

# LANGAGE 1 : Le langage LUA

ISN - Chapitre 4

## TABLE DES MATIÈRES

---

---

<b>I Démarrage avec le langage LUA</b>	<b>2</b>
I 1 Le langage LUA et la TI-Nspire CX CAS . . . . .	2
I 2 Hello World! . . . . .	2
I 3 FOR, IF THEN ELSE, etc . . . . .	4
I 4 Synthèse . . . . .	4
I 5 Retour au mini-projet 01 . . . . .	4
<b>II Quelques exemples de scripts réalisés en Lua</b>	<b>5</b>
II 1 Placer un point mobile . . . . .	5
II 2 Afficher une factorielle . . . . .	5
II 3 Signe d'un polynôme du second degré . . . . .	5
<b>III Projet par binôme : différentes frises multicolores</b>	<b>6</b>
<b>IV Projet par groupe : un jeu simple de Pac Man</b>	<b>6</b>

# I DÉMARRAGE AVEC LE LANGAGE LUA

## I 1 Le langage LUA et la TI-Nspire CX CAS

Depuis l'OS 3.0, Les calculatrices TI-Nspire possèdent une nouvelle fonctionnalité en matière de programmation : le Lua. Ce dernier est un langage de programmation à la fois moderne et rapide. Les scripts écrits en Lua avec un simple éditeur de texte peuvent être convertis en fichiers .tns (format des documents TI-Nspire) depuis le logiciel de la TI-Nspire sur ordinateur. De tels scripts donnent de nouvelles possibilités pour la création de documents TI-Nspire et l'élaboration d'algorithmes graphiques ou de petits programmes (jeux, QCM...).

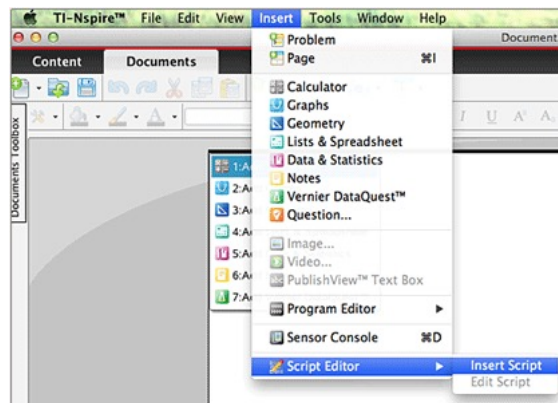
### Liens utiles :

- ▶ Inspired-Lua
- ▶ Lua.org
- ▶ Et bien sûr... Google!

## I 2 Hello World !

Commençons par lancer un script très simple pour se familiariser avec le logiciel de la calculatrice.

- ▶ Ouvrir le logiciel sur l'ordinateur et insérer une nouvelle page :

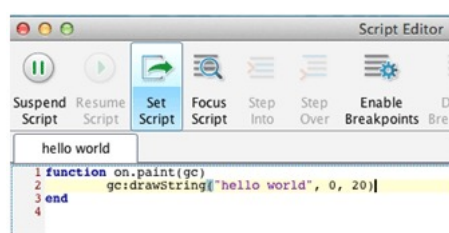


- ▶ Nommer le script :



- ▶ Entrer le code suivant dans la fenêtre de script :

```
function on.paint(gc)
gc :drawString("hello world", 0, 20)
end
```



► Cliquer sur "Définir le script" :



Et voilà votre premier programme en Lua ! Maintenant, nous allons essayer d'améliorer un peu ce programme en changeant la position du texte, la couleur, et en testant d'autres boucles :

### Le tutoriel Inspired-Lua :

Commencer la lecture de la page 1 du tutoriel et répondre aux questions suivantes :

#### 1. Découverte du Lua

- Qu'appelle-t-on un langage sensible à la casse ? Le TI Basic en est-il un ?
- Quel est le rôle de la fonction `on.paint` ?
- Quel est le rôle de l'instruction `platform.window :invalidate()` ?

#### 2. La fenêtre Lua

- Quel est l'intérêt d'utiliser une fenêtre Lua pour afficher un texte dans un programme, par rapport à un programme en TI basic ?
- Comment est défini le système de coordonnées ?
- Expliquer le rôle de `platform.window :height()` et `platform.window :width()`.
- Écrire une synthèse sur la façon dont sont définies les dimensions de l'écran.

#### 3. La mise en forme

- Comment afficher le texte en gras ? En italique ? En gras et italique ?
- Comment changer la couleur du texte ?
- Expliquer le code RGB. En particulier, comment obtenir du blanc ? Du noir ? Du jaune ? Du orange ?

#### 4. Pour conclure

- Reproduire le code ci-dessous, tiré du tutoriel et le tester :

```
function on.paint(gc)
  local h=platform.window:height()
  local w=platform.window:width()
  gc:setFont("sansserif", "b", 12)
  gc:setColorRGB(158, 5, 8)
  local sw = gc:getStringWidth("Ceci est mon texte")
  local sh = gc:getStringHeight("Ceci est mon texte")
  gc:drawString("Ceci est mon texte", w/2 - sw/2, h/2 + sh/2)
end
```

- Expliquer chacune des fonctions utilisées dans ce code.
- Le texte n'est pas centré. Expliquer où est l'erreur dans le code.

### I 3 FOR, IF THEN ELSE, etc

#### La boucle FOR :

Réaliser un script en LUA qui affiche 5 lignes numérotées et centrées à l'écran à l'aide d'une boucle FOR.

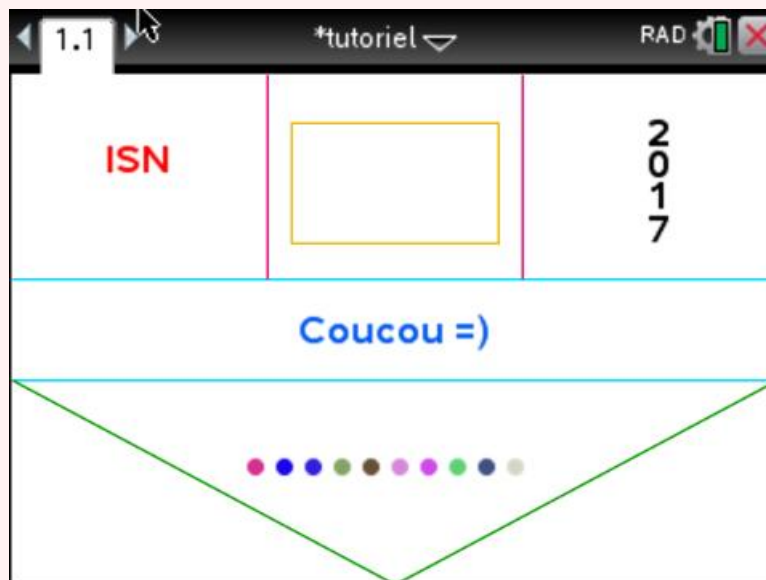
#### IF THEN ELSE et l'aléatoire :

Réaliser un script en LUA qui affiche aléatoirement un prénom parmi les 7 prénoms des élèves d'ISN.

### I 4 Synthèse

#### Travail à effectuer

Reproduire l'écran ci-dessous :



**Attention** : les dix pointillés dans la partie inférieure de l'écran doivent avoir des couleurs aléatoires, qui changent à chaque exécution.

### I 5 Retour au mini-projet 01

Vous disposez désormais de toutes les connaissances nécessaires pour programmer l'algorithme réalisé dans le cadre du mini-projet 01. *A vous de jouer !*

#### Travail à effectuer

Rappel du travail : programmer un algorithme qui permette d'obtenir une vue aérienne de la figure obtenue avec les trous progressivement creusés par le chien.

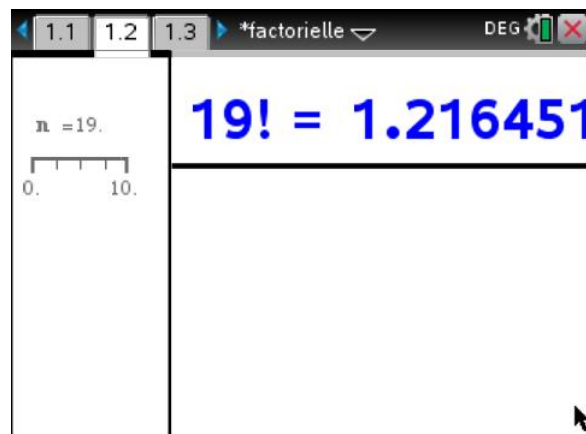
*Indications* : on pourra modéliser la position des prisonniers par un triangle ABC isocèle en B, la position initiale du chien par le point M milieu du segment [AB], et on se limitera aux 10 000 premiers trous creusés par le chien.

## II QUELQUES EXEMPLES DE SCRIPTS RÉALISÉS EN LUA

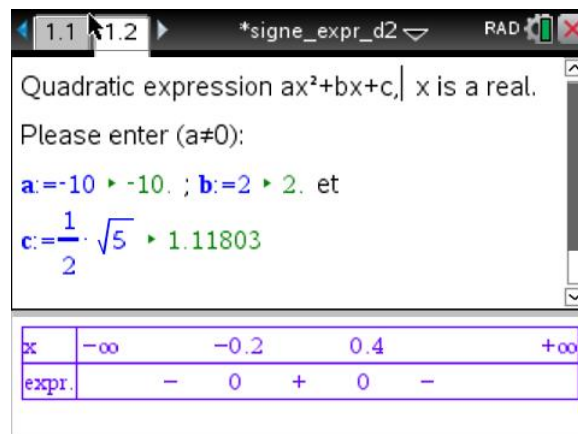
### II 1 Placer un point mobile



### II 2 Afficher une factorielle



### II 3 Signe d'un polynôme du second degré



### III PROJET PAR BINÔME : DIFFÉRENTES FRISES MULTICOLORES

---

Voir le PDF du mini-projet.

### IV PROJET PAR GROUPE : UN JEU SIMPLE DE PAC MAN

---

Repartir du fichier du point mobile et réaliser un jeu (règles au choix).

Quelques idées :

- Le point doit aller du coin en haut à gauche au coin en bas à droite, en évitant certaines zones (murs, surface...).
- On peut personnaliser les couleurs, le fond d'écran, le personnage.
- On peut afficher un message « Game Over » si perdu, ou « Bravo » si le jeu est achevé.
- On peut interagir avec d'autres touches que les flèches. Par exemple, un appui sur la touche **ENTRÉE** peut réinitialiser le jeu ou changer certains paramètres.

Pour les plus ambitieux, on peut pousser le projet jusqu'à la création d'un jeu à la PacMan, où le point mobile doit attraper toutes les pastilles de l'écran pour gagner la partie :

