

11 sept	TD1 : les premières notions d'algorithmique	Structure d'un algorithme. Notion de variable. Trace.
--------------------	--	--

Les variables :

En informatique, une variable est un emplacement mémoire qui va stocker des données qui peuvent se modifier lorsque le programme s'exécute.

Une variable possède :

- un nom
- une valeur qui n'est pas nécessairement numérique (2019 ; 18,7525 ; vrai ; « Louis » ; [15,36,8,111,846],...)
- un type (entier, réel, booléen chaîne de caractères, tableau, ...)
- une portée (locale ou globale).

Exemple :

```

AlgoIMC

VARIABLES
    m, t, IMC réels
    pren chaîne de caractères
DÉBUT
    Afficher(« Quel est votre prénom ? »)
    Saisir (pren)

    Afficher(« Saisir votre taille en m »)afficher
    Saisir (t)

    Afficher ( « Saisir votre masse en kg »)
    Saisir(m)

    IMC ← m/t2 //affectation, on donne une valeur à IMC
    Afficher (pren + « votre IMC est : » + IMC) //concaténation : mettre bout à bout 2
chaînes de caractères
    t ← t*100
    t ← t *10
    m ← m *1000
FIN

```

Établir la trace de cet algorithme avec Louis qui pèse 70kg et mesure 1,75m.

TRACE :

m	t	IMC	pren
70	1,75		« Louis »
			m
		22,85714286	kg
	175		cm
	1750		mm
70000			g

18 sept	TD2 : prise en main de XCAS	Découverte de XCAS Structure d'un programme. Affectation Saisie Affichage Test
--------------------	------------------------------------	--

Implémenter et tester le programme fait à partir de l'algorithme du TD1.

Demander à l'utilisateur son année de naissance, puis lui indiquer s'il sera ou non majeur en 2019

Tester le programme,

19sept	TD3 : IMC	Structure d'un programme. Affectation Saisie Affichage Tests imbriqués
---------------	------------------	---

Exercice I (source Wikipédia)

Selon la classification de l'OMS : l'indice de masse corporelle est défini par : **Erreur ! Objet incorporé incorrect.**

ex : une personne de 80kg et de 1,85m a un IMC de **Erreur ! Objet incorporé incorrect.**, c'est-à-dire de 23,374, on dira qu'il est de corpulence normale.

IMC ($\text{kg} \cdot \text{m}^{-2}$)	Interprétation
16,5 à 18,5	maigre
18,5 à 25	corpulence normale
25 à 30	surpoids
30 à 35	obésité modérée
35 à 40	obésité sévère
plus de 40	obésité morbide ou massive

Les valeurs de 18 et 25 constituent des repères communément admis pour un IMC normal (donc présentant un rapport de risque acceptable, c'est-à-dire dans la norme statistique)

Compléter le programme sous XCAS de l'ex I afin qu'il affiche aussi l'interprétation de l'IMC.

2 oct	TD4	Boucle pour Concaténation
--------------	------------	----------------------------------

La boucle « pour » s'utilise pour répéter une action quand on sait à l'avance combien de répétitions seront nécessaires

Exercice I

Afficher les nombres impairs de [0;30]

Afficher les multiples de 4 de [0;40]

Afficher les nombres impairs de 2018 à 2100

sans les écrire tous dans le programme.

Exercice II (triangle de A)

Effectuer un programme qui demandera à l'utilisateur de saisir un entier naturel n et qui permettra d'afficher un triangle de n lignes de A .

Exemple :

A

AA

AAA

AAAA si l'utilisateur saisit 4

9 oct	TD5	Boucle pour somme de valeurs
--------------	------------	-------------------------------------

- 1) Créer un programme qui demandera à l'utilisateur de saisir un nombre entier naturel et qui affichera tous ses diviseurs.
- 2) Demander à l'utilisateur de saisir un autre nombre entier naturel et afficher tous ses diviseurs.
- 3) Tester si ces deux entiers sont « **amicaux** », c'est à dire si leur quotient est égal au quotient des sommes de leurs diviseurs.

16 oct	TD6	Boucle pour Irem() debug()
---------------	------------	--

Vérification des programmes faits

Exercice I

- a) Ecrire un algorithme, puis un programme permettant d'afficher les diviseurs positifs d'un nombre entier qui sera saisi par l'utilisateur.
- b) Effectuer et afficher la somme de tous les diviseurs.
- c) Indiquer également le nombre de diviseurs du nombre saisi par l'utilisateur.

Partie A

1) Voici un algorithme :

Mystère

VARIABLES

k entier

a, b, c, tempo : réels

DEBUT

Afficher (« donner une valeur »)

Saisir (a)

Afficher (« donner une valeur »)

Saisir (b)

Afficher(a-b)

tempo ← a

a ← b

b ← tempo

c ← a-b

Afficher(c)

Afficher (« la nouvelle valeur de a est »
+a+ « et la nouvelle valeur de b est »+b)

Pour k de 1 jusque 3 faire

 c ← 2xk

Fin de pour

FIN

a) Etablir la trace de cet algorithme

a	b	c	tempo	k

avec a=9
et b=2.

s de (a-b) et celui de c? Que constatez-vous ?

b)Quels
sont les
affichage

b)

Quel est donc le but de ces trois instructions?

tempo ← a

a ← b

b ← tempo

d) Ecrire ci-dessous un **algorithme** qui demandera à l'utilisateur de saisir deux nombres entiers dans l'ordre croissant ou pas et qui affichera tous les nombres entiers compris entre ces deux nombres.

Exemples :

- si l'utilisateur saisit 5 puis 9, il faudra afficher 5,6,7, 8 et 9
- si l'utilisateur saisit 10 puis 7, il faudra afficher 7,8,9,10.

Partie B

1) Ecrire un **programme** sous XCAS qui demandera à l'utilisateur de saisir deux entiers dans l'ordre croissant ou pas et qui affichera tous les nombres entiers compris entre ces deux nombres.

2) Ecrire un deuxième **programme** sous XCAS qui :

- demandera à l'utilisateur de saisir un nombre entier positif.
- Si l'utilisateur saisit -5 ou 3,2 afficher le message d'erreur « saisie incorrecte » jusqu'au moment où la saisie sera bien un nombre entier positif.

Indication : on pourra utiliser la fonction floor() exemple floor(3,2) renverra 3.

- Affichera les 10 premiers multiples positifs non nuls du nombre saisi

Exemple pour 5 afficher 5,10,15,20,25,30,35,40,45,50

- Effectuera et affichera la somme de ces dix nombres.
- Effectuera et affichera le produit de ces dix nombres.
- Effectuera et affichera la moyenne de ces dix nombres.

Partie A

<p>2) Voici un algorithme :</p> <p>Mystère</p> <p>VARIABLES</p> <p>k entier</p> <p>a, b, c, tempo : réels</p> <p>DEBUT</p> <p>Afficher (« donner une valeur »)</p> <p>Saisir (a)</p> <p>Afficher (« donner une valeur »)</p> <p>Saisir (b)</p> <p>Afficher(a-b) 7</p> <p>tempo ← a</p> <p>a ← b</p> <p>b ← tempo</p> <p>c ← a-b</p> <p>Afficher(c) -7</p> <p>Afficher (« la nouvelle valeur de a est » +a+ « et la nouvelle valeur de b est »+b)</p> <p>Pour k de 1 jusque 3 faire</p> <p style="padding-left: 20px;">c ← 2*k</p> <p>Fin de pour</p> <p>FIN</p>	<p>a) Etablir la trace de cet algorithme</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">a</th> <th style="width: 10%;">b</th> <th style="width: 10%;">c</th> <th style="width: 10%;">tempo</th> <th style="width: 10%;">k</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>9</td> <td>2</td> <td></td> <td>9</td> <td></td> </tr> <tr> <td>2</td> <td>9</td> <td>-7</td> <td></td> <td>1</td> </tr> <tr> <td></td> <td></td> <td>2</td> <td></td> <td>2</td> </tr> <tr> <td></td> <td></td> <td>4</td> <td></td> <td>3</td> </tr> <tr> <td></td> <td></td> <td>6</td> <td></td> <td></td> </tr> </tbody> </table> <p style="text-align: right; margin-right: 20px;">avec a=9 et b=2.</p> <p>b) Les affichages</p> <p>(a-b) et celui de c donnent des valeurs opposées.</p> <p>c) Elles servent à échanger les valeurs de a et de b.</p>	a	b	c	tempo	k	0	0	0	0	0	9	2		9		2	9	-7		1			2		2			4		3			6		
a	b	c	tempo	k																																
0	0	0	0	0																																
9	2		9																																	
2	9	-7		1																																
		2		2																																
		4		3																																
		6																																		

Ecrire ci-dessous un **algorithme** qui demandera à l'utilisateur de saisir deux nombres réels et qui affichera tous les nombres entiers compris entre ces deux nombres.

Exemples :

- si l'utilisateur saisit 5,3 puis 9, il faudra afficher 6, 7, 8 et 9
- si l'utilisateur saisit 10,1 puis 7, il faudra afficher 10, 9, 8 et 7.

<p>algoEncadre</p> <p>VARIABLES</p> <p>k ,deb : entier</p> <p>a, b: réels</p> <p>DEBUT</p> <p>Afficher (« donner une valeur »)</p> <p>Saisir (a)</p> <p>Afficher (« donner une valeur »)</p> <p>Saisir (b)</p> <p>Si a>b</p> <p style="padding-left: 20px;">alors échanger les valeurs de a et de b</p> <p>fin de si ;</p> <p>Si a n'est pas entier</p> <p style="padding-left: 20px;">alors deb ← la partie entière de a +1</p> <p style="padding-left: 20px;">sinon deb ← a</p> <p>fin de si</p> <p>Pour k de deb jusque (partie entière de b) faire</p> <p style="padding-left: 20px;">afficher (k)</p> <p>Fin de pour</p> <p>FIN</p>
--

1) Découverte de la boucle tantque sur multisiteeric

2) Le premier janvier 2018, Paul avait 20€ dans sa tirelire. Le 2 de chaque mois ses parents lui donnent 5€ et Paul dépense ensuite un dixième de la somme dont il dispose le 3 de chaque mois.

a) Afficher les sommes dont il disposait le premier jour des 12 premiers mois de l'année 2018,

b) Si le 1 janvier 2018 est le mois numéroté 1, quel sera le numéro du mois où Paul dépassera pour la première fois 44€

20 nov	TD 10	Boucles pour et tant que
--------	-------	--------------------------

Exercice I

2018 est le code à saisir pour accéder à une application.

Demander à l'utilisateur de saisir trois fois le code, si le code est bon afficher «accès autorisé » et « accès refusé » sinon.

Exercice II

Même contexte, mais on répète la saisie tant que le bon code n'est pas saisi.

Exercice III

Même contexte, mais on répète la saisie tant que le bon code n'est pas saisi, mais pas plus de trois fois.

27 nov	TD 11	Chaînes de caractères et/ou
--------	-------	-----------------------------

Exercice I

1) Créer un programme qui demandera à l'utilisateur de saisir une chaîne de caractères et qui affichera :

Le caractère d'indice 0 est ...

Le caractère d'indice 1 est ...

Le caractère d'indice 2 est ...

Jusqu'au dernier caractère.

Exemple : si l'utilisateur saisit « mur », il faudra afficher :

Le caractère d'indice 0 est m

Le caractère d'indice 1 est u

Le caractère d'indice 2 est r

2)Afficher ensuite le nombre de voyelles de cette chaîne de caractères.

Exercice II

Demander à l'utilisateur de saisir une phrase, puis une lettre.

Indiquer l'indice de cette lettre dans la phrase, ou indiquer l'absence de cette lettre dans la phrase.

11 déc	TD12	Chaînes de caractères
--------	-------------	------------------------------

Créer un algorithme puis un programme qui testera si un mot saisi par l'utilisateur est un palindrome.

Exemple : si l'utilisateur saisit « kayak » on affiche « kayak est un palindrome. » et si l'utilisateur saisit « hiver » on affiche « hiver n'est pas un palindrome. »

18 déc	TD13	Tableaux
--------	-------------	-----------------

Réaliser un programme qui demande à l'utilisateur de saisir l'indice n et qui affichera le tableau contenant 1,2,4,8,16,,,,,2n

Adapter ce programme afin de réaliser la conversion d'un nombre binaire en base 10

8 janv	TD14	Tableaux
--------	-------------	-----------------

Découverte de la variable de type tableau.

- 1) Créer un tableau de 20 nombres entiers tirés aléatoirement entre 0 et 20
- 2) Trouver le minimum de ce tableau
- 3) Trouver le maximum de ce tableau

15 janvier	TD 15	Tri à bulle
------------	--------------	--------------------

- 1) Créer un tableau de 15 nombres entiers tirés aléatoirement entre 0 et 50.
- 2) Afficher ce tableau.
- 3) Placer le plus grand élément au bout du tableau, en utilisant le tri à bulle.
- 4) Trier le tableau initial par ordre croissant, en utilisant le tri à bulle.

28 janvier	TD 16	Complexité et tri par sélection
------------	--------------	--

Complexité

Dans le tri à bulle d'un tableau de 15 valeurs, combien de comparaisons fait-on ?

Dans le tri à bulle d'un tableau de n valeurs, combien de comparaisons fait-on ?

tri par sélection

- 1) Créer un tableau de 15 nombres entiers tirés aléatoirement entre 0 et 50.
- 2) Afficher ce tableau.
- 3) Placer le plus grand élément au bout du tableau, en utilisant le tri par sélection.
- 4) Trier le tableau initial par ordre croissant, en utilisant le tri par sélection.

Complexité

Dans le tri par sélection d'un tableau de 15 valeurs, combien de comparaisons fait-on ?

Dans le tri par sélection d'un tableau de n valeurs, combien de comparaisons fait-on ?

29 janvier	TD 17	Fusion de deux tableaux
------------	--------------	--------------------------------

Soit $T=[5,16,18,20,35,68,81,2019]$ sera initialisé,

Créer un tableau T1 de 8 valeurs classées par ordre croissant, aléatoirement.

Fusionner les tableaux T et T1, pour former un troisième tableau, dont les valeurs seront classées par ordre croissant.

4 février	TD 18	CCF blanc 2
-----------	--------------	--------------------

25 février	TP 19	Codage de César
------------	--------------	------------------------

Exercice I

Saisir un texte en majuscules sans ponctuation. On saisira « MARDI JE SUIS EN FORME »

On souhaite coder ce texte avec un codage de [César de clé 3](#), c'est-à-dire que chaque lettre sera codée par la lettre qui se trouve 3 lettres après dans l'alphabet.

Autrement dit A sera codé en D, B en E, C en F...X en A, Y en B et Z en C.

L'alphabet est alors considéré comme une liste circulaire.

- 1) A la main, coder « MARDI JE SUIS EN FORME ».
- 2) Créer une chaîne de caractères « ABCD.....WXYZABC » que vous utiliserez pour réaliser un programme de codage par la méthode de César avec une clé de 3.
- 3) Adapter le programme afin de permettre le codage du message avec une clé comprise entre 1 et 25 qui sera demandée à l'utilisateur.
- 4) Créer un autre programme permettant le décodage avec une clé donnée.

5 mars	TP 20	Codage de César 2 présents +1TS en immersion+1 étudiante du G2
--------	--------------	--

Correction du TD précédent
Révisions sur les variables.

Amélioration à apporter : traitement des espaces et ' dans une phrase qui devra être codée.

12 mars	TP 21	Codage de César
---------	--------------	------------------------

Correction du TD précédent

Révisions sur les variables.

Amélioration à apporter : traitement des espaces et ' dans une phrase qui devra être codée.

19 mars	TP 22	Code ASCII fonction
---------	--------------	------------------------

Découverte du codage ASCII :

https://fr.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange

1) Créer un programme permettant d'afficher les caractères associés à chaque code ASCII

2)

a) Créer une fonction dont le paramètre sera un mot écrit en minuscules, et qui retournera le mot écrit en majuscules.

Quels soucis peut-on déplorer ? Proposer des améliorations.

b) Créer un programme principal qui demandera à l'utilisateur de saisir un mot écrit en minuscules et qui l'affichera en majuscules.

c) Utiliser le code ASCII pour réaliser un codage de César, avec une clé à choisir.