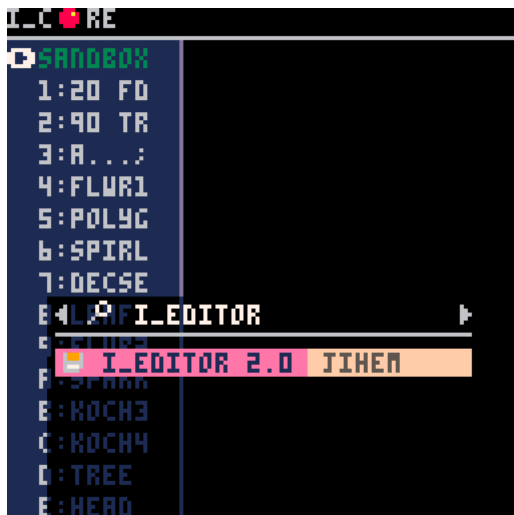


I\_CORE est un jeu de programmation dans lequel vous écrivez des scripts intelligents pour reproduire des modèles de dessin.

I\_EDITOR.P8.PNG

I\_EDITOR est un jeu de programmation. L'objectif est d'écrire des scripts simples pour reproduire un dessin à l'écran. Les commandes utilisées dans les scripts déplacent un pointeur ("tortue"). Le pointeur a un stylo, qui peut être vers le haut ou vers le bas. Lorsque le stylo est vers le bas, chaque mouvement laisse une marque. Il existe des instructions supplémentaires pour calculer, évaluer les conditions et appeler les séquences déclarées précédemment. La syntaxe et la sémantique sont organisées dans un langage appelé I\_CORE. Cette documentation décrit les principales fonctionnalités de ce langage et de l'éditeur pour écrire correctement les scripts.

STARTUP



Si vous n'avez pas la cartouche, la façon la plus simple de la télécharger et de la démarrer est d'utiliser SPLORE, tapez :

> **SPLORE**

Naviguez avec ← et → jusqu'à l'option SEARCH (loupe), sélectionnez [SEARCH], tapez :

**I\_EDITOR**

Sélectionner l'entrée "I\_EDITOR n.m" (où n.m est le numéro de version, ex.: 2.0).

Si vous avez déjà la cartouche ou que vous souhaitez installer la vôtre (à partir d'un téléchargement sur le site web codyssea.com ou lexicaloffle.com), lancez PICO-8 comme d'habitude, copiez la cartouche dans le dossier

des cartouches "carts", puis chargez la cartouche :

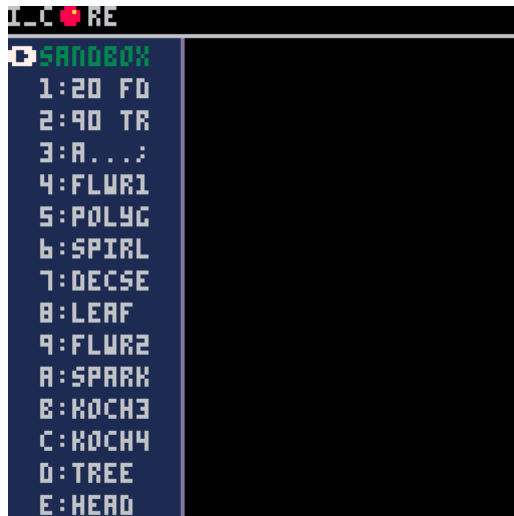
> **LOAD I\_EDITOR.P8.PNG**

et démarrez-la :

> **RUN**

- Le texte en **JAUNE** doit être tapé à l'invite PICO-8.
- Pour ouvrir le dossier "carts" à partir de PICO-8, tapez:  
> **FOLDER**

## MENU PRINCIPAL



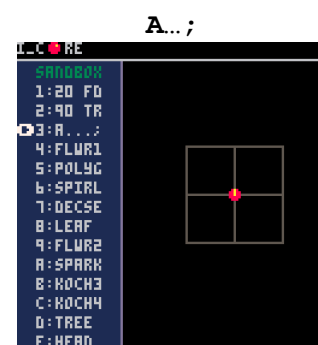
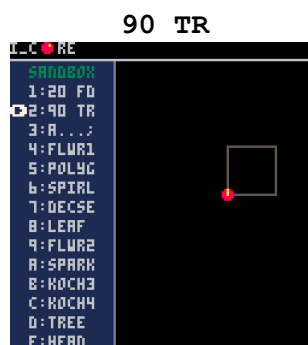
Les options du menu sont listées à gauche. Vous pouvez utiliser **↑** et **↓** pour sélectionner une option. Pour activer l'option sélectionnée, pressez le bouton **X**.

Si vous avez oublié la configuration actuelle de votre clavier, tapez :

> **KEYCONFIG**

La bac à sable (**SANDBOX**) vous permet de faire ce que vous voulez. Il n'y a pas de dessin à reproduire. Vous pouvez utiliser cette option pour apprendre le langage, expérimenter ou concevoir de nouveaux modèles de dessin à reproduire. Toutes les autres options sont des modèles à reproduire (de 1: à E:). Quand vous sélectionnez un modèle, son dessin est affiché pour

vous donner un aperçu de la difficulté. Vous pouvez essayer de reproduire n'importe quel modèle quand vous voulez dans l'ordre que vous le souhaitez. Gardez toutefois à l'esprit que le niveau de difficulté augmente progressivement. Ainsi, si vous n'êtes pas parvenu à reproduire un modèle le suivant risque d'être plus difficile.



## EDITEUR



Une fois que vous avez activé une option du menu principal, l'éditeur s'affiche. Vous pouvez retourner au MENU PRINCIPAL avec le bouton **O**.

L'éditeur dispose de 4 zones (du haut à gauche vers le bas à droite) :

- Titre et statistiques
- Liste des commandes
- Script
- Zone de griffonnage (avec le pointeur "tortue" au milieu)

La ligne jaune sur la tortue rouge indique son orientation : vers le haut de l'écran sur l'image.

## TITRE ET STATISTIQUES



Les statistiques I\_CORE sont affichées dans le coin supérieur droit. Les nombres en violet 0000:0000 indiquent le numéro de la ligne courante et le nombre de lignes du script. Le nombre en orange montre le nombre de pas d'exécution nécessaire à l'accomplissement du script.

Les 3 premiers éléments (en rouge) dans la colonne (2nde) dédiée au script sont les commandes spéciales de l'éditeur :

- <CP Copie à partir du presse-papier
- >CP Copie dans le presse-papier
- CLR Efface le script courant

L'emploi de <CP et >CP est réservé à PICO-8 (voir pico-8.txt) ; sans effet dans l'export HTML.

## TOUCHES DE L'ÉDITEUR

### Liste des commandes

- Passe à la colonne du script
- ↑ ↓ Sélectionne une commande
- X Insert la commande sélectionnée après la ligne courante (marquée avec le curseur) dans le script
- O Aller au MENU PRINCIPAL

Le contenu de l'éditeur est sauvegardé quand vous retournez au menu principal (pour chaque option). Ainsi, vous pouvez naviguer entre le menu et ses différentes options).

### Script

- ← Passe à la colonne de la liste des commandes
- ↑ ↓ Sélectionne une ligne de script ou une commande spéciale de l'éditeur
- X Active le mode Édition sur la ligne courante du script (disponible sur quelques commandes)
- O Supprime la ligne

### Mode Édition

- ← → Incrémente/Décrémente la valeur
- X O Valide (et quitte le mode Édition)

#### Nombres

360  
90  
0

#### Variables

<A  
>A

#### Séquences

:A  
A  
?A

Les nombres peuvent être édités pour changer leurs valeurs (+/-1). Le nom des variables et des séquences peut être modifié de A à C (et de C à A).

#### ZONE DE GRIFFONNAGE

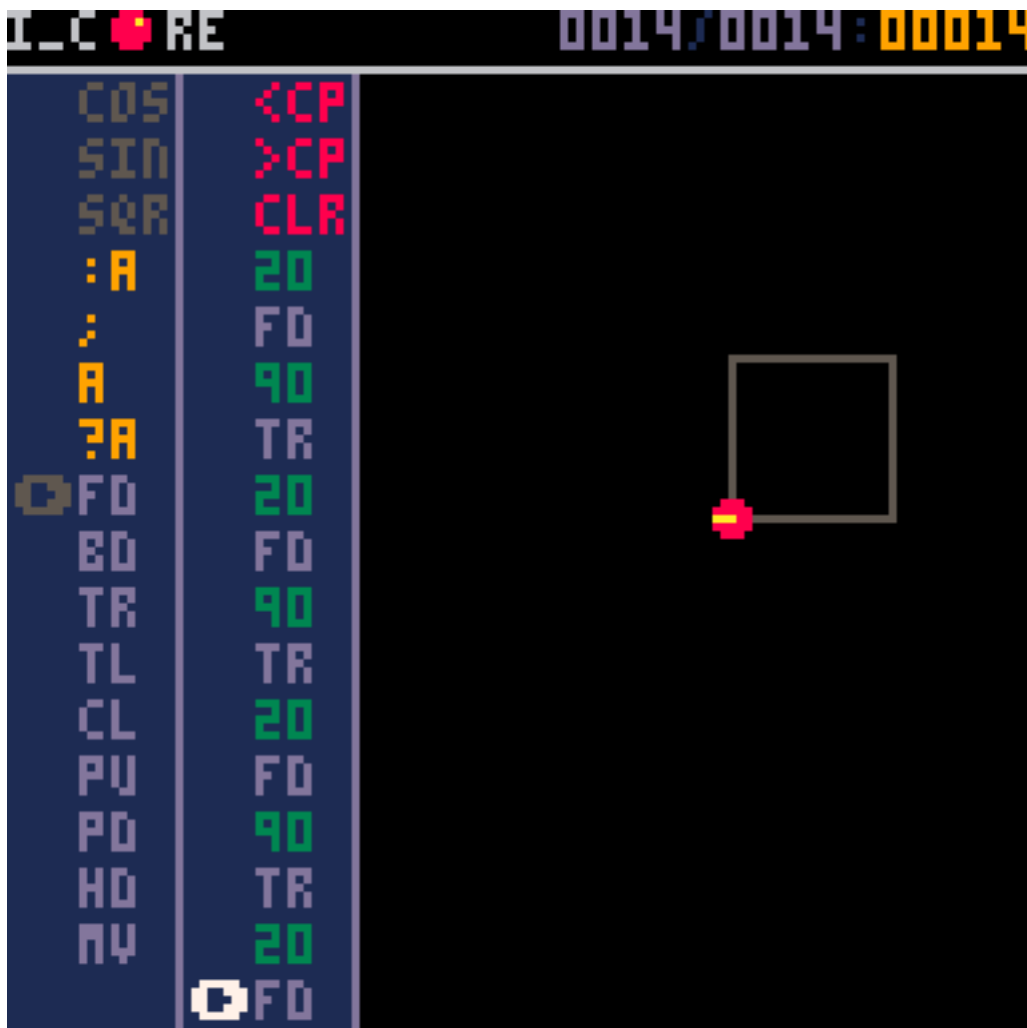
---

Chaque fois que le script est modifié, il est redémarré. Lorsque des commandes de dessin sont exécutées, le résultat est affiché (en temps réel) dans la zone de griffonnage. Ainsi, vous voyez exactement ce que votre script produit.

#### RÉFÉRENCE DU LANGAGE

---

La tortue peut avancer (FD) ou reculer (BD). L'unité de distance unit est le pixel, ex. : **20 FD** avance la tortue de 20 pixels. La tortue peut tourner à gauche (TL) ou à droite (TR) d'un angle exprimé en degré, ex. : **90 TR** fait pivoter la tortue de 90° vers la droite (angle droit).



```

I_C RE 0010/0010:00020
<CP <CP <CP
>CP >CP >CP
CLR CLR CLR
:A :A :A
; 20
?A FD
FD 90
FD TR
BD ;
TR ; A A A
TL ; A A A
CL ; A A A
PU ; A
PD
HD
RV
    
```

Lorsqu'une séquence d'instructions doit être répétée, un nom peut lui être associé (A, B ou C). L'ensemble d'instructions commence avec le préfixe **:A**, **:B** ou **:C**. Le caractère **;** termine l'ensemble. Les lettres **A**, **B** et **C** exécutent la séquence (déclarée avec **:A**, **:B** or **:C**). La taille du script qui dessine un carré peut être diminuée... mais cela augmente le nombre de pas d'exécution (20 ⇔ 14).

```

I_C RE 0004/0004:00004
360 <CP
90 >CP
0 CLR
>A 4
<A 5
+ .
+ - * / H . < > = !
ABS
FLR
    
```

Le langage I\_CORE peut manipuler des nombres et les opérateurs élémentaires **+**, **-**, **\***, **/**. I\_CORE est un petit langage basé sur une pile. Les scripts sont composés de séquences d'instructions, ex.: **4 5 +**. Les deux premiers nombres sont successivement empilés, l'opérateur **+** les retire et évalue le résultat de l'addition qu'il empile. La dernière instruction **.** prend la dernière valeur empilée et l'affiche à l'écran.

```

I_C RE 0009/0010:00062
360 <CP
90 >CP
0 CLR
>A :A
<A 1
+ - H H
+ - * / H . < > = !
ABS
FLR
9 8 7 6 5 4 3 2 1 0
    
```

**?A**, **?B** et **?C** sont des appels conditionnels. Si la dernière valeur empilée n'est pas zéro, la séquence (A, B or C) est exécutée. La pile est de type LIFO (dernière valeur empilée, première dépilée). **#** duplique la dernière valeur empilée (dans la pile).

**10 A** empile 10 dans la pile et appelle A.

**:A** (pile avant → pile après):

- 1 -** soustrait 1 to à la dernière valeur (10 → 9),
- # #** duplique 2 fois la dernière valeur (9 → 9 9 9)
- .** affiche la dernière valeur (9 9 9 → 9 9)
- ?A** appelle A si la der. valeur <> 0 (9 9 → 9) ⇔ 9 A

**>A** place la dernière valeur (de la pile) dans la variable A

**<A** empile le contenu de la variable A dans la pile

Les variables doivent être initialisées (non égales à 0 par défaut).

```
I_C RE 0014/0016:00046
360 <CP
90 >CP
0 CLR
>R :R
<R 1
+ -
- H
* H
/ <R
H +
< >R
> ?R
; S
= >R
! <R
ABS
FLR 15
```

:a 1 - # # <a + >a ?a ; 5 >a <a a <a .

Évalue  $5 + 4 + 3 + 2 + 1 \rightarrow 15$

```
I_C RE 0012/0012:00012
360 <CP
90 >CP
0 CLR
>R 90
<R 0
+ <
- <
* 90
/ 0
H >
< 90
> 0
= 0
! 0
ABS
FLR 0
```

90 0 < .

90 est-il plus petit que 0 ? Non (0)

90 0 > .

90 est-il plus grand que 0 ? Oui (1  $\leftrightarrow$  pas 0)

90 0 = .

90 est-il égal à 0 ? Non (0)

! est l'opérateur *non* : 90 0 = ! .

90 est-il différent (non égal) de 0 ?

Hum... Laissez-moi réfléchir... Oui ?

## MATH

- ABS Valeur absolue, `- 9 ABS .`  $\rightarrow 9$
- FLR Seuil (partie entière), `10 3 / FLR .`  $\rightarrow 3$
- COS Cosinus, `0 COS .`  $\rightarrow 1$
- SIN Sinus, `90 SIN .`  $\rightarrow -1$
- SQR Racine carrée, `9 SQR .`  $\rightarrow 3$

Le cercle trigonométrique est inversé car l'axe des ordonnées (y) croit vers le bas (d'où  $\text{go SIN} = -1$  et non 1).

## TURTLE

---

- CL Change la couleur (0 → 15), **9 CL** J'aime l'orange ;-)  
PD Crayon baissé : la tortue laisse une trace lorsqu'elle se déplace, **PD**  
PU Crayon levé : la tortue ne laisse pas de trace en déplacement, **PU**  
HD Direction : oriente la tortue (0 → 360), **90 HD**  
MV Déplace la tortue à la position (x, y), **0 0 MV**

La surface de griffonnage est une grille de 116x80 pixels : 0 0 coin haut gauche, 79 115 coin bas droit.  
Vous pouvez dessiner en dehors de cette zone mais vous ne verrez qu'une partie du résultat.

Il y a beaucoup de choses à faire avec le langage `L_CORE`. Maintenant, c'est à votre tour d'explorer... En espérant que vous y prendrez du plaisir ;-)

## SCORES

---

20 FD	0002:00002	90 TR	0012:00034	A...;	0020:00148
FLWR <sub>1</sub>	0026:00422	POLYG	0031:00438	SPIRL	0014:00442
DECSE	0026:00849	LEAF	0016:00198	FLWR <sub>2</sub>	0026:01226
SPARK	0031:01069	KOCH <sub>3</sub>	0067:00439	KOCH <sub>4</sub>	0067:01783
TREE	0020:02034	HEAD	0092:04176		

Less is beautiful

- \* -

Tous mes remerciements à Zep pour PICO-8  
et à tous les autres qui ont survécu à mon addiction au code.  
Kind regards

Jihem