**Manuel d’utilisation de l’Interpréteur de Piet**

**Par Hoegy Luc**

**Table des matières**

Table des matières

[Introduction 3](#_Toc194409923)

[Interface 4](#_Toc194409924)

[Launge 4](#_Toc194409925)

[Zone d’introduction 4](#_Toc194409926)

[Zone de compilation des programmes 5](#_Toc194409927)

[Interpréteur Graphique 6](#_Toc194409928)

[Zone 1 – Widgets 6](#_Toc194409929)

[Zone 2 - Programme 8](#_Toc194409930)

[Zone 3 – Stack, Input, Output 8](#_Toc194409931)

# Introduction

Ce projet, né en 2022 dans mon esprit malade, se veut être un IDE pour interpréter le langage PIET, de façon simple, rapide et avec peu de moyens. Tout ce dont vous avez besoin est d’une version plutôt récente de python, et de la librairie Tkinter (manuel d’installation disponible en annexe).

Aucune aide extérieure si ce n’est la documentation en ligne et les IDE préexistants ne m’a été offerte. Toute ressemblance ambigüe avec d’autres projets serait totalement fortuite. Mon objectif avec ce projet est avant tout de me challenger, et si possible, d’initier d’autres programmeurs à ce langage que j’apprécie énormément. Trop de fois j’ai vu les IDE en ligne agir bizarrement à l’exécution d’un programme PIET : par exemple, la fonction d’export de programme dans l’IDE présent à cette adresse :

<https://gabriellesc.github.io/piet/>

ne fonctionne pas (ou du moins pas pour moi). De même, les interfaces que nous pouvons retrouver dans ces projets sont souvent pauvres ou négligées, pour se concentrer surtout sur la partie fonctionnelle. Je ne pourrai pas excuser les propres défauts de mon logiciel, et je ne critique pas les autres projets justes pour les critiquer. Je pose ce regard car je les respecte, et je respecte leurs auteurs, car ils ont eu avant moi l’ambition de permettre l’édition et l’exécution de ce langage, ce que je trouve admirable. Mes critiques se poursuivent plutôt dans les choix que j’ai fait pour implémenter ce logiciel, travail émergeant de longues années de travail, d’abandon et d’échecs parfois.

Je vous invite à me contacter par mail : [hoegyluc@gmail.com](mailto:hoegyluc@gmail.com) afin de me rendre des critiques, des propositions et des questions sur ce projet, afin qu’il prenne l’ampleur que j’espère qu’il pourra avoir un jour.

# Interface

## Launge

Le launge se présente par cette interface :

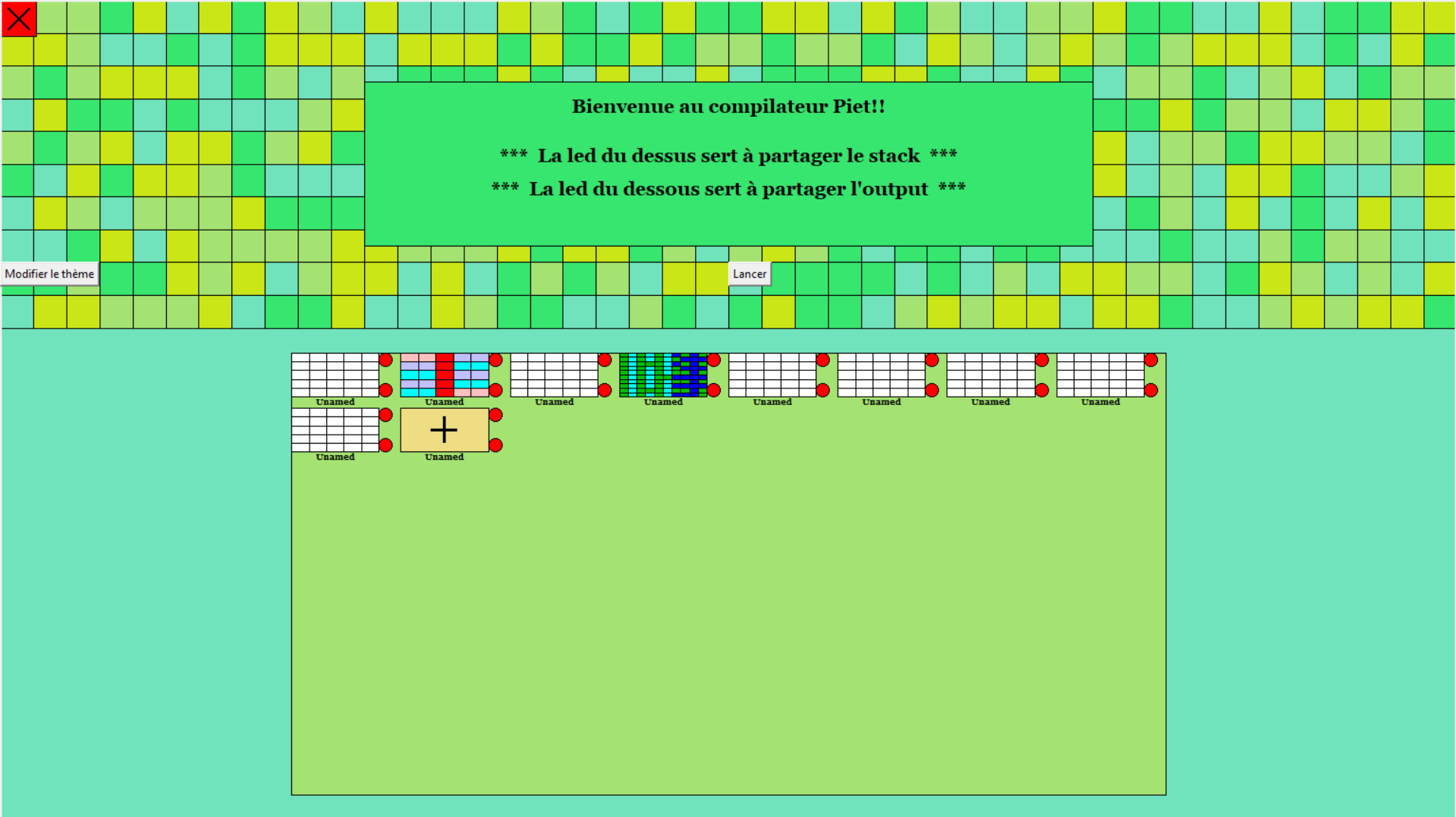


Figure - Interface du Launge de programmes

### Zone d’introduction

La zone en haut de l’écran est un texte brut contenant deux indications :

* L’utilité de la LED du haut (rond rouge haut à côté de chaque programme) ;
* L’utilité de la LED du bas (rond rouge bas à côté de chaque programme).

Une croix en haut à gauche de l’écran est présente, et sert à quitter le programme. La touche *Echap* peut aussi être utilisée.

Un bouton « Modifier le thème » est également présent. C’est un test surtout, permettant de faire osciller l’interface entre 5 thèmes de couleurs allant d’une teinte plus ou moins foncée (par défaut, le thème est le thème 3 sur 5, donc moyennement foncé).

Enfin, un bouton « Launch » est présent, lançant à la suite les programmes présents dans la Zone de Compilation de Programmes (ZCP). Si vous insérez un nouveau programme, il sera enregistré automatiquement, et au moment du lancement, sera interprété en dernier, après interprétation de tous les autres programmes. Par défaut, le programme est un programme blanc basé sur une grille 5x5, qui s’arrête assez vite sans rien effectuer.

### Zone de compilation des programmes

La zone en bas de l’écran est un « compilateur » de programmes PIET. Cela signifie que l’on peut y insérer plusieurs programmes, grâce au bouton ‘+’ sur fond orange. Dès qu’un programme est inséré, une nouvelle zone d’ajout de programme (bouton ‘+’ sur fond orange) est ajoutée à sa suite. Le nombre de programmes maximum possible pour le moment est de 8x8 = 64. Dans de futures mises à jour, il sera possible d’en ajouter autant que souhaité.

Chaque programme est accompagné de deux LED, une au-dessus de l’autre. Elles sont appelées L-Stack (LED de partage de Stack) et L-Output (LED de partage d’output). Si vous connaissez bien le langage PIET, vous savez que c’est un langage de programmation basé sur **Pile**. La langage PIET est évidemment muni aussi d’un **Output**, c’est-à-dire d’une zone dans laquelle nous pouvons afficher du texte. La Pile communique avec l’Output via deux opérations de base proposées par le langage PIET : out(num) et out(char), qui respectivement, dépile (pop) la donnée située au sommet de la pile pour insérer dans l’Output le caractère correspondant sous sa forme numérique ou symbolique (sous forme de caractère référencé dans la table ASCII).

Ainsi, au moment du lancement, si un programme Pn est interprété, il va prendre en entrée la Pile et l’Output du dernier programme à avoir partagé via ses LED (le partage est effectué lorsque la LED est verte). Si aucune Pile ou aucun Output n’est partagé, ils sont mis par défaut à vide.

## Interpréteur Graphique

L’interface de l’interpréteur graphique est la suivante :

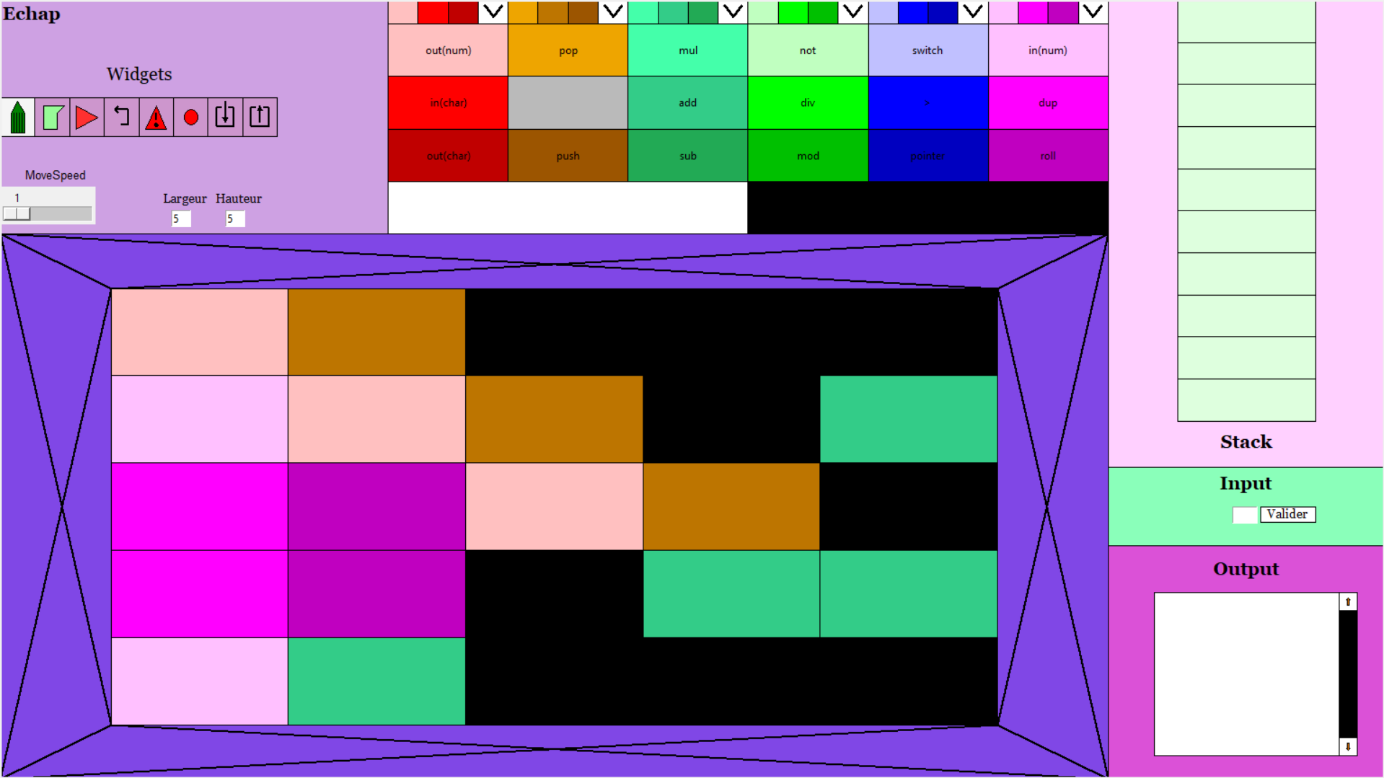


Figure - Interface de l'interpréteur Graphique

### Zone 1 – Widgets

#### Zone à wigets

Dans la version actuelle de l’Interpréteur, il y a 8 widgets disponibles :

1. Crayon – Mode « Single » :

Lors de l’interaction avec une case du programme, la case va changer de couleur en fonction de la couleur sélectionnée dans la zone de Sets (Tableau de couleur à côté des Widgets)

1. Seau – Mode « Cascade » :

Lors de l’interaction avec une case du programme, la case et tous ses voisins de même couleur vont changer de couleur en fonction de la sélection dans la zone de Sets (Tableau de couleur à côté des Widgets). Par « voisins », on entend que les voisins directs de la case cliquée vont changer, mais également les voisins directs de ses voisins directs, etc. Comme pour le seau sur le logiciel Paint par exemple.

1. Start :

Lance l’interprétation du programme, en commençant par le Codel en haut à gauche du programme.

1. Refresh :

Rafraichît la grille de code tout en blanc, et de taille n\*m, selon ce qui est renseigné dans les zones de saisie « Largeur » et « Hauteur ».

1. Stop :

Arrête totalement l’interprétation du programme en cours

1. Pause :

Met en pause l’interprétation du programme en cours. L’interprétation peut être reprise si un clic est effectué à nouveau sur le bouton Pause.

1. Import :

Permet d’importer un programme PIET bien formatté.

1. Export :

Enregistre un fichier sur l’ordinateur, qui est un programme PIET bien formatté, en fonction du programme présent dans la zone 2.

#### Zone des sets

Zone interactive. Si l’on clique sur une case du tableau, elle se met en gris. La couleur sélectionnée pourra ensuite être transmise dans le programme par un simple clic. Les textes affichés sur les autres cases correspondent à l’opération qui sera interprétée à la suite de l’interprétation de la couleur sélectionnée (voir la documentation sur le langage PIET en ligne).

Sur le haut du tableau sont présent des boutons cliquables sous la forme de flèches pointant vers le bas. Cette mise à jour est apparue lors de la version 5.0 de l’Interpréteur PIET. Ces sets permettent de remplacer dans le programme et dans le tableau des Sets certaines couleurs (toujours selon 3 teintes allant du plus clair au plus foncé). Cette mise à jour est purement stylistique et ne change rien au fonctionnement du langage PIET de base.

#### Valeur « MoveSpeed »

Définie la vitesse de lecture au moment de l’interprétation du programme. Entre 1ms et 1000ms entre chaque étape de l’interprétation. Par défaut et sans toucher au Scale, la vitesse de lecture est de 1000ms par étape, soit 1 seconde.

### Zone 2 - Programme

#### Zone du programme

Grille de hauteur et largeur variable (voir bouton refresh, et les zones de saisie « Largeur » et « Hauteur »).

Comme dans le langage PIET de base, le curseur commence en haut à gauche, et poursuit sa lecture selon les contraintes imposées par PIET (voir documentation en ligne). Les cases peuvent être cliquées en mode « Single » ou en mode « Cascade » (widgets n°1 et n°2) pour être modifiées selon la dernière couleur cliquée dans le tableau des Sets (dans la Zone 1). Par défaut la couleur sélectionnée est le Blanc.

### Zone 3 – Stack, Input, Output

#### Zone de Stack (Pile)

Représente une pile de valeur (n’affiche que les 10 premières valeurs du sommet de la pile). Le sommet de la pile est toujours la valeur la plus haute.

#### Zone d’Input (entrée)

Constituée d’une zone de saisie (caractères numériques ou alphanumériques) et d’un bouton valider. Le programme se met en pause (attente passive) lors de l’appel à l’instruction PIET in(char) ou in(num). Cliquer sur le bouton valider en ayant mis une valeur correcte (par exemple, la valeur ‘a’ lors de l’appel à in(num) est incorrect) met fin à la pause et le programme reprendra en mettant au sommet de la pile la valeur rentrée dans la zone de saisie. Attention, la zone de saisie n’est pas vidée entre chaque saisie.

#### Zone d’Output (sortie)

Valorisée lors de l’appel aux instructions PIET out(num) et out(char) (out(char) place la valeur ASCII du sommet de la pile). Ces deux instructions effectuent un pop de la valeur au sommet de la pile.

La valeur out(char) sur la valeur 10 effectue un saut de ligne dans l’Output. Dans la version actuelle, une trentaine de caractères peuvent être placés sur 1 ligne. Les sauts de lignes sont automatiques.

Les flèches « bas » et « haut » à côté de l’Output permettent de se balader entre les lignes déjà poussées dans l’Output.