

# A propos du classeur Excel 'Arith\_Macros.xlt'



Comme on peut le constater d'après l'icône ou d'après l'extension du fichier, il s'agit d'un fichier modèle. Ainsi, lorsque vous cliquez sur l'icône, un classeur vierge -sans titre- apparaît.

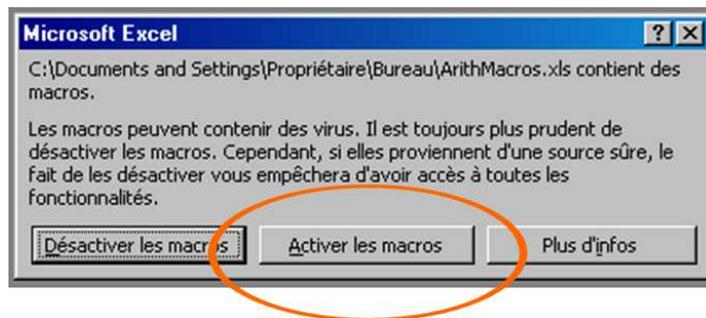
Ce classeur contient des **macros** qu'il vous faut activer.

## Activation des macros du classeur

Cette activation suit des protocoles variables selon la version du programme Excel que vous utilisez. Voici quelques indications pour les cas les plus typiques. Si votre cas n'est pas pris en charge ici, n'oubliez pas de consulter l'aide de votre logiciel, en lançant par exemple une recherche sur l'item 'activer les macros'.

### Excel 97 et équivalent

Si vous travaillez avec une version ancienne d'Excel, vous devriez voir un panneau comme celui-ci :

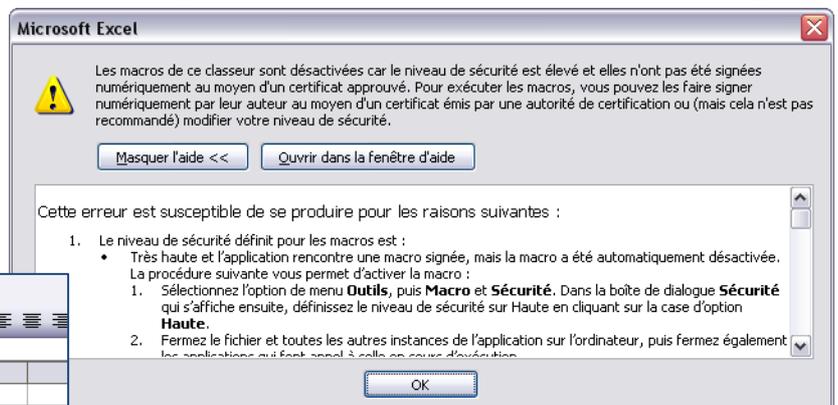


Cliquez sur le bouton [Activer les macros] puis suivez les instructions.

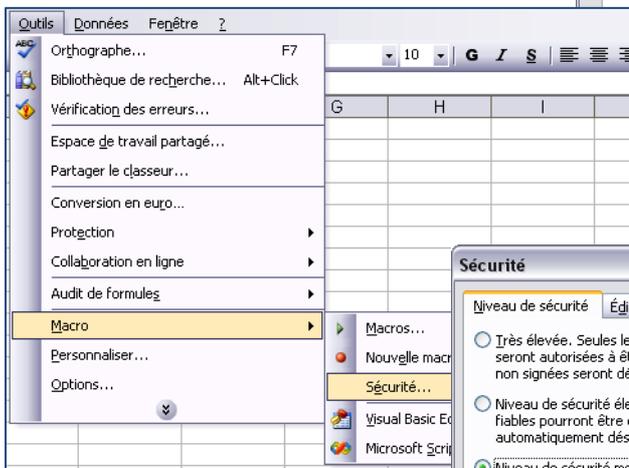
### Excel 2003

Microsoft a introduit un dispositif à double détente, dans le noble but de renforcer la sécurité des postes. Cette démarche est compréhensible dans la mesure où les macros peuvent véhiculer des instructions très dommageables pour votre machine, en clair des virus.

Si un panneau de sécurité comme celui exposé ci-contre à droite apparaît au lancement du classeur Excel, c'est que les réglages de sécurité par défaut sont trop élevés. Vous devez donc les abaisser.



Ce qui suit ne doit être appliqué que si votre poste est équipé d'un anti-virus à jour et correctement configuré. Activez le

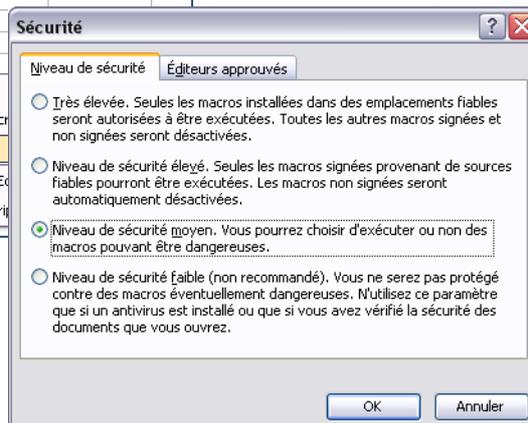


menu [Outils/macro/Sécurité...] comme illustré ci-contre.

Un panneau de réglage apparaît. Optez pour le niveau de sécurité moyen, puis cliquez sur le bouton de validation [Ok].

Lancez, ou relancez, le classeur Excel 'Arith\_Macros.xlt'.

Cette fois-ci, vous devez voir le

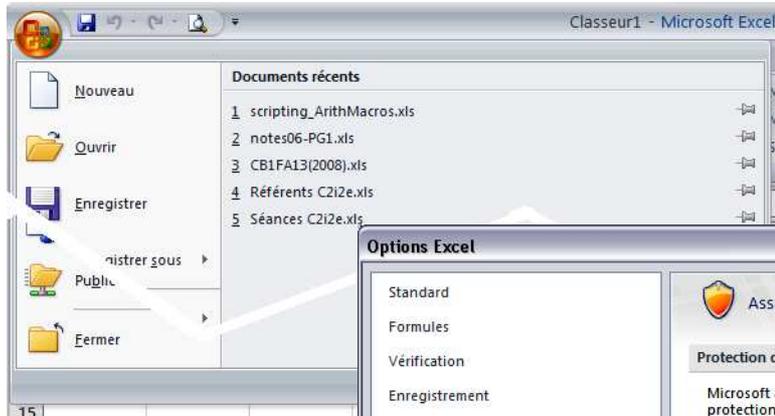


panneau d'alerte standard :

Cliquez sur le bouton [Activez les macros ...] pour pouvoir profiter de mon petit travail.

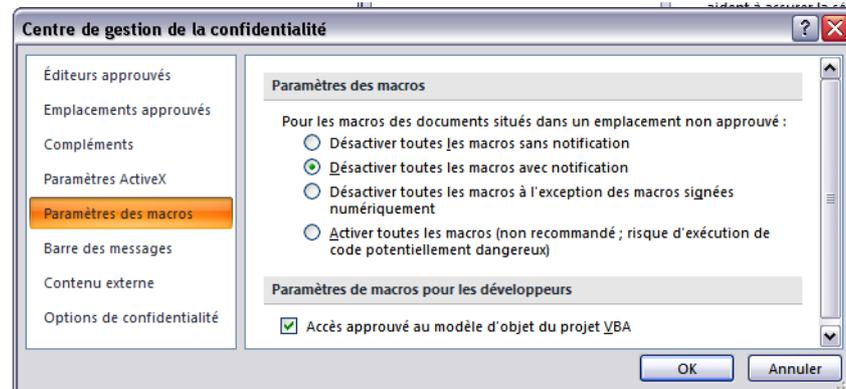
## Excel 2007

Lancez le logiciel puis cliquez sur le bouton [Option Excel] dans le panneau qui se déroule du gros bouton en haut et à gauche de la fenêtre de travail.



Il ne vous reste plus qu'à cliquer sur ce bouton :

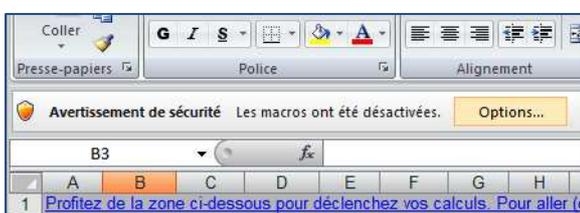
Ouf, le panneau qui nous importe s'affiche. Cliquez sur l'article 'Paramètres des macros', comme ci-dessous, puis sur le bouton 'Désactiver toutes



Vous pouvez maintenant lancer le classeur 'Arith\_macros.xlt', sauf que vous n'êtes pas tout à fait au bout de vos peines. Vous allez rencontrer ce premier message :

En effet, quand j'avais programmé mes petites macros, les fonctions PGCD et PPCM n'étaient pas directement disponibles, en sorte qu'il était plus simple que je les incorpore à mon travail. Je n'ai pas retouché mon travail pour garantir la compatibilité du fichier avec le maximum de versions d'Excel.

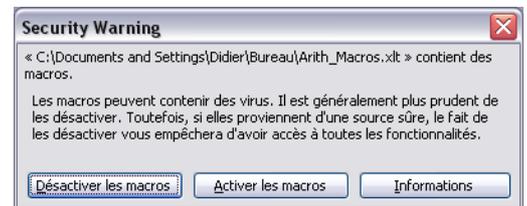
Cliquez sur le bouton [Ok] et cochez la case [Ne plus afficher ...] éventuellement.



Un avertissement de sécurité vous est alors prodigué.

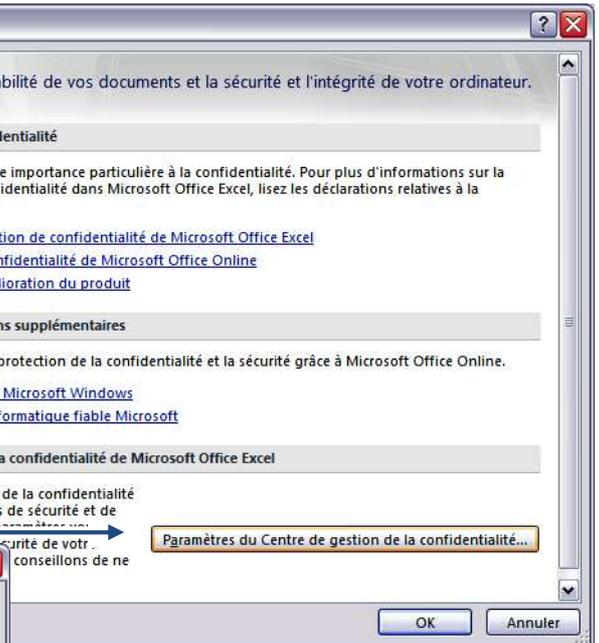
Cliquez sur le bouton

[Options...] juste à droite de l'avertissement. Encore un panneau, rassurez-vous c'est le dernier, dans lequel vous choisissez l'option 'Activer ce contenu' avant de le refermer en cliquant sur [Ok].



Repérez l'article [Centre de gestion de la confidentialité] dans le sommaire à gauche du panneau qui s'ouvre alors. Cliquez dessus.

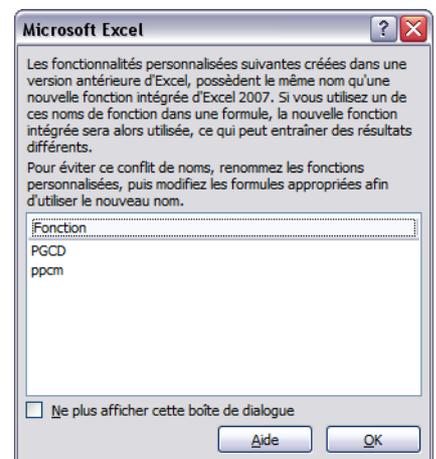
Le panneau évolue comme ici :



les macros avec notification'.

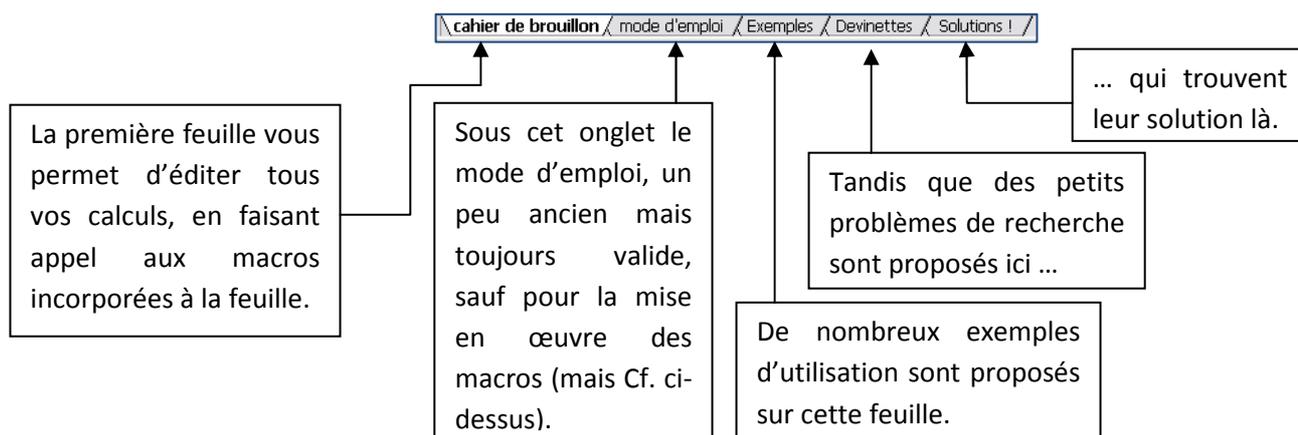
Validez ce choix en cliquant sur le bouton [Ok] en bas du panneau, chaque fois qu'il se présente.

Ces réglages sont fastidieux mais vous n'aurez pas à les refaire par la suite.



## Présentation du classeur 'Arith\_macro.xlt'

Commencez par observer le bas de la fenêtre Excel (ici dans sa version 2003 sous XP) :



Circulez d'une feuille à l'autre, en cliquant sur les onglets, pour compléter votre information. Notez que tous les exemples proposés sont situés dans des cellules que vous pouvez donc copier puis coller ailleurs.

La feuille des devinettes comme celle des solutions appellent un commentaire.

La feuille des devinettes retient 32 petits problèmes. Chacun d'eux reçoit un numéro de type Dnm (par exemple D12). Ces numéros sont disposés en entête de chaque problème ; ce sont en fait des sources de liens hypertexte pointant vers leur solution. Donc, en cliquant sur un numéro de type Dnm, vous affichez la feuille des solutions, au niveau de la solution Snm correspondante (de D12 vous passez à S12). Bien entendu la feuille des solutions adopte le principe inverse : chaque conseil reçoit un numéro de type Spq (par exemple S09) qui permet de revenir au problème correspondant.

Ces différentes situations sont par ailleurs récapitulées dans un document autonome : AR\_exos.pdf .

## Utilisation de la feuille de calcul principale

### Syntaxe générale

Vous entrez dans une cellule une formule du genre `= <nom de la fonction> (<arguments>)`

La fonction attend 1 ou 2 arguments. Dans ce cas, ils doivent être séparés par un point-virgule. Les valeurs passées sont des nombres de tous types, sauf pour la fonction **BaseDec** qui attend une chaîne de caractère.

Lorsque des nombres sont passés en argument, ils sont systématiquement arrondis à l'entier le plus proche. La valeur maximum admissible est a priori 2147483646. Cette borne peut descendre du fait de la mémoire disponible pour les calculs ; c'est le cas par exemple de la fonction **SuitePremiers(n)** qui plafonne aux alentours de 1700.

La valeur 0 n'est acceptée que lorsqu'elle ne déclenche pas d'erreur de calcul.

Tous les modes de référencement sont admissibles :  
=EstPremier(51)  
=EstPremier(C8)  
=EstPremier(le\_nombre) où le\_nombre référence une cellule.

Attention : Il n'est pas possible de passer comme argument un nom désignant une plage de cellules.

En retour la fonction rend soit un booleen, soit une valeur numérique, soit une chaîne alphanumérique. Les différents cas possibles sont précisés ci-dessous.

### Les fonctions autour de la notion de nombre premier

**EstPremier(n)** Rends **VRAI** si n est un nombre premier, **FAUX** sinon.  
n compris strictement entre 0 et 2147483647

**SuitePremiers(n)** Rends sous forme d'une liste plate l'ensemble des nombres premiers compris entre 2 et n. Cette fonction est assez gourmande en mémoire et ne permet pas une valeur d'entrée très grande. Par ailleurs, la chaîne rendue peut excéder très rapidement les 255 caractères

admissibles par une cellule. Avant de lancer l'appel à la fonction, prévoir de fusionner suffisamment de cellules. Un exemple est fourni sur la feuille Exemples. Comparer avec la fonction ci-dessous.

- ListePremiers(De; A)** Rends sous forme d'une liste plate l'ensemble des nombres premiers compris entre 2 et n. Cette fonction reste gourmande en mémoire, mais permet des explorations plus lointaines à condition de garder assez proches la valeur des deux arguments. Notez le gain de cette fonction par rapport à la précédente : **ListePremiers(255000;257000)** retourne une liste de 159 nombres premiers, là où **SuitePremiers(255000)** échoue lamentablement. Même contrainte au niveau de l'accueil par Excel du résultat : penser à fusionner !
- Facteurs(n)** Rends sous forme d'une liste plate l'ensemble des facteurs premiers entrant dans la décomposition du nombre n. Par exemple **Facteur (2004)** rend **2 2 3 167**.
- Décomposition(n)** Rends sous forme d'une chaîne la décomposition en facteurs premiers du nombre n. Par exemple **Décomposition (2004)** rend **2^2 \* 3 \* 167** Malheureusement, la fonction **Eval** des versions précédentes d'Excel ayant disparu, il n'est pas possible de réinjecter le résultat dans un autre calcul. Le moins mal commode consiste à copier la cellule, à lancer un collage spécial (par valeurs) avant d'effectuer les manipulations désirées.

## Autour de la divisibilité

- estmultiple(n1;n2)** Rends VRAI si n2 divise n1 ; n2 ne peut être nul évidemment.
- Diviseurs(n)** Rends sous forme d'une liste plate l'ensemble des diviseurs du nombre n. Ainsi **Diviseurs(2004)** retourne la liste **1 2 3 4 6 12 167 334 501 668 1002 2004** Comme pour les autres fonctions retournant une chaîne alphanumérique, prévoir une place suffisante, en groupant plusieurs cellules au besoin.
- PGCD(n1;n2[;n3;...;n5])** Remplace la fonction fournie par Excel et accessible après appel au menu [Outils/Macros complémentaires] dans les anciennes versions. Dans les versions les plus récentes, ces macros sont implémentées par Microsoft. J'ai gardé mon propre code par souci de compatibilité descendante. Dans cette implémentation les deux premiers arguments **n1** et **n2** sont obligatoires, les 3 autres facultatifs. Tous doivent être positifs.
- PPCM(n1;n2[;n3;...;n5])** Remplace la fonction fournie par Excel et accessible après appel au menu [Outils/Macros complémentaires] dans les anciennes versions. Même remarque que ci-dessus pour les versions les plus récentes d'Excel. Dans cette implémentation les deux premiers arguments **n1** et **n2** sont obligatoires, les 3 autres facultatifs. Tous doivent être positifs.
- Bezout(n1,n2)** Retourne une chaîne alphanumérique de la forme  $u * n1 - v * n2 = d$ , où **u** et **v** sont deux entiers et **d** le pgcd des entiers **n1** et **n2**. Noter que **Bezout(n2;n1)** rend un résultat différent :  
**Bezout (2355 ;825) → 48 \* 2355 - 137 \* 825 = 15**  
**Bezout (825 ;2355) → 20 \* 825 - 7 \* 2355 = 15**

## Changements de base

- DecBase(n;base)** Rends sous forme d'une chaîne alphanumérique l'expression dans la base **base** du nombre **n** fourni. La valeur admissible pour **base** est comprise entre **2** et **36**.
- BaseDec(chaine;base)** Rends sous forme d'un nombre la valeur décimale d'un nombre exprimé dans la base **base** sous forme d'une chaîne **chaine**. Lorsque la base excède **10**, compléter les chiffres par les premières lettres de l'alphabet. Par exemple {**0 1 2 3 4 5 6 7 8 9 A B**} sont utilisés pour la base **12**, {**0 1 ... D E F**} pour la base **16**. La valeur admissible pour **base** est comprise entre **2** et **36** : **BaseDec (« xyz » ;36) → 44027**.