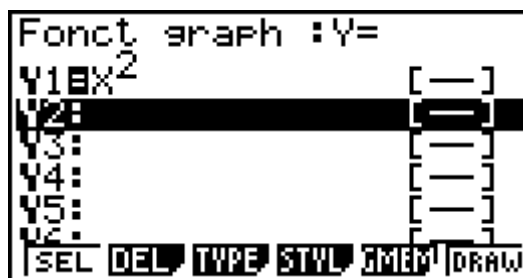
Mémoire: 20 Ko



Mémoire: 3 Mo



Mémoire: 16 Mo





Mise à jour des calculatrices Graph 35+E II et G90

Mise à jour des systèmes d'exploitation des calculatrices G35+E II et G90+E :

- Cliquez sur les liens ci-dessous pour télécharger la mise à jour:
 - [Mise à jour de la Graph 35+E II sous Windows](#)
 - [Mise à jour de la Graph 90+E sous Windows](#)

- Remplissez le formulaire, acceptez les conditions du contrat de licence et validez votre demande.

Télécharger la version d'essai gratuite/Série Subscription

Entrez les informations ci-dessous afin d'obtenir la version d'essai.

Les champs indiqués par un * sont obligatoires.

Profession *	☐ Professeur	[Choisissez dans la liste] ▼
	☐ Étudiant/élève	[Choisissez dans la liste] ▼
	☐ Autres	[Choisissez dans la liste] ▼
Pays*	[Choisissez dans la liste] ▼	

☒ J'ai lu et j'accepte le contrat de licence [Accord de licence du logiciel](#)

Réinitialisation

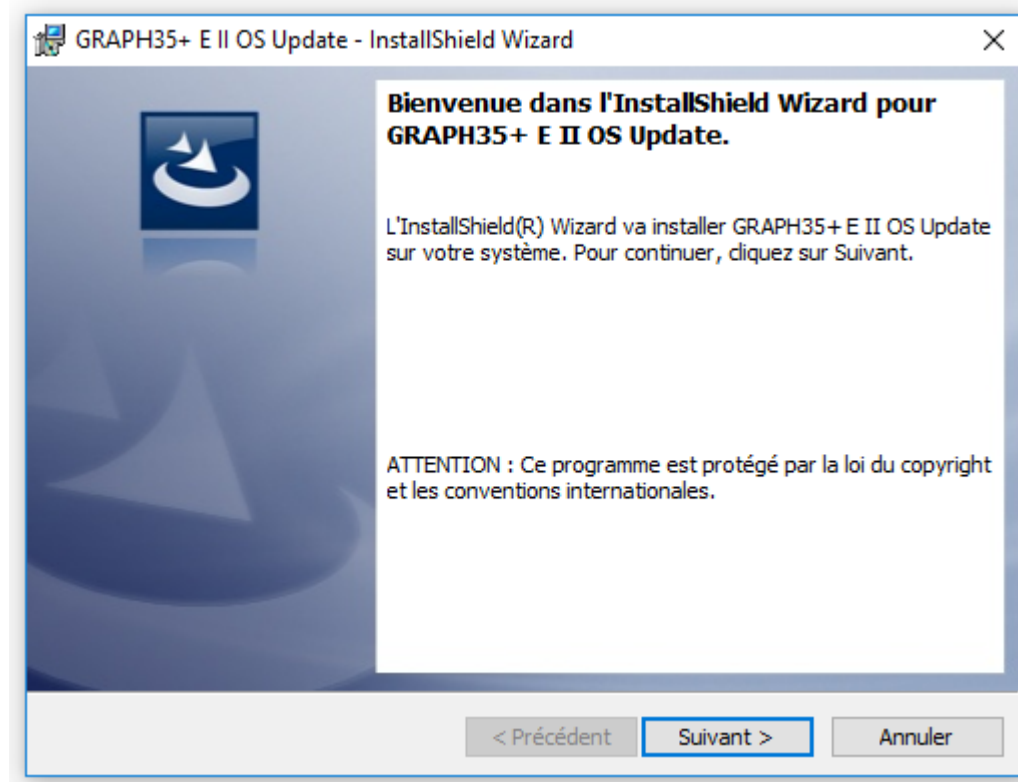
Télécharger

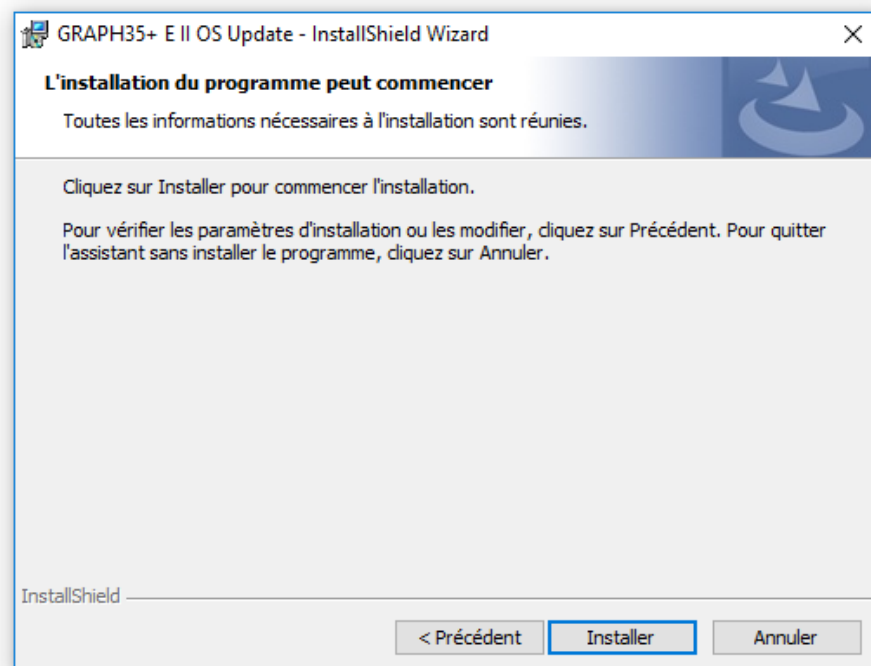
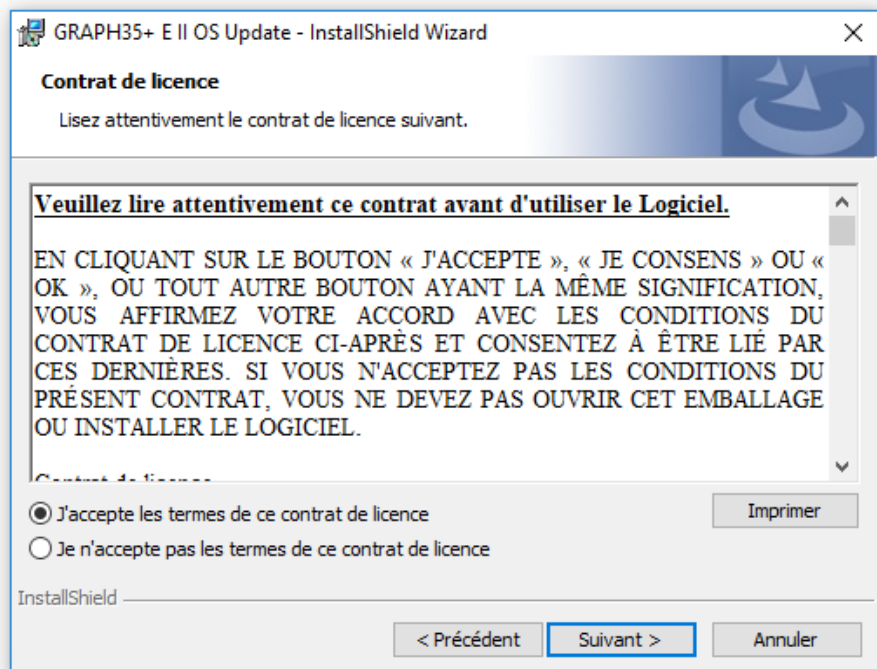
➤ Dès validation du formulaire, le téléchargement du fichier se lance automatiquement

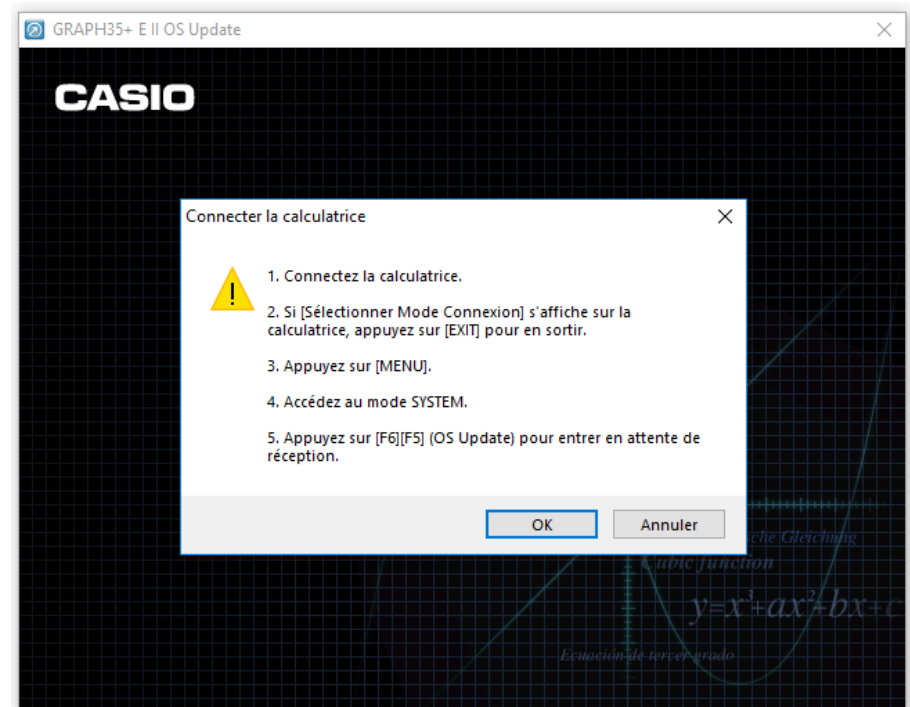
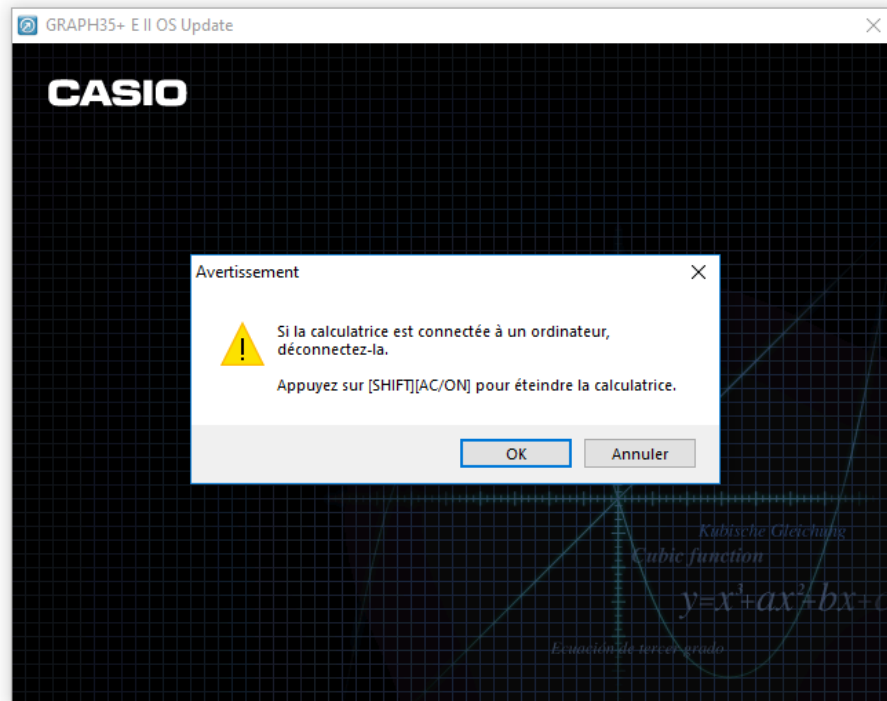
g35+e2_update_330_2b pour la Graph 35+E II

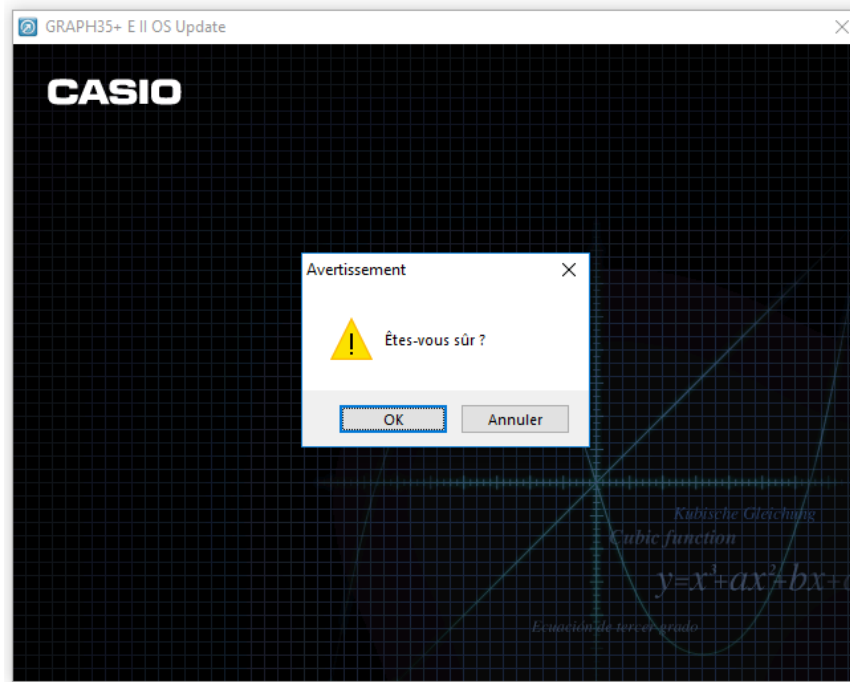
fx-cg50_update_340_2b pour la Graph 90+E

➤ Dézippez le fichier puis double-cliquez sur le programme téléchargé pour lancer l'installation:









- Utilisez le câble USB fourni avec votre calculatrice pour la connecter à votre ordinateur.
- Un écran apparaît sur votre calculatrice. Appuyer alors sur la touche EXIT. Se déplacer dans le menu Système, puis appuyer sur **[F6]** et **[F5]** (OS Update).
- Patientez plusieurs minutes jusqu'à ce que la mise à jour soit installée sur votre calculatrice. Déconnectez ensuite la calculatrice de votre ordinateur afin de finaliser la mise à jour du système d'exploitation.

Ajout des bibliothèques matplotlib et turtle :

- Téléchargez les deux bibliothèques graphiques sur votre ordinateur :
 - Graph 35+E II : [matplotlib](#) et [turtle](#)
 - Graph 90+E : [matplotlib](#) et [turtle](#)
- Reconnectez votre calculatrice à l'ordinateur. Un écran apparaît automatiquement sur votre calculatrice et vous demande de choisir le mode de connexion. Pressez **F1** (Clé USB).
- Ouvrez ensuite le disque amovible de votre calculatrice et copiez-collez les fichiers des bibliothèques à la racine du dossier.
- Déconnecter la calculatrice de votre ordinateur.



Mise à jour des émulateurs Graph 35+E II et G90 sur clé USB

**Mise à jour des systèmes d'exploitation des
émulateurs G35+E II et G90+E :**

- Cliquez sur les liens ci-dessous pour télécharger la mise à jour:
 - [Mise à jour de la Graph 35+E II sous Windows](#)
 - [Mise à jour de la Graph 90+E sous Windows](#)

- Remplissez le formulaire, acceptez les conditions du contrat de licence et validez votre demande.

- Dès validation du formulaire, le téléchargement du fichier se lance automatiquement.

- Connectez la clé USB émulateur à votre ordinateur. Dézippez le fichier téléchargé puis double-cliquez sur le fichier exécutable *GRAPH35+_E_II_Simulator_Ver_USB* ou *GRAPH90+_E_Simulator_Ver_USB*. L'émulateur sur clé USB se met à jour automatiquement.

Ajout des bibliothèques matplotlib et turtle :

- Téléchargez les deux bibliothèques graphiques sur votre ordinateur :
 - Graph 35+E II : matplotlib et turtle
 - Graph 90+E : matplotlib et turtle
- Si ce n'est pas déjà fait, connectez la clé USB émulateur à votre ordinateur.
Si vous n'avez jamais utilisé les émulateurs sur votre clé, lancez les émulateurs de la Graph 35+E II et / ou de la Graph 90+E afin de faire apparaître les différents dossiers.
- Ouvrez le dossier *User_Data*. Sélectionnez la calculatrice pour laquelle vous souhaitez ajouter les bibliothèques graphiques et ouvrez le dossier *FLASH_Folder*. Copiez-collez les fichiers des bibliothèques Matplotlib et Turtle dans le dossier *FLASH_Folder*.
- Eteignez puis relancez l'émulateur sur clé USB afin de faire apparaître les fichiers `matplotl.py` et `turtle.py` dans le menu PYTHON.



PYTHON

Un peu d'histoire ...

Le nom de ce langage ferait référence aux Monty Python. On pourra remarquer le nom de l'environnement de développement IDLE qui fait aussi penser à Eric Idle, membre des Monty Python.



Programmation fonctionnelle

La syntaxe est assez simple, l'indentation délimite la portée des instructions. Ce langage permet beaucoup de possibilités et notamment de travailler de manière fonctionnelle.

```
def terme(n):  
    LIST=[]  
    for i in range(n):  
        LIST=LIST+[2*i-3]  
    return LIST
```

Python sur la Graph 35+E II et la Graph 90+E

Le Menu Python de la Graph 35+E II et de la Graph 90+E est basé sur Micropython (Python 3).

Attention !

Simple d'utilisation, ce nouveau menu permettra de **consolider l'apprentissage fait en salle informatique** par une utilisation au quotidien en salle de classe.

```
Python[ ]
majeur.py : 77
MOYENNE.py : 19

[RUN] [OPEN] [NEW] [SHELL] [DEL] [SRC]
```

L'ÉCRAN D'ACCUEIL

```
==majeur 001/005
def maj(a):
    if a>=18:
        return("MAJEUR")
    else:
        return("MINEUR")

[FILE] [RUN] [SVEL] [CHAP] [A↔a] [D]
```

L'ÉDITEUR

```
TCASIO COMPUTER CO.,
>>>from majeure import
>>>maj(12)
'MINEUR'
>>>maj(21)
'MAJEUR'
>>>
[RUN] [A↔a] [CHAP]
```

LE SHELL (L'interpréteur)

```
Python[
majeur.py : 77
MOYENNE.py : 19

[RUN OPEN NEW SHELL DEL SRC
```

{SHELL}: **[F4]**

```
>>>12*5
60
>>>41%5
1
>>>41//5
8
>>>print("bonjour")
bonjour
>>>
[RUN [A↔] CHAR
```

Le SHELL permet d'exécuter un programme, de tester une instruction ou tout simplement de faire un calcul.

Toutes les commandes et les symboles sont disponibles dans le **CATALOG** (**[SHIFT]** **[4]**).

L'onglet {CHAR} permet d'accéder rapidement à %

Les touches sont traduites en python: **[X]**:* **[÷]**: / **[x²]**: **2


```
>>>for i in range(4):Print(i)
0
1
2
3
>>>
```

Il est aussi possible d'exécuter des instructions plus complexes à condition de les écrire sur une seule ligne.

- Si la valeur initiale n'est pas précisée, la boucle for commence à 0.
- Si l'incrément n'est pas précisée celle-ci est de 1.
- La valeur de fin n'est pas atteinte.

Nous pouvons tester aussi la ligne suivante avec un Copier/Coller:

```
>>>for i in range(1,10,2):Print(i)
1
3
5
7
9
>>>
```

Il est possible de Copier / Coller facilement une ligne et de changer un paramètre:

- Remonter sur la ligne en question
- Appuyer sur **EXE**

MODE ALPHABETIQUE

Graph 90+E

1 minuscule:

Plusieurs minuscules:

1 majuscule:

Plusieurs majuscules:

Graph 35+E II

1 minuscule:

Plusieurs minuscules:

1 majuscule:

Plusieurs majuscules:

```
Catalogue [I]
if
if:
if:else
if:elif
if:and:else
if:or:else
INPUT CAT
```

CATALOG

```
Catalogue
if
if:
if:else
if:elif
if:and:else
if:or:else
INPUT CTGV
```

```
Sélection Caractère
"#$%&'()*+,-./0123
456789:;<=>?@ABCDEF
GHIJKLMNOPQRSTUVWXYZ
Z[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

{CHAR}

```
Sélection Caractère
"#$%&'()*+,-./0123
456789:;<=>?@ABCDEF
GHIJKLMNOPQRSTUVWXYZ
Z[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

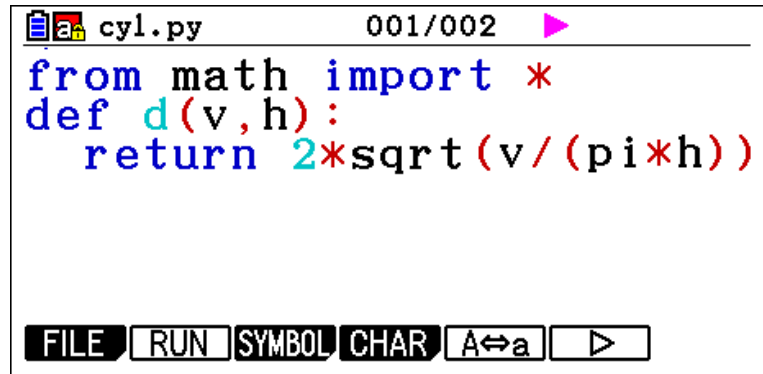
ÉNONCE:

Construire un programme en langage Python permettant de déterminer le diamètre d'un cylindre connaissant sa hauteur et son volume.

```
cyl.py 001/002 ▶  
def d(v,h):  
    return 2*sqrt(v/(pi*h))  
  
FILE RUN SYMBOL CHAR A↔a ▶
```

```
MicroPython v1.9.4  
|CASIO COMPUTER CO.,  
>>>from cyl import *  
>>>d(150,12)  
File "cyl.py", line 2  
NameError: name 'sqrt'  
>>>  
RUN A↔a CHAR
```

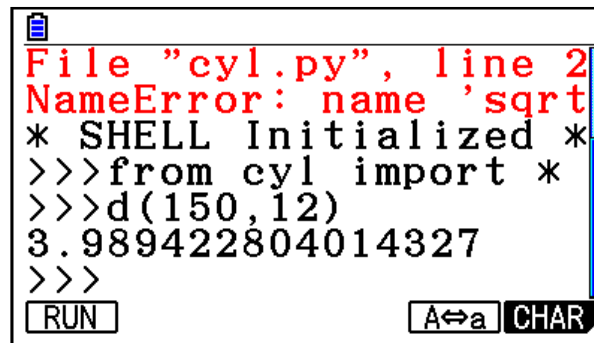
- ✓ Si on teste la fonction dans le SHELL avec $V=150$ et $h=12$, le shell nous indique une erreur: la fonction `sqrt` n'est pas reconnue.
- ✓ Il s'arrête sur la première erreur qu'il rencontre. Le même type d'erreur serait aussi obtenu pour la fonction π .



```
cyl.py 001/002 ▶  
from math import *  
def d(v,h):  
    return 2*sqrt(v/(pi*h))
```

FILE RUN SYMBOL CHAR A↔a ▶

Il faut importer le module math pour utiliser la racine carrée et le nombre pi.



```
File "cyl.py", line 2  
NameError: name 'sqrt'  
* SHELL Initialized *  
>>>from cyl import *  
>>>d(150,12)  
3.989422804014327  
>>>
```

RUN A↔a CHAR

Matplotlib

- ✓ Tracer et visualiser les données sous formes de graphiques
- ✓ Compatibilité ordinateur-calculatrice
- ✓ Calculatrice: `matplotlib`
- ✓ Ordinateur: `matplotlib.pyplot`

Turtle

- ✓ Déplacer une icône à l'écran afin de dessiner au moyen de segments et d'arcs de cercle de couleurs variées
- ✓ Compatibilité ordinateur-calculatrice
- ✓ Calculatrice: `turtle`
- ✓ Ordinateur: `turtle`

Matplotlib

- ✓ **bar** : histogrammes
- ✓ **scatter** : nuages de points
- ✓ **boxplot** : boîtes à moustaches (calculs à l'anglo-saxonne comme sur Matplotlib)
- ✓ **boxplotFR** : boîtes à moustaches (calculs à la française)
- ✓ **arrow** : vecteurs
- ✓ **plot** : courbes
- ✓ **axis**: modification de la fenêtre graphique
- ✓ **text**: ajouter du texte
- ✓ **show**: visualiser le graphique

Turtle

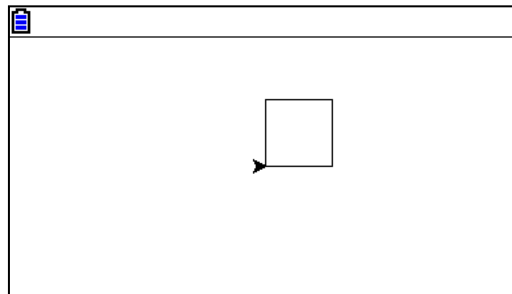
- ✓ **pendown / penup / pencolor / pensize** : paramétrage du crayon
- ✓ **forward / backward / right / left / goto** : déplacements
- ✓ **setheading** : orientation selon un angle donné
- ✓ **circle** : cercles et arcs de cercles
- ✓ **speed**: vitesse du tracé

ENONCE:

A l'aide d'un programme Python, tracer un carré.

```
car.py 001/005 ▶  
from turtle import *  
for i in range(4):  
    forward(50) # avancer de 50  
    left(90)    # tourner a gauche de 90 degrees  
FILE RUN SYMBOL CHAR A↔a ▶
```

✓ Les commandes **forward** et **left** ne sont pas accessibles dans le catalogue.



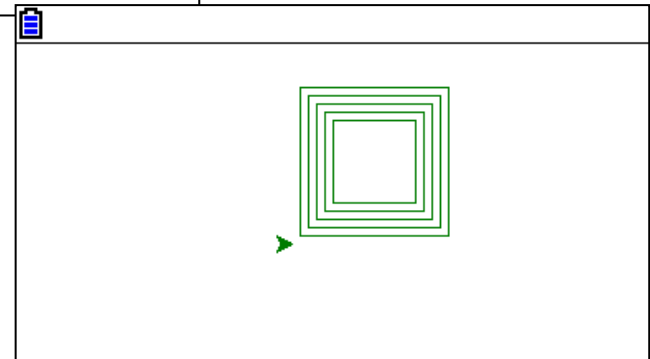
ÉNONCE:

A l'aide d'un programme Python, tracer 5 carrés concentriques.

```
carres5.py 001/016
from turtle import *

speed(0)                # vitesse du trace
k = 10
for j in range(5):
    down()               # abaisser le stylo
    pencolor("green")    # choisir la couleur
    for i in range(4):
        forward(50+j*k)  # avancer de 50+j*k
        left(90)          # tourner a gauche de 90 degres
    up()                 # relever le stylo
    left(90)             # tourner a gauche de 90 degres
    backward(k/2)        # reculer de k/2
    left(90)             # tourner a gauche de 90 degres
    forward(k/2)         # avancer de k/2
    left(180)            # tourner a gauche de 180 degres
```

FILE RUN SYMBOL CHAR A↔a ▶



Une même intensité a été mesurée à l'aide de 20 multimètres identiques.

Intensités mesurées (en mA)	119.1	119.3	119.5	119.6	119.9	120.0	120.1	120.3	120.4
Fréquence d'apparition	1	3	4	3	1	3	1	3	1

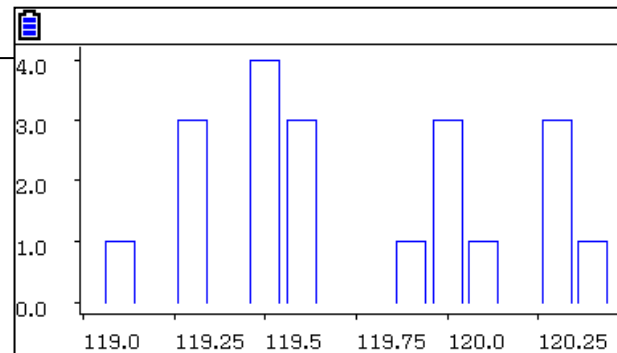
Ecrire, en langage Python, un programme qui permet de tracer un histogramme associé à cette série de mesures.

```
barre.py 001/011 ▶
# bar(x,y,largeur)
# x: position de chaque barre sur l'axe des x
# y: hauteur de chaque barre
# largeur: largeur relative de chaque barre

from matplotlib import *

x=[119.1,119.3,119.5,119.6,119.9,120.0,120.1,120.3,120.4]
y=[1,3,4,3,1,3,1,3,1]
bar(x,y,0.08)
show()
```

FILE RUN SYMBOL CHAR A↔a ▶



ÉNONCE:

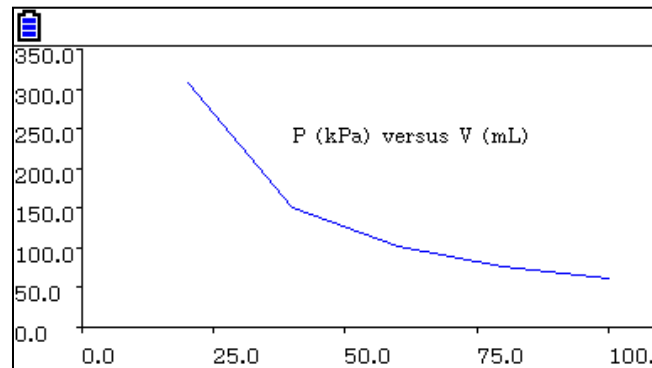
A l'aide d'un programme Python, tracer la courbe représentant la pression (en kPa) en fonction du volume (en mL).

Les pressions et les volumes sont consignés dans le tableau ci-dessous:

Volume (mL)	20	40	60	80	100
Pression (kPa)	307	150	101.7	75.8	61

```
volupres.py 001/008 ▶  
from matplotlib import *  
  
V=[20,40,60,80,100]  
P=[307,150,101.7,75.8,61]  
axis([0,110,0,350])  
plot(V,P,"blue")  
text(40,250,"P (kPa) versus V (mL)")  
show()  
FILE RUN SYMBOL CHAR A↔a ▶
```

- ✓ `axis([xmin,xmax,ymin,ymax])`: modification de la fenêtre graphique
- ✓ `plot(x,y, "couleur en anglais")`: ligne brisée continue
- ✓ `text(x,y, "texte")`: ajouter du texte
- ✓ `show()`: visualier le graphique



ÉNONCE:

Écrire, en langage Python, un programme permettant de calculer le taux d'avancement final d'une transformation modélisée par la réaction d'un acide sur l'eau.

Pour rappel:

$$\text{taux d'avancement} = t = \frac{10^{-pH}}{C}$$

Avec C en mol.L⁻¹

```

taux.py 001/010
def taux(ph,C):
    t=(10**(-ph))/C
    if t==0:
        return("Inerte")
    elif t<1:
        return("Limitee",t)
    elif t==1:
        return("Totale",t)
    else:
        return("Probleme")

```

FILE RUN SYMBOL CHAR A↔a ▶

```

MicroPython v1.9.4
|CASIO COMPUTER CO.,
>>>from taux import *
>>>taux(2.9,0.01)
('Limitee', 0.1258925)
>>>

```

RUN A↔a CHAR



ENONCE:

Écrire, en langage Python, un programme donnant la liste de résultats de n lancers indépendants d'un dé à 6 faces.

```
lancers.py 001/006
from random import *
def des(n):
    liste=[]
    for i in range(n):
        liste=liste+[randint(1,6)]
    return liste
```

FILE RUN SYMBOL CHAR A↔a ▶

Pour pouvoir utiliser la fonction `randint` (disponible dans le catalogue), fonction qui génère aléatoirement un nombre entier, nous devons importer le module `random`.

```
CASIO COMPUTER CO.,
>>>from lancers impor
>>>des(6)
[5, 6, 1, 2, 6, 6]
>>>des(3)
[3, 3, 2]
>>>
```

RUN A↔a CHAR

Les symboles `[]` et `=` sont disponibles dans l'onglet `{CHAR}` (`F4`).

On peut aussi utiliser:

`[]` `SHIFT` `+` `SHIFT` `-`
`=` `SHIFT` `.`

ÉNONCE:

Écrire un programme en Python qui détermine la fréquence d'apparition en pourcentage d'une lettre donnée dans une chaîne de caractères.

```
freq1.py 001/006 ▶
def freq(texte, lettre):
    nbcar=0
    for i in range(len(texte)):
        if texte[i]==lettre:
            nbcar=nbcar+1
    return int(100*nbcar/len(texte))
FILE RUN SYMBOL CHAR A↔a ▶
```

- ✓ `len(texte)` la longueur de la chaîne texte
- ✓ `texte[i]` le caractère en position `i` dans la chaîne de caractères texte sachant que `texte[0]` désigne la première lettre

```
CASIO COMPUTER CO.,
>>>from freq1 import
>>>freq("hello","l")
40
>>>freq("hello","o")
20
>>>
RUN A↔a CHAR
```



ÉNONCE:

Compléter le programme afin d'obtenir le temps nécessaire à un élément de demi-vie T pour atteindre la proportion souhaitée. Un paramètre "e" permettra de spécifier le nombre de chiffres significatifs.

Pour rappel:

$$\lambda = \frac{\ln 2}{T}$$

$$N(t) = N_0 e^{-\lambda t}$$

```

radioact.py 001/011 ▶
# p = N(t)/N0
# N(t): nombre de noyaux a t
# N0: nombre de noyaux a t = 0

from math import *
def radio(T,p,e):
    t=0
    l=log(2)/T
    while exp(-?*?)>?:
        ?+=e
    return ?

```

FILE RUN SYMBOL CHAR A⇌a ▶

✓ En Python, la fonction `ln` s'écrit `log`.

```
radioact.py 001/011 ▶  
# p = N(t)/N0  
# N(t): nombre de noyaux a t  
# N0: nombre de noyaux a t = 0  
  
from math import *  
def radio(T,p,e):  
    t=0  
    l=log(2)/T  
    while exp(-l*t)>p:  
        t+=e  
    return t  
FILE RUN SYMBOL CHAR A↔a ▶
```

```
* SHELL Initialized *  
>>>from radioact impo  
>>>radio(122,0.1,1)  
406  
>>>radio(122,0.2,1)  
284  
>>>  
RUN A↔a CHAR
```



www.casio-education.fr



education-france@casio.fr



[CASIO Education](https://www.youtube.com/CASIO Education)



[CASIO calculatrices](https://www.facebook.com/CASIO calculatrices)



CASIO®

MERCI