

Laboratoire # 3

Marchons un peu...

Utilisation de conditions, boucles et autres éléments de contrôle d'exécution

Semaine 3

Ce troisième laboratoire a pour but de vous faire pratiquer les notions éléments de contrôle d'exécution vus dans le module 2. Vous remarquerez que nous avançons un peu plus vite en classe que dans les travaux dirigés : il est important de bien pratiquer tous les éléments vus depuis le début de la session, car ces notions seront réutilisées tout au long de la session.

Nous profiterons également de ce laboratoire pour vous présenter l'outil de débogage de PyCharm. Les outils de débogage sont essentiels pour tout programmeur, les utiliser vous permettra de plus facilement corriger les erreurs dans vos programmes.

1 Calcul d'année bissextile

1.1 Une première solution

Une année bissextile est une année comptant 366 jours au lieu de 365, c'est à dire une année comprenant un 29 février. Selon Wikipédia, voici la règle pour déterminer si une année est bissextile ou non.

1. Si l'année est divisible par 4, passez à l'étape 2. Sinon, passez à l'étape 5.
2. Si l'année est divisible par 100, passez à l'étape 3. Sinon, passez à l'étape 4.
3. Si l'année est divisible par 400, passez à l'étape 4. Sinon, passez à l'étape 5.
4. L'année est une année bissextile (elle a 366 jours).
5. L'année n'est pas une année bissextile (elle a 365 jours).

Sans définir de fonctions, écrivez un programme qui vérifie si une année donnée par l'utilisateur est bissextile ou pas. Après avoir donné la réponse à l'utilisateur, posez la question à nouveau, jusqu'à ce que celui-ci saisisse l'année zéro.

Astuce : utilisez l'opérateur % (modulo) dans vos calculs. Utilisez une boucle `while` pour déterminer si le programme doit s'arrêter ou continuer.

1.2 Décomposition fonctionnelle

Améliorez votre solution précédente en décomposant votre programme en plusieurs fonctions.

2 Validation de nombre premier

Toujours d'après Wikipedia, un nombre premier est un entier naturel qui admet exactement deux diviseurs distincts entiers et positifs (qui sont alors 1 et lui-même). Cette définition exclut 1, qui n'a qu'un seul diviseur entier positif. Elle exclut aussi 0, qui est divisible par tous les entiers positifs.

Sans regarder les notes de cours (essayez de le refaire par vous même), écrivez un programme qui vérifie si un entier entré par l'utilisateur est un nombre premier ou non. Une fois votre programme fonctionnel, prenez quelques minutes pour vous assurer que vous ne faites pas de calculs inutiles.

Indice : Est-ce que le nombre 1000 peut être divisible par 501 ? Et par 502 ? Et par 999 ?

3 Table de multiplication

Écrivez un programme qui calcule et affiche la table de multiplication de 1 à x versus 1 à y, où x et y sont saisis par l'utilisateur. Par exemple, si $x = 10$ et $y = 8$, le programme pourrait afficher :

	1	2	3	4	5	6	7	8

1	1	2	3	4	5	6	7	8
2	2	4	6	8	10	12	14	16
3	3	6	9	12	15	18	21	24
4	4	8	12	16	20	24	28	32
5	5	10	15	20	25	30	35	40
6	6	12	18	24	30	36	42	48
7	7	14	21	28	35	42	49	56
8	8	16	24	32	40	48	56	64
9	9	18	27	36	45	54	63	72
10	10	20	30	40	50	60	70	80

Astuce : utilisez le formatage de chaînes de caractères vu la semaine dernière et deux boucles imbriquées, tel que vu en classe.