

TP03 PETITS EXERCICES ENTRE AMIS

Pensez à utiliser GithubDesktop pour faire un « Pull » et récupérer le dossier du TP qui aura été déposé sur Gitlab par le Général Kléber.

Pensez aussi à écrire votre code *en premier lieu* sur feuille pour vous entraîner aux futures épreuves écrites d'informatique.

Partie I

Simple fonction

Écrire une fonction `simple_fonction(x)` qui renvoie la valeur $f(x)$ de la fonction affine f telle que $f(0) = 15$ et $f(10) = 0$.

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

Partie II

Positivité

Construire une procédure booléenne `positif(L)` qui prend en entrée une liste L de nombres réels et qui renvoie **True** si tous les termes de la liste sont positifs ou nuls et **False** dans le cas contraire.

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

Partie III

Fonction conditionnelle

Programmer dans `fonction_conditionnelle(x)` la fonction f définie de la manière suivante :

$$f: \begin{cases} \mathbb{R} \longrightarrow \mathbb{R} \\ x \longmapsto \begin{cases} 1 & \text{si } x < 0 \\ x + 1 & \text{si } x \in [0; 2] \\ -3x & \text{si } x > 2 \end{cases} \end{cases}$$

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

Partie IV

Petits jeux récurifs

1. Écrire une fonction récursive (c'est-à-dire qui s'appelle elle-même et sans aucune boucle explicite!) `compte_a_rebours(n)` qui procède au compte-à-rebours avant le lancement de la dernière fusée de SpaceX à l'aide de l'instruction `print`.

```
>>> n = 3
>>> compte_a_rebours(n)
3
2
1
Décollage !
```

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

On rappelle qu'on peut afficher une ligne de `n` points avec `print("." * n)`.

2. Écrire une fonction récursive (boucle interdite!), appelée `triangle_haut(n)`, qui étant donné `n` produit¹

```
>>> n = 5
>>> triangle_haut(n)
.....
....
...
..
.
```

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

3. De même pour `triangle_bas(n)`

```
>>> n = 5
>>> triangle_bas(n)
.
..
...
....
.....
```

1. Si $n = 0$, il ne faut rien afficher, même pas une ligne vide.

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

4. Un dernier pour la route. Écrire la fonction récursive `regle(n)` telle que

```
>>> regle(1)
-
>>> regle(2)
-
--
-
>>> regle(3)
-
--
-
---
-
--
-
```

Partie V

Chiffres à sommer

1. Proposer une procédure `somme_chiffres(n)` qui renvoie la somme tous les chiffres de l'écriture décimale de l'entier n . Par exemple `somme_chiffre(139)` doit renvoyer 13 car $1 + 3 + 9 = 13$. NB : une procédure récursive est possible mais pas forcément nécessaire.
2. Armé de cette procédure, écrivez-en deux autres : `probleme_16()` et `probleme_20()` qui, sans prendre aucun argument, renvoient les réponses respectives des problèmes 16 et 20 du projet Euler, voir

<http://projecteuler.net/problem=16> et <http://projecteuler.net/problem=20>

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

Partie VI

Quelle heure est-il ?

Un site de voyage permet de calculer les temps de parcours entre deux villes sous forme d'un dictionnaire qui contient 4 clefs ('jours', 'heures', 'minutes' et 'secondes') dont les valeurs associées représentent respectivement les durées en jours, heures, minutes et secondes pour le voyage, le tout de manière unique, c'est-à-dire que 27 heures vaut en fait 1 jour et 3 heures. Néanmoins, si vous faites plusieurs escales, vous voudriez bien connaître la durée totale que vous aurez passée dans les transports. On va décomposer ce problème en plusieurs sous-problèmes plus simples à résoudre.

1. Écrire une fonction `decomposition(duree)` qui prend en argument une durée exprimée en secondes et renvoie un dictionnaire à valeurs entières dont les clefs sont `'jours'`, `'heures'`, `'minutes'` et `'secondes'`

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

2. Écrire la fonction inverse `secondes(dico)` qui prend en argument un dictionnaire du type précédent pour renvoyer la valeur correspondante en secondes.

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

3. En utilisant les deux fonctions précédentes, écrire la fonction `addition(dico1,dico2)` qui va additionner correctement deux dictionnaire `dico1` et `dico2` correspondant à deux voyages successifs et renvoyer le dictionnaire du même type correspondant à la durée totale du voyage.

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

4. Écrire une fonction `affichage(dico)` qui affiche (à l'aide de `print`) le temps total passé en transport de manière un peu plus lisible pour le commun des mortels. Par exemple, l'appel à la fonction `affichage({'jours': 3, 'heures': 22, 'minutes': 10, 'secondes': 54})` doit afficher

Vous allez voyager un total de 3 jours, 22 heures, 10 minutes et 54 secondes

NB : Pour ne pas compliquer les choses, on ne s'occupera pas des « s » en trop dans le cas où l'on devrait écrire « 1 heures » ou « 0 secondes »...

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

5. Enfin, écrire une fonction `temps_total(liste_de_durees)` qui prend en argument une liste (de taille arbitraire) de durées sous la forme des dictionnaires précédents et renvoie le temps total de parcours à l'aide de la fonction `affichage(dico)` précédente.

STOP Gitlab

Allez sur GithubDesktop pour faire un commit. Choisissez (avec pertinence) le résumé. Pensez, si possible, à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.