

Plan des sujets abordés en OC Informatique 3^{ème} année 2019 - 2020

Ce plan est souple et peut changer, en fonction de vos demandes et de vos intérêts.

Nous discuterons de sujets à choix

1) **Initiation à la programmation, avec Blockly, une programmation visuelle.**

- Blockly est utilisé dans des universités américaines, pour apprendre à programmer. Deux avantages importants sont l'aspect visuel et facile à comprendre et l'élimination de toute erreur de syntaxe dans la rédaction du code.
- Un autre avantage est de pouvoir voir le code équivalent en Javascript et en Python.

2) **Les logiciels libres**

- L'origine de GNU/Linux, (re)-naissance des logiciels libre.

3) **Initiation à LibreOffice**

- LibreOffice Writer, Draw, Math, Calc, Impress
- LibreOffice Writer vous sera utile pour la rédaction de votre TM.
- LibreOffice Calc peut être utile pour traiter et analyser des expériences. Déjà vu par des élèves d'OS application des mathématiques.

4) **Traitement d'images avec GIMP**

- Différence entre les images bitmap et vectorielles.
- Différences les formats .jpg et .png. Dans quelle situation utiliser un format plus que l'autre.
- Manipulation de bases, particulièrement pour diminuer la taille d'un fichier contenant un image.

5) **Initiation aux pages Web et à l'HTML**

Site Web, Blogs, forums de discussion, wiki (wikipedia), ...

6) **Initiation au CSS, complément du HTML**

7) **Initiation à la programmation avec le langage Javascript**

Rosetta Code, exemples de programmes dans de nombreux langages, contribution

8) **Initiation à la programmation avec le langage Python**

Rosetta Code, exemples de programmes dans de nombreux langages, contribution

9) **Le codage des nombres entiers, des caractères et des couleurs en informatique**

Quelques informations

- Plusieurs dizaines de feuilles seront distribuées → ayez un **classeur** !
- Il y aura des feuilles de cours, comme celle-ci et des feuilles d'exercices nommées "Série ..."
- Page Web avec des informations supplémentaires :

<http://www.juggling.ch/gisin>

ATTENTION d'écrire cette adresse URL en minuscule.

- 1) Allez sous : <https://blockly.games/>
 - i) Résolvez l'activité : "Puzzle", pour vous habituez au fonctionnement.
 - ii) Vous avez 40 minutes pour résoudre l'activité : "Labyrinthe".
C'est le niveau 10 qui vous posera problème.

- 2) Le tour de magie : "L'âge automatique" :
Depuis : <http://www.juggling.ch/gisin/program/blockly/abgex/ex2000/ex2000.html>
Configuration > Divers exemples, au choix... > Choix 3.

Programmez l'algorithme suivant :

- a) Pensez au nombre de soirs que vous voulez sortir par semaine, en moyenne (au cinéma ou théâtre ou au resto ou voir un ami ou autre...).
- b) Placez ce nombre une variable, (z par exemple).

Vous avez ainsi un chiffre en tête.

- Multipliez par 5 votre chiffre
- Additionnez 100
- Multipliez par 20
- Additionnez 19
- Si vous avez déjà eu votre anniversaire cette année, additionnez 1.
- Soustrayez votre année de naissance.

Vous obtenez tous un nombre de 3 chiffres !

- Le chiffre des centaines est le nombre de fois que vous désirez sortir le soir par semaine !
- Les deux autres chiffres indiquent votre âge !

c) Saurez-vous montrer mathématiquement les deux affirmations précédentes ?

d) Affichez ces deux constatations.

- Saurez-vous extraire du nombre de 3 chiffres, le chiffre des centaines ?
- Saurez-vous extraire du nombre de 3 chiffres, les deux premiers chiffres ?

Synthèse de ce que vous avez vu la dernière fois :

- 1) Les **instructions**.
- 2) La notion de **boucle**.
- 3) Les **tests** : **if ... else ...**

Dans le tour de magie : "L'âge automatique"

Dans <http://www.juggling.ch/gisin/program/blockly/abgex/ex2000/ex2000.html>

Configuration > Divers exemples, au choix... > Choix 3.

- 4) La notion de **variable** est essentielle.
- 5) Les **Opérateurs**,
 - i) **Arithmétiques**, + - * ** / % //
- 6) L'affichage d'un résultat. **80% d'un programme sert à son esthétique.**
- 7) Commente extraire le chiffre des centaines d'un nombre entier
- 8) Commente extraire le chiffre des unités ou unité et dizaine d'un nombre entier

Question : comment extraire le chiffre des dizaines d'un nombre entier quelconque ?

Observez que vous pouvez visualiser les codes Javascript, Python et Lua correspondant à l'algorithme que vous avez écrit avec Blockly.

*) Allez sous : <http://www.juggling.ch/gisin/program/blockly/abgex/ex1705/ex1705.html>

Résolvez les 4 étapes.

Techniquement, il n'y a rien de nouveau, c'est même plus simple que ce qui précède, mais il y a quelques subtilités.

Place pour vos commentaires...

*) Si vous avez de l'avance ou si vous êtes curieux.

Allez sous : <http://www.juggling.ch/gisin/program/blockly/abgex/ex1505/ex1505.html>

Résolvez les 7 étapes.

Ceci vous entraînera aux notions suivantes : *à vous de compléter au fur et à mesure*

Voici un autre tour de magie, à programmer.

*) Le tour de magie : "Quel est votre âge ?"

° Pensez à un chiffre entre 1 et 9.

° Multipliez ce chiffre par 9.

° À 10 fois votre âge, soustrayez le résultat obtenu.

-> Le chiffre des unités est le chiffre auquel vous avez pensé au départ.

-> Ajoutez le chiffre des unités au nombre constitué des autres chiffres, cela donnera votre âge.

Exemples :

C = 8, âge = A = 17

Résultat = $10 * 17 - 9 * 8 = 98$

-> Le chiffre des unités vaut bien C

-> Le chiffre des unités plus celui des dizaines vaut $8 + 9 = 17 =$ l'âge.

C = 3, âge = A = 17

Résultat = $10 * 17 - 9 * 3 = 143$

-> Le chiffre des unités vaut bien C

-> Le chiffre des unités plus celui formé par les autres chiffres vaut $3 + 14 = 17 =$ l'âge.

- *) Vous avez déjà abordé le binaire.
- Saurez-vous écrire un algorithme, qui pour un nombre entier positif donné, affiche ce nombre à l'envers.
Exemple : donnée = 1234567 ; résultat affiché = 7654321.
 - Comme l'algorithme précédent, mais au lieu d'afficher le nombre à l'envers, stocke le nombre à l'envers dans une nouvelle variable.
 - Saurez-vous écrire un algorithme, qui pour un nombre donné en binaire, crée un nombre correspondant à la conversion en base 10 du nombre donné.
Exemple : donnée = 1101 ; résultat = 13.
 - Saurez-vous écrire un algorithme, qui pour un nombre donné en base dix, crée un nombre correspondant à la conversion en binaire du nombre donné.
Exemple : donnée = 20 ; résultat = 10100.

Dans <http://www.juggling.ch/gisin/program/blockly/abgex/ex2000/ex2000.html>

Configuration > Divers exemples, au choix... > Choix 3.

Vous pouvez écrire les programmes.

Place pour vos commentaires...

Hashcoding et Blockchain

c.f. https://fr.wikipedia.org/wiki/Fonction_de_hachage

Définition :

Une **fonction de hachage** est une fonction *Hash* qui à un nombre entier positif pouvant être très grand (des millions de chiffres) fait correspondre un nombre entier positif inférieur à un nombre donné.

Ceci n'est que le début de la définition.

Un exemple très simple est :

$$\text{Hash}(n) = n \% 787$$

$n \% 787$ est le reste de la division du nombre n par 787.

Définition complétée, pour son utilisation dans une Blockchain :

Une **fonction de hachage** est une fonction *Hash* qui à un nombre entier positif pouvant être très grand (des millions de chiffres) fait correspondre un nombre entier positif inférieur à un nombre donné.

Elle doit satisfaire en plus les conditions suivantes :

- 1) Elle doit être "facile" à calculer.
- 2) pour un nombre s donné, il doit être très difficile de trouver une valeur n , telle que $\text{Hash}(n) = s$.
(s comme signature)

Autrement dit, calculer *Hash* doit être facile, mais aller dans l'autre sens doit être difficile.

De cette manière, une fonction de hachage peut servir de **signature d'un document numérique**.
 n correspondrait au document numérique.

$s = \text{Hash}(n)$ correspondrait à la signature du document.

- ° La signature " s " est facile à obtenir, car la fonction est "facile" à calculer.
- ° Créer une falsification du document numérique " n_2 " ayant la même signature est difficile, car cela revient à aller dans l'autre sens.

La signature " s " étant donné, il est difficile de trouver n_2 tel que $\text{Hash}(n_2) = s$.

Pour information :

Les exemples standards de fonctions de hachage (hashcoding) utilisé pour signer des documents sont : MD5 ; SHA-1 ; SHA-256 et SHA-512.

(Les deux premières sont déconseillées de nos jours, car moins sûres que les deux suivantes.)

SHA-256 crée une signature de 256 bits, ce qui donne un nombre allant jusqu'à _____

SHA-512 crée une signature de 512 bits, ce qui donne un nombre allant jusqu'à _____

Ces fonctions sont considérées comme facile à calculer.

Regardez leur algorithme de calcul et vous verrez que "facile" n'est pas le terme qui serait utilisé par tout le monde. c.f. <https://fr.wikipedia.org/wiki/SHA-2#SHA-256>

En page suivante, une fonction de hachage plus simple à calculer est donnée. Son défaut est de ne pas être si difficile à inverser pour un professionnel.

Un exemple de fonction de hachage, donné sous forme d'algorithme.

Un **algorithme** est une suite d'instructions.

Soit n le nombre à "hacher".

Suite d'instructions dans un pseudo code :

hash = 0

tant que $n > 0$ faire :

 chiffre = reste de la division de n par 10

 hash = (hash + chiffre) * 384049

 hash = reste de la division de hash par 7538492917

$n =$ arrondi vers le bas de $(n / 10)$

retourner hash ; *donc* hash = Hash (n).

Exercice 1

Allez dans : <http://www.juggling.ch/gisin/program/blockly/abgex/ex2000/ex2000.html>

Configuration > Divers exemples, au choix... > Choix 3

- Programmez une fonction "Hash(n)", selon l'algorithme précédent qui calcule la fonction de hachage du nombre n .
- Tester votre fonction sur : $n = 314159265$ vous devez obtenir Hash (n) = 3062988386
- Comparez votre fonction avec celle d'autres élèves.
- Trouvez un autre nombre n_2 tel que sa fonction de hachage Hash (n_2) = 3062988386

Blockchain

c.f. <https://www.youtube.com/watch?v=SccvFbyDaUI>

c.f. mieux (anglais) <https://www.youtube.com/watch?v=bBC-nXj3Ng4&t=35s>

Supposons avoir un nombre *Big*, qui contient une information. Il peut être très grand.

On cherche un nombre que l'on nommera "*preuve*" tel que le nombre n formé des chiffres de *Big* suivit des chiffres de *preuve* satisfait :

- Hash (n) donne un nombre plus petit que 10'000.
(10'000 pouvant être remplacé par n'importe quelle autre limite.)

La difficulté est de trouver un nombre "*preuve*" pour que Hash (n) soit plus petit qu'une limite donnée. Le travail des **mineurs** est de trouver un nombre "*preuve*".

La **blockchain** est une suite de nombre (très grands) $B_1 ; B_2 ; B_3 ; B_4 ;$ etc. (B pour Bloc)

Chaque nombre (Bloc) B_j est formé en mettant les chiffres des trois nombres suivants à la suite.

- Le Hash du bloc précédent, Hash (B_{j-1})
- Une information convertie sous forme de nombre I_j
- Une "*preuve*" de travail, qui est telle que le Hash de ce bloc B_j soit plus petit qu'une valeur donnée.

Exemple :

$B_1 = 00000000 1234 138$; 0 car c'est le premier bloc ; 1234=information ; 138= preuve t.q. Hash(B_1) = 40734111 < 10^8 .

$B_2 = 040734111 0314 1508$; 407341110=Hash(B_1) ; 3141=information ; 1508=preuve t.q. Hash(B_2) = 6848 < 10^7 .

$B_3 = 000006848 2718 18328$; 6848=Hash(B_2) ; 2718=information ; 18328=preuve t.q. Hash(B_3) = 840645 < 10^6 .

$B_4 = 000840645 9876 21139$; 840645=Hash(B_3) ; 9876=information ; 21139=preuve t.q. Hash(B_4) = 77922 < 10^5 .

$B_5 = 000077922 1414 158167$; 77922=Hash(B_4) ; 1414=information ; 158167=preuve t.q. Hash(B_5) = 5753 < 10^4 .

$B_6 = 000005753 1618 19114405$; 5753=Hash(B_5) ; 1618=information ; 19114405=preuve t.q. Hash(B_6) = 953 < 10^3 .

$B_7 = 000000953 5555 63700198$; 953=Hash(B_6) ; 5555=information ; 63700198=preuve t.q. Hash(B_7) = 17 < 10^2 .

Habituellement, l'information est beaucoup plus longue. Tous les nombres sont plus grands habituellement.

Les calculs de "*preuve*" deviennent de plus en plus longs, car la contrainte de preuve de travail est de plus en plus restrictive, il faut que le Hash soit de plus en plus petit.

Une première page Web

Lorsqu'un navigateur reçoit une page Web, il reçoit en réalité un fichier contenant une série d'instructions qu'il interprète pour afficher la page telle que l'utilisateur la voit.

On nomme "**le client**", celui qui reçoit les pages Web.

Voici le fichier codant une page Web minimaliste :

```
1 <!DOCTYPE HTML>
2 <html lang="fr">
3 <head>
4 <meta charset="utf-8">
5 <title>Première page Web</title>
6 </head>
7 <body>
8 Ceci est ma première page Web.
9 </body>
10 </html>
```

Des numéros de lignes ont été ajoutés pour s'y référer, ils ne font pas partie du fichier.

La première ligne est toujours la même et indique que ce qui suit est de l'HTML 5.

HTML signifie "Hyper Text Markup Language", c'est un langage de description de pages Web.

Tous les textes qui sont entre les symboles < et > sont des **balises**, qui codent la manière d'afficher et de traiter le fichier, pour afficher la page Web.

Lignes :

2 <html> et 10 </html> indiquent le début et la fin du codage.

Si l'on veut spécifier la langue on peut remplacer <html> par <html lang="fr">

3 <head> et 6 </head> codent l'en-tête.

4 indique que le codage des caractères se fait en UTF-8.

Deux autres codages sont le iso-8859-15 et le windows-1252 utilisés sous Windows.

5 <title> ... </title> indique le titre de la fenêtre contenant la page web

6 </head> indique la fin de l'en-tête

7 <body> indique le début du contenu (du corps) de la page Web

8 "Ceci est ..." contient le contenu de la page Web, ici, il se limite à une seule ligne.

9 </body> indique la fin du contenu (du corps) de la page Web

10 </html> indique la fin du codage.

Les lignes 9 et 10 se trouvent toujours en fin de fichier.

Il est possible de taper les lignes qui précèdent dans n'importe quel éditeur de texte, qui permet de sauver les caractères en utilisant le codage utf-8. Sous Linux et Mac, c'est le standard.

Il est possible de taper une page Web avec un éditeur WYSIWYG (What You See Is What You Get), qui cache le code et montre directement le contenu qui sera affiché. Cela a des avantages de simplicité, mais également des désavantages de perte de contrôle de structure de codage.

Voici quelques logiciels d'édition WYSIWYG :

- BlueGriffon. Bon éditeur pour débuter, il a le défaut de modifier le code source HTML d'une page que l'on modifie en WYSIWYG.
- Seamonkey, qui est à la fois un navigateur Web, un gestionnaire d'e-mail et d'adresse et un éditeur de page Web. Il a le même défaut que BlueGriffon. Il génère de l'HTML 4, qui est dépassé.
- Kompozer, était très prometteur, mais n'est plus développé. Ancêtre de BlueGriffon.
- Dreamweaver, est développé par Adobe. Il est propriétaire, payant et assez cher, mais a une très bonne réputation.

Précédemment, une page Web minimaliste a été présentée. Le but de ce qui suit est de mettre en place un début de structure de site Web et d'écrire quelques pages Web, avec des liens, des images et une certaine mise en forme.

Structure de votre site Web

Un site Web est constitué de nombreux fichiers et dossiers.

Une page Web est créée à partir d'un fichier texte dans les langages **HTML** et **CSS**, qui décrivent le **contenu** et le **formatage** de la page Web.

Donc une page Web ne contient pas d'images ou de sons ou de vidéos, mais que du texte. Ce texte indique entre autre où se trouve chaque image.

HTML signifie "Hyper Text Markup Language"

CSS signifie "Cascading Style Sheet" « Feuille de Style en Cascade »

Pour développer et tester un site Web, vous devez respecter une structure. Les fichiers que vous déposerez sur le serveur devront être placés dans le **dossier "/Prenom"** ou dans un de ses sous-dossiers.

Vous reproduirez cette structure sur votre ordinateur. Créez un **dossier "website"**, sur votre ordinateur ou clé USB, qui contiendra l'image de ce qui se trouve sur le serveur.

Ce dossier doit contenir un sous-dossier **"images"** qui contiendra les images. (nom en **minuscules** !)

D'autres sous-dossiers ("sons", "styles", "videos" etc.) viendront plus tard.

Le dossier "website" doit aussi contenir un fichier "index.html".

Les noms de fichiers et dossiers sont constitués de lettres, chiffres et de "-" et de "_".

!! Ne mettez aucun espace ou accent dans ces noms de fichiers !

La page : index.html

Chaque site Web possède une page d'accueil qui se nomme : **"index.html"**. C'est le point d'entrée dans un site Web, qui redirige vers d'autres sections du site.

Vous devrez donc avoir une page **"index.html"** qui aura pour but de diriger le visiteur vers d'autres pages de votre site Web.

Création de votre site Web

- Lancez «BlueGriffon» pour écrire votre première page Web.
 - Sauvez votre première page Web dans « ???\website » et nommez-la « index.html ».
Vous devrez définir le titre de la page, qui est celui qui apparaît en haut de fenêtre.
 - Le fichier « index.html » est celui de la page d'accueil.
 - Écrivez une deuxième page Web, que vous nommerez « page02.html » et que vous stockerez dans le même dossier.
 - Créez un lien qui envoie de « index.html » sur « page02.html ».
 - Créez un lien dans l'autre sens.
 - Testez votre page Web dans Firefox.
 - En cliquant en bas sur « <HTML> Source », vous pouvez voir le code source HTML.
(Hyper Text Markup Language).
 - Dans le dossier "website\images", placez une image que vous voudrez voir dans "page02.html".
 - Dans « page02.html », insérez l'image du point précédent.
Indiquez que le lien doit être relatif.
 - Regardez le code source HTML, pour vous rendre compte que l'image n'a pas été insérée, mais uniquement un lien sur l'image.
En conséquence, les images qui apparaissent dans une page Web, doivent être stockées dans le sous dossier "images" du site Web et pas n'importe où dans l'ordinateur.
 - Sauvez, testez et regardez le code source.
 - Créez une "page03.html". Ayez des liens entre ces pages.
 - Pour s'y retrouver, il est utile d'indiquer en bas de votre site Web, sa structure et la date de la dernière modification de la page. Référez-vous à :
http://www.juggling.ch/zgisin/a2019_oc3/index.html
- Toutes vos pages doivent avoir l'indication de cette structure !

Balises HTML

Les balises HTML sont placées entre des « crochets » « < » et « > ».

Elles donnent des indications sur la manière d'organiser et partiellement d'afficher une page Web. Nous verrons plus loin que c'est par des instructions CSS que la forme de présentation sera codée.

-> Ouvrez votre page web « page02.html », éditez-la et visualisez le code « <HTML> Source ».

La structure minimale est :

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Titre de la page.</title>
</head>
<body>
Texte de la page...<br>
</body>
</html>
```

Le texte en *italique* est arbitraire.

Votre texte diffère de celui qui se trouve ci-dessus. Pour l'instant, ce n'est pas important.

La première ligne informe votre navigateur que le document est de type « html », et quelle est la version utilisée. Par défaut, comme ci-dessus, c'est la dernière version qui est utilisée. Actuellement, en 2020, c'est de l'html 5 qui est utilisé.

<html> indique le début de codage en html. La dernière ligne, </html> indique la fin de la page.

<head> indique le début de l'en-tête, </head> indique la fin de l'en-tête.

<meta ...> donne des informations supplémentaires au navigateur concernant cette page Web.

L'information « charset="utf-8" » indique le type de codage des caractères.

<title> ... </title> indique qu'on donne le titre de la page Web, celui qui est affiché en haut de fenêtre.

<body> indique le début du contenu de la page Web et </body> la fin.

 indique un retour à la ligne.

<p> ... </p> indique un début et une fin de paragraphe.

 ... indique un lien sur une autre page ou dans une autre partie de la page.

 indique que l'image décrite doit être affichée dans la page Web.

Beaucoup d'autres balises existent. Elles sont associées à un style CSS pour la mise en forme.

c.f. : <http://www.w3schools.com/html/default.asp> pour des balises html des attributs.

c.f. : <http://www.w3schools.com/tags/default.asp> pour la liste de toutes les balises de l'html.

Votre site Web

Chacun d'entre vous a un petit site Web visible depuis le monde entier, sur lequel il peut stocker ses diverses créations. Ce site est visible depuis le monde entier !

! Le nom du site Web est : `http://www.juggling.ch/zoc3/votre_prénom`
où `votre_prénom` est à remplacer par votre prénom.

Transfert de votre site Web sur le serveur et tests

Pour transférer des données sur votre site Web, vous avez besoin d'un logiciel et de trois données.

Les trois données sont :

1^{ère} donnée est l'**Hôte**, (ou *nom de domaine*) qui est : `borj.ftp.infomaniak.com`

2^{ème} donnée est l'**Identifiant**, (ou *Utilisateur*) qui est : `borj_zoc3`.

3^{ème} donnée est le **Mot de passe**, que je vous ai donné.

La donnée du Port n'est pas essentielle, on peut la laisser vide.

(Le port par défaut est 21. Il n'y a pas besoin de l'indiquer.)

Tout doit être tapé en **minuscule**.

Le logiciel **Filezilla** qui est installé par défaut sur les ordinateurs du collège permet le transfert.

◦ **À la fin, DÉCONNECTEZ-VOUS, sinon l'accès FTP sera bloqué pour tout le monde !**

Il est aussi possible d'effectuer le transfert depuis le site Web :

<https://manager.infomaniak.com/ftp/>

Traitement d'images avec GIMP

Le but de ce qui suit est de donner quelques informations de bases concernant les images en informatique et de s'initier au traitement d'images avec le logiciel GIMP.

En informatique, on distingue deux grandes classe d'images.

Les images **vectérielles** et les images **bitmap**.

Dans la suite, nous nous intéresserons exclusivement aux images **bitmap**.

Dans une page Web, de même que dans un document Libreoffice Writer, on insère souvent des images. Leur taille est souvent trop grande et la qualité inutilement bonne. Ceci a pour désavantage que la taille du fichier contenant l'image est grande. Le chargement d'une page Web contenant une telle image sera donc plus lent.

Quelques formats de fichiers stockant des images :

- .bmp** est un format standard sous Windows, qui est ancien et n'a plus beaucoup d'intérêt.
- .gif** (Graphics Interchange Format) est un format qui est beaucoup utilisé pour les images sur le Web. Il ne permet d'avoir que 256 nuances de couleurs, mais elles peuvent être choisie parmi les $16'777'216 (=256^3)$ nuances usuelles de couleurs en informatique.
- .png** (Portable Network Graphics) est similaire au format **.gif** et le remplace de plus en plus souvent, car il provient du monde du logiciel libre, il permet d'avoir plus de nuances de couleurs et les fichiers qu'il génère sont plus petits.
- .jpg** ou **.jpeg** (Joint Photographic Experts Group) est le format utilisé pour les photos.

Avantage et désavantage de divers formats :

Le format **.jpg** a l'avantage de compresser l'information contenue dans une image, ce qui rend le fichier correspondant plus petit. Le désavantage est que la compression se fait avec une perte de qualité d'image. L'utilisateur a le choix de la qualité qu'il désire conserver, en sachant que « meilleure qualité » signifie aussi « plus grand fichier ».

Ce format s'utilise pour stocker des photos. Il n'est pas adéquat pour stocker du texte, ni des dessins contenant de grandes zones de couleurs uniforme, comme certains logos.

Le format **.png** comprime aussi l'information, mais sans perte de données. Pour une même photo, celle stockée au format **.png** utilisera un fichier de plus grande taille que celle stockée au format **.jpg**.

Ce format s'utilise pour stocker des images contenant du texte et/ou des dessins contenant de grandes zones de couleurs uniforme, comme certains logos.

Lorsqu'on écrit un texte sur une photo, il y a un conflit et souvent on stocke l'image au format **.jpg** avec une grande qualité, typiquement de 90 %.

Le format **.gif** est similaire au format **.png** et s'utilise dans les mêmes situations. En principe le format **.png** lui est préférable.

Le format **.bmp** ne comprime pas les images par défaut, il doit être évité.

Notion de dimension d'une image, de couleur et de résolution.

Une image **bitmap** est définie par un tableau de points, appelés « pixels » pour « PicturE LEment ». Typiquement une image peut contenir 800 lignes de 1200 points chacune, ou autrement dit, elle contient 1200 colonnes de 800 points chacune.

Chaque point est un mélange de trois couleurs : du Rouge, du Vert et du Bleu.

C'est le codage **RVB** ou en anglais : **RGB** (Red Green Blue).

Chaque pixel est défini par trois nombres entre 0 et 255. Le premier indique la quantité de **Rouge**, le second la quantité de **Vert** et le dernier la quantité de **Bleu**.

Rouge = `rgb(255, 0,0)` ; Vert = `rgb(0,255,0)` ; Bleu = `rgb(0,0,255)` ; cyan = `rgb(0,255,255)`.

On utilise aussi la notation hexadécimale. `#ff0000` = `rgb(255,0,0)` `#4080c0` = `rgb(64, 128, 192)`.
`a=10` ; `b=11` ; `c=12` ; `d=13` ; `e=14` ; `f=15`. `#xy` = `x · 16 + y`

Lorsqu'on imprime une image, il faut spécifier le nombre de points (=nombre de pixels) qu'il faut imprimer par centimètre. Vu que l'informatique est dominée par les États-Unies, on spécifie le nombre de points par pouce, ou « **dpi** » « Dot Per Inch ».

Ce nombre de points par pouce (ou par centimètre) s'appelle **la résolution de l'image**.

Il est utile de savoir qu'**un pouce (= un Inch) = 2,54 centimètres**.

Une résolution très courante est de 300 dpi.

Une résolution de 600 dpi est de très (trop) bonne qualité.

Les moniteurs d'ordinateur ont généralement une résolution légèrement inférieure à 100 dpi.

Question 1.

Si une image est de 1200x800 pixels et qu'on l'imprime avec une résolution de 300 dpi, quelle sera sa taille en centimètres ?

Manipulation 1.

- Chercher une photo, de dimension entre 600x400 et 1200x800 pixels et sauvegardez-la.
- Lancez le logiciel **GIMP**, ouvrez la photo.
- **Exportez**-la au format **.jpg** en ajoutant « _000 » à la fin de son nom, sauveez-la avec une qualité = 0.
- Recommencez en **exportant** la même photo avec les qualités suivantes : 10 ; 20 ; 30 ; 50 ; 70 ; 90 ; 95 ; 100. Nommez chaque fois le nom du fichier de manière adéquate.
- Regardez, dans le dossier contenant ces images, leur taille et visualisez-les pour voir leur qualité.

Manipulation 2.

- Dans **GIMP**, avec la photo précédente, allez dans « Image » > « Échelle et taille de l'image... »
- Définissez la « Largeur » et la « Hauteur » à 50 %, peut importe la résolution.
- Exportez votre nouvelle image avec une qualité de 90 %.
- Comparez la taille du fichier contenant la plus petite image à celle sauvée précédemment avec une qualité de 90 %.

Lorsque vous incluez des images dans vos pages Web, **il est important de diminuer la taille de l'image** pour l'ajuster à celle désirée. De plus, si c'est une photo, réduisez la qualité en fonction de ce qui est utile. Souvent on met une petite photo, de faible qualité, qui envoie vers la même photo, de plus grande taille et de meilleure qualité.

Manipulation 3.

- Cherchez une image contenant de grandes zones de couleur uniforme. C.f. <http://openclipart.org/>
- Si l'image est au format **.svg**, elle est stockée sous un format vectoriel. Vous pouvez quand-même l'ouvrir avec GIMP, mais il vous sera demandé la largeur, la hauteur, les ratios et la résolution. Vous pouvez garder les dimensions par défaut.
- Exportez cette image au format **.png** et au format **.jpg** (qualité 90%) et comparez la taille des fichiers, ainsi que la qualité des images.

Manipulation 4.

- Dans **GIMP** créer une nouvelle image de taille 300x200.
- Faites un dessin quelconque.
- Effacer une partie du dessin, la couleur sera celle du fond de l'image, généralement blanc. On veut souvent que le fond soit transparent. Pour cela :
- Allez dans « Calques et brosses », fait un clique droit sur l'image miniature et ajoutez un « canal alpha »
- Sélectionnez la gomme et effacez une partie du dessin, vous voyez que même le fond s'efface. La région effacée est transparente.
- Toujours avec la gomme sélectionnée, « effacez » la partie transparente, en pressant en même temps sur la touche « Alt ». Le dessin qui était effacé réapparaît !

Mise en forme avec le CSS

« CSS » signifie « Cascading Style Sheet », « Feuille de Style en Cascade »

L'idée est la suivante :

Dans le code HTML, on code l'information que l'on désire afficher dans la page Web et dans les instructions CSS on stocke la manière de présenter cette information.

Nous verrons 3 manières de définir le style (format) du texte dans une page Web avec le CSS.

- > Lancez SeaMonkey, dans : Édition > Préférence, Compositeur, cochez la case :
« Utiliser les styles CSS plutôt que les éléments et attributs HTML »
- > Éditez une de vos pages Web, ou créez-en une nouvelle et écrivez un petit texte, en mettant quelques mots en gras, d'autres en italique, une partie en souligné, certains en gras et italique.
 - Changez la taille de caractères.
 - Écrivez des titres (Format > paragraphe > titre 1) etc.

Ensuite, visualisez le code « <HTML> Source ».

Vous pourrez constater que pour mettre un texte en gras, l'instruction est :

```
<span style="font-weight: bold;">texte est en gras</span>
```

L'attribut « style » est la **première manière** de formater le texte.

Une ancienne manière est d'écrire `ce texte est en gras`, qui utilise la balise ``.

Il est conseillé de la combiner avec la deuxième manière de formater le texte en CSS.

Définition d'une feuille de style CSS pour modifier la présentation de la page Web

La **deuxième manière** utilise la balise `<style>` et terminent par `</style>`.

Elles se placent dans la section `<head> ... </head>`.

Les commentaires sont entre `/*` et `*/`, donc

```
<style> /* Ceci est un commentaire */ </style>
```

On peut définir le style d'un certain type de balise, exemple :

```
<style>
p { color:red } /* indiquera que le texte de tous les paragraphes sont en rouge (red) */
b { font-weight: bold; } /* c'est bien de l'écrire, même si c'est le comportement par défaut */
</style>
```

On peut définir une classe, définir son style et indiquer qu'une balise est de la classe donnée. Exemple :

```
<style>
.MonNomDeClasse { text-align:center; font-size:120%; font-weight:bold ; }
</style>
```

L'indication suivante, indiquera la classe de la balise :

```
<p class="MonNomDeClasse"> blablabla ... </p>
```

Donc le texte blablabla ... sera centré, de taille 120% et en gras.

c.f. <http://www.juggling.ch/zgisin/> pour des exemples.

c.f. <http://www.w3schools.com/cssref/default.asp> pour la référence.

3^{ème} manière de définir les styles :

Si on désire définir plusieurs pages avec le même type de style, on peut définir les styles dans un fichier et l'inclure avec l'instruction :

```
<link rel="stylesheet" type="text/css" href="styles/nom_du_fichier_de_style.css">
```

en supposant que ce fichier se trouve dans le sous-répertoire « styles ».

Remarque :

Commentaires dans du code html : `<!-- commentaires -->`

Commentaires dans du code CSS : `/* commentaires */`

Commentaires dans du code javascript : `// commentaires sur une seule ligne.`

Allez sur ma page Web : http://www.juggling.ch/zgisin/a2015_oc3/index.html

Allez sur ma "**page060_....html**", *regardez son code source* et sauvegardez-le chez vous sous le nom : "ex001_styles.html"

Lancez Netbeans et ouvrez ce fichier "ex001_styles.html".

Lisez et étudiez le code html de cette page.

Chaque balise peut avoir un attribut "style", qui permet de mettre en forme le texte.

Deux balises ont été introduites, juste pour mettre en forme des portions de pages.

1) La balise **** ne modifie en rien votre page Web. C'est le style qu'on lui donne qui modifiera la page. Exemples :

a) `rien ne change !`

b) `texte est en gras`

c) `celui-ci en italique`

2) La balise **<div>** crée un nouveau paragraphe. Elle impose un retour à la ligne.

Exemples :

`<div style="text-align:center">`

Autre essai de définition de style avec " text-align:center " dans un div.

`</div>`

`<div style="width:660px; border-style:solid; text-align:left; margin-top:5px; margin-right:8px; margin-bottom:5px; padding:5px 8px 5px 8px; ">`

Cette section est centrée et cardée. `
`

Référez-vous au code html du fichier que vous venez de sauver.

`</div>`

Créez une page `ex002_styles.html` pour faire d'autres essais !

Par exemple, ajouter un style à la balise `<body>` :

`<body style="background-color:rgb(255, 255, 128);">`

"rgb" indique la quantité de couleurs Red, Green, Blue utilisée. 0 = minimum, 255 = maximum.

Ce même style peut être ajouté à une balise ``, `<div>` ou n'importe quelle autre.

Exemple :

`spécialement`

Regardez les exemples de votre fichier "`ex001_styles.html`" et testez-en d'autres dans votre fichier : "`ex002_styles.html`"

Les quatre pages qui suivent donne de nombreux styles possibles et de nombreux styles possibles.

Netbeans vous aide en indiquant des suggestions, des erreurs et les positions des balises correspondantes à celle sur laquelle se trouve le curseur.

Mémento des propriétés CSS Source : "Réalisez votre SITE WEB avec HTML5 et CSS3", de Mathieu Nebra. Issu du Site OpenClassRooms, <http://fr.openclassrooms.com/>
 Pour beaucoup d'exemples, voir : http://www.juggling.ch/zgisin/a2015_oc3/index.html
 Pour une liste complète, c.f. : <http://www.w3schools.com/cssref/default.asp>

Propriétés de mise en forme du texte (et des images). C'est-à-dire de tout ce qui touche à la présentation du texte proprement dit : le gras, l'italique, le souligné, la police, l'alignement, etc.

Propriété	Description
font-family	Nom de police (Arial, Helvetica, Times, Courier, etc.)
@font-face	Police personnalisée
font-size	Taille de texte, typique est 12pt; ou font-size:150%; augmente de 50% la taille.
font-weight	font-weight:400; <=> normal, font-weight:700; <=> gras
font-style	font-style:normal; ou <i>italic</i> ; ou oblique;
text-decoration	text-decoration:none; ou <u>overline</u> ; ou <u>underline</u> ; ou line-through ;
font-variant	font-variant:normal; ou SMALL-CAPS;
text-transform	:none; :capitalize; :uppercase; :lowercase; change minuscules - majuscules
font	Super propriété de police, Combine : font-weight, font-style, font-size, font-variant- font-family.
text-align	:left; :center; :right; :justify; alignement horizontal
vertical-align	:top; middle; :bottom; :baseline; :sub; :super; :text-top; :text-bottom;
line-height	hauteur de ligne, typique est line-height:24pt;
text-indent	Indentation de la première ligne
white-space	Césures
word-wrap	Césures forcées
text-shadow	Ombre de texte

Propriétés de couleur et de fond

Propriété	Description
color	Couleur du texte, color:rgb(<i>rouge, vert, bleu</i>); color:#ffffff; color:blue;
background-color	Couleur du fond
background-image	Image de fond, ex. background-image:url(images/L_intrus.png);
background-attachment	Fond fixe
background-repeat	Répétition du fond
background-position	Position du fond
background	Super propriété, qui regroupe les 4 dernières propriétés ci-dessus.
opacity	Défini l'opacité et donc la transparence, par un nombre entre 0 et 1.

Propriétés des boîtes (et de ligne horizontale <hr>)

Propriété	Description
width	Largeur, typique en pixels "px" ou en relatif "%"
height	Hauteur, typique en pixels "px" ou en relatif "%"
min-width	Largeur minimale
max-width	Largeur maximale
min-height	Hauteur minimale
max-height	Hauteur maximale
margin-top	Marge du haut, position du haut du cadre relativement à ce qui précède.
margin-right	Marge à droite
margin-bottom	Marge du bas
margin-left	Marge à gauche, position de la gauche du cadre.
margin	Super propriété qui combine les 4 précédentes.
padding-top	Marge intérieure en haut, position du texte, dans le cadre, relativement au haut
padding-right	Marge intérieure à droite
padding-bottom	Marge intérieure en bas
padding-left	Marge intérieure à gauche
padding	Super propriété qui combine les 4 précédentes.
border-width	Épaisseur de la bordure
border-color	Couleur de la bordure
border-style	Type de bordure, none; solide; double; dotted; dashed; hidden; groove; ridge; ...
border	Super propriété qui combine les 3 précédentes.
border-radius	Bordure arrondie
box-shadow	Ombre de boîte

Propriétés de positionnement et d'affichage

Propriété	Description
display	none; (non affiché et ne prend pas de place) inline-block; est affiché en un petit bloque, block; inline;
visibility	visible; hidden; (caché, mais prend de la place), collapse
clip	Affichage d'une partie de l'élément, mais pas ce qui sort du "cadre"
overflow	Définit le comportement en cas de dépassement
float	Avec "display:inline-block", permet un affichage au-dessus du reste de la page.
clear	Arrêt d'un flottant
position	Positionnement, static; relative; absolute; fixed;
top	Positionnement par rapport au haut
bottom	Positionnement par rapport au bas
left	Positionnement par rapport à la gauche
right	Positionnement par rapport à la droite
z-index	Ordre d'affichage en cas de superposition, très utile avec le javascript et déplacement d'images.

Utilisé avec "hover", permet de faire apparaître un texte, c.f. http://www.juggling.ch/gisin/bgweb/a2css1030_liste_exemples_style_suite_2.html

Propriétés des listes

Propriété	Description
list-style-type	Type de liste, disc; circle; square; decimal; ...
list-style-position	Position en retrait, en px; ou en %
list-style-image	Puce personnalisée par une image.
list-style-	Super propriété, qui combine les 3 précédentes

Propriétés des tableaux `<table style="..."> ... </table>`

Propriété	Description
border-collapse	Fusion des bordures de tableaux, très souvent utilisé : <code><table style="border-collapse:collapse;"></code>
empty-cells	Affiche les cellules vides
caption-side	Position du titre du tableau

Propriétés du curseur de la souris

Propriété	Description
cursor	Définit l'apparence du curseur de la souris, c.f. la fin de : http://www.juggling.ch/gisin/bgweb/a2css1010_liste_exemples_style.html

Quelques balises de structuration du texte

`` est une balise générique, de type inline, qui suit donc le cours de la ligne.

`<div>` est l'autre balise générique, de type block, qui passe à la ligne suivante.

Ces deux balises sont très souvent utilisées pour définir des styles particuliers de portions de pages Web.

`<sup>` Exposant `<sub>` Indice `` Mise en valeur forte `` Mise en valeur normale

`<h1>` à `<h6>`, divers niveaux de titre

`` pour insérer une image

`<audio ...>` pour insérer un son, avec contrôle du son.

`<video ...>` pour insérer une vidéo, avec contrôle de la vidéo.

`<source ...>` indique la source pour les formats son et audio.

`` pour les liens hypertextes

`
` pour les sauts de lignes

`<p>` pour les paragraphes

`<hr>` pour afficher une ligne horizontale, séparatrice.

`<pre>` pour un texte formaté, les sauts de lignes et espaces sont affichés comment dans la source.

Balises de listes

`` Liste à puce, non numérotée

`` Liste numérotée

`` Élément de la liste à puces

`<dl>` Liste de définitions

`<dt>` Terme à définir

`<dd>` Définition du terme

Balise de tableau

Balise	Description
<table>	Définition du début d'un tableau
<caption>	Titre du tableau, optionnel
<tr>	Ligne de tableau, qui contient plusieurs <td> (cellules) et éventuellement un <th>
<th>	En-tête de cellule, optionnel
<td>	Cellule, se trouvant dans une ligne de tableau
<thead>	Section de l'en-tête du tableau, optionnel
<tbody>	Section du corps du tableau, optionnel
<tfoot>	Section du pied du tableau, optionnel

Balise de formulaire

Balise	Description
<form>	Définition du début d'un formulaire
<fieldset>	Groupe de champs
<legend>	Titre d'un groupe de champs
<label>	Libellé d'un champ
<input>	Champ de formulaire, <input name="nom" value="777" size="5" type="text"> ou <input name="nom" value=" en accord !" type="button">
<textarea>	Zone de saisie multiligne
<option>	Élément d'une liste déroulante
<optgroup>	Groupe d'éléments d'une liste déroulante

class="nom" indique le nom de la classe CSS à utiliser. Elle est définie par *.nom*
 id="nom" indique le nom de l'id CSS à utiliser. Il est **unique**. Il est défini par *#nom*

Pour plus d'information sur les formulaires, voir les pages :
http://www.juggling.ch/gisin/bgweb/a3js1040_formulaires.html
http://www.juggling.ch/gisin/bgweb/a3js1050_formulaires_avec_code.html

Pour plus d'information sur Javascript, voir la page :
<http://www.juggling.ch/gisin/bgweb/aprod0010/javascript.html>

Pour plus d'information sur la gestion de son avec la balise <audio>, voir la page :
<http://www.juggling.ch/gisin/javascript/PlaySound/PlaySound03.html>

Pour plus d'information sur la gestion audio, voir la page :
<http://www.juggling.ch/gisin/javascript/Videos/videos.html>

Initiation à Javascript

Javascript est un langage de programmation qui prend de plus en plus d'importance dans le monde d'Internet.

Le but de ce cours est de vous **initier aux bases de l'algorithmique et des langages de programmation**. Nous utiliserons le langage Javascript, car il est simple d'approche, permet d'obtenir rapidement des résultats intéressants. L'approche par **exemples et expérimentation** sera privilégiée par rapport à la théorie.

Tout langage de programmation contient :

- ° des variables, pour stocker de l'information ;
- ° des opérateurs (+ - x / = && || ==) pour manipuler le contenu des variables ;
- ° des tests (if) pour exécuter des instructions seulement sous certaines conditions ;
- ° des boucles (for while do) pour exécuter plusieurs fois certaines instructions ;
- ° des fonctions, pour clarifier et simplifier le code ;
- ° des "inputs" pour accepter des entrées d'un utilisateur, d'un micro, d'une caméra, etc. ;
- ° des "outputs" pour afficher des résultats ou agir sur l'écran, l'imprimante, des haut-parleurs, etc. ;
- ° des gestions d'événements, pour réagir à certains événements, tels que des clics de la souris.

L'essentielle de ce cours se fera dans des séries d'exercices sur le site :

<http://www.juggling.ch/gisin/coursinfo3oc/index.html>

et des exemples sur le site :

http://www.juggling.ch/zgin/a2015_oc3/index.html

Pour un **micro résumé de Javascript**...

c.f. http://www.juggling.ch/gisin/bgweb/adocs/Javascript_syntaxe_resume.pdf

Pour un exercice et par curiosité :

Dans les pages qui suivent, viennent un calendrier perpétuel et un algorithme donnant le jour de la semaine correspondant à une date donnée. L'algorithme peut se faire de tête !

Correspondance, calendrier, en fonction de l'année **19xx**, ainsi que 15xx ; 23xx ; 27xx ; etc.

xx 00 01 02 03 **04** 05 06 07 **08** 09 10 11 **12** 13 14 15 **16** 17 18 19 **20** 21 22 23 **24** 25 26 27 **28**
 29 30 31 **32** 33 34 35 **36** 37 38 39 **40** 41 42 43 **44** 45 46 47 **48** 49 50 51 **52** 53 54 55 **56**
 57 58 59 **60** 61 62 63 **64** 65 66 67 **68** 69 70 71 **72** 73 74 75 **76** 77 78 79 **80** 81 82 83 **84** 85 86 87 **88** 89 90 91 **92** 93 94 95 **96** 97 98 99
 0 1 2 3 **15** 6 0 1 **13** 4 5 6 **11** 2 3 4 **16** 0 1 2 **14** 5 6 0 **12** 3 4 5 **10** 1 2 3 **15** 6 0 1 **13** 4 5 6 **11** 2 3 4

Correspondance, calendrier, en fonction de l'année **20xx**, ainsi que 16xx ; 24xx ; 28xx ; etc.

xx **00** 01 02 03 **04** 05 06 07 **08** 09 10 11 **12** 13 14 15 **16** 17 18 19 **20** 21 22 23 **24** 25 26 27 **28**
28 29 30 31 **32** 33 34 35 **36** 37 38 39 **40** 41 42 43 **44** 45 46 47 **48** 49 50 51 **52** 53 54 55 **56**
56 57 58 59 **60** 61 62 63 **64** 65 66 67 **68** 69 70 71 **72** 73 74 75 **76** 77 78 79 **80** 81 82 83 **84** 85 86 87 **88** 89 90 91 **92** 93 94 95 **96** 97 98 99
16 0 1 2 **14** 5 6 0 **12** 3 4 5 **10** 1 2 3 **15** 6 0 1 **13** 4 5 6 **11** 2 3 4 **16** 0 1 2 **14** 5 6 0 **12** 3 4 5 **10** 1 2 3

Correspondance, calendrier, en fonction de l'année **21xx**, ainsi que 17xx ; 25xx ; 29xx ; etc.

xx 00 01 02 03 **04** 05 06 07 **08** 09 10 11 **12** 13 14 15 **16** 17 18 19 **20** 21 22 23 **24** 25 26 27 **28**
 29 30 31 **32** 33 34 35 **36** 37 38 39 **40** 41 42 43 **44** 45 46 47 **48** 49 50 51 **52** 53 54 55 **56**
 57 58 59 **60** 61 62 63 **64** 65 66 67 **68** 69 70 71 **72** 73 74 75 **76** 77 78 79 **80** 81 82 83 **84** 85 86 87 **88** 89 90 91 **92** 93 94 95 **96** 97 98 99
 4 5 6 0 **12** 3 4 5 **10** 1 2 3 **15** 6 0 1 **13** 4 5 6 **11** 2 3 4 **16** 0 1 2 **14** 5 6 0 **12** 3 4 5 **10** 1 2 3 **15** 6 0 1

Correspondance, calendrier, en fonction de l'année **22xx**, ainsi que 18xx ; 26xx ; 30xx ; etc.

xx 00 01 02 03 **04** 05 06 07 **08** 09 10 11 **12** 13 14 15 **16** 17 18 19 **20** 21 22 23 **24** 25 26 27 **28**
 29 30 31 **32** 33 34 35 **36** 37 38 39 **40** 41 42 43 **44** 45 46 47 **48** 49 50 51 **52** 53 54 55 **56**
 57 58 59 **60** 61 62 63 **64** 65 66 67 **68** 69 70 71 **72** 73 74 75 **76** 77 78 79 **80** 81 82 83 **84** 85 86 87 **88** 89 90 91 **92** 93 94 95 **96** 97 98 99
 2 3 4 5 **10** 1 2 3 **15** 6 0 1 **13** 4 5 6 **11** 2 3 4 **16** 0 1 2 **14** 5 6 0 **12** 3 4 5 **10** 1 2 3 **15** 6 0 1 **13** 4 5 6

0

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Lu
Ma	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Ma
Me	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Me
Je	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Je
Ve	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Ve
Sa	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Sa
Di	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Di
	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Lu
Ma	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Ma
Me	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Me
Je	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Je
Ve	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Ve
Sa	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Sa
Di	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Di

1

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Lu
Ma	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Ma
Me	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Me
Je	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Je
Ve	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Ve
Sa	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Sa
Di	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Di
	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Lu
Ma	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Ma
Me	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Me
Je	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Je
Ve	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Ve
Sa	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Sa
Di	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Di

2

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Lu
Ma	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Ma
Me	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Me
Je	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Je
Ve	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Ve
Sa	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Sa
Di	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Di
	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Lu
Ma	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Ma
Me	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Me
Je	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Je
Ve	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Ve
Sa	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Sa
Di	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Di

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Lu
Ma	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Ma
Me	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Me
Je	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Je
Ve	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Ve
Sa	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Sa
Di	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Lu
Ma	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Ma
Me	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Me
Je	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Je
Ve	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Ve
Sa	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Sa
Di	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Di

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Lu
Ma	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Ma
Me	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Me
Je	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Je
Ve	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Ve
Sa	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Sa
Di	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Lu
Ma	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Ma
Me	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Me
Je	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Je
Ve	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Ve
Sa	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Sa
Di	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Di

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Lu
Ma	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Ma
Me	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Me
Je	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Je
Ve	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Ve
Sa	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Sa
Di	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Lu
Ma	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Ma
Me	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Me
Je	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Je
Ve	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Ve
Sa	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Sa
Di	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Di

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	2 9 16 23 30	6 13 20 27	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Lu
Ma	3 10 17 24 31	7 14 21 28	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Ma
Me	4 11 18 25	1 8 15 22	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Me
Je	5 12 19 26	2 9 16 23	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Je
Ve	6 13 20 27	3 10 17 24	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Ve
Sa	7 14 21 28	4 11 18 25	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Sa
Di	1 8 15 22 29	5 12 19 26	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Lu
Ma	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Ma
Me	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Me
Je	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Je
Ve	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Ve
Sa	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Sa
Di	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Di

Correspondance, calendrier, en fonction de l'année **19xx**, ainsi que 15xx ; 23xx ; 27xx ; etc.

xx	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28															
	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56																
	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	
	0	1	2	3	15	6	0	1	13	4	5	6	11	2	3	4	16	0	1	2	14	5	6	0	12	3	4	5	10	1	2	3	15	6	0	1	13	4	5	6	11	2	3	4

Correspondance, calendrier, en fonction de l'année **20xx**, ainsi que 16xx ; 24xx ; 28xx ; etc.

xx	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28															
	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56															
	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99
	16	0	1	2	14	5	6	0	12	3	4	5	10	1	2	3	15	6	0	1	13	4	5	6	11	2	3	4	16	0	1	2	14	5	6	0	12	3	4	5	10	1	2	3

Correspondance, calendrier, en fonction de l'année **21xx**, ainsi que 17xx ; 25xx ; 29xx ; etc.

xx	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28															
	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56																
	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	
	4	5	6	0	12	3	4	5	10	1	2	3	15	6	0	1	13	4	5	6	11	2	3	4	16	0	1	2	14	5	6	0	12	3	4	5	10	1	2	3	15	6	0	1

Correspondance, calendrier, en fonction de l'année **22xx**, ainsi que 18xx ; 26xx ; 30xx ; etc.

xx	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28															
	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56																
	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	
	2	3	4	5	10	1	2	3	15	6	0	1	13	4	5	6	11	2	3	4	16	0	1	2	14	5	6	0	12	3	4	5	10	1	2	3	15	6	0	1	13	4	5	6

10

	Janvier					Février					Mars					Avril					Mai					Juin								
Lu	2	9	16	23	30		6	13	20	27		5	12	19	26		2	9	16	23	30		7	14	21	28		4	11	18	25		Lu	
Ma	3	10	17	24	31		7	14	21	28		6	13	20	27		3	10	17	24		1	8	15	22	29		5	12	19	26		Ma	
Me	4	11	18	25		1	8	15	22	29		7	14	21	28		4	11	18	25		2	9	16	23	30		6	13	20	27		Me	
Je	5	12	19	26		2	9	16	23		1	8	15	22	29		5	12	19	26		3	10	17	24	31		7	14	21	28		Je	
Ve	6	13	20	27		3	10	17	24		2	9	16	23	30		6	13	20	27		4	11	18	25		1	8	15	22	29		Ve	
Sa	7	14	21	28		4	11	18	25		3	10	17	24	31		7	14	21	28		5	12	19	26		2	9	16	23	30		Sa	
Di	1	8	15	22	29		5	12	19	26		4	11	18	25		1	8	15	22	29		6	13	20	27		3	10	17	24		Di	
	Juillet					Août					Septembre					Octobre					Novembre					Décembre								
Lu	2	9	16	23	30		6	13	20	27		3	10	17	24		1	8	15	22	29		5	12	19	26		3	10	17	24	31		Lu
Ma	3	10	17	24	31		7	14	21	28		4	11	18	25		2	9	16	23	30		6	13	20	27		4	11	18	25		Ma	
Me	4	11	18	25		1	8	15	22	29		5	12	19	26		3	10	17	24	31		7	14	21	28		5	12	19	26		Me	
Je	5	12	19	26		2	9	16	23	30		6	13	20	27		4	11	18	25		1	8	15	22	29		6	13	20	27		Je	
Ve	6	13	20	27		3	10	17	24	31		7	14	21	28		5	12	19	26		2	9	16	23	30		7	14	21	28		Ve	
Sa	7	14	21	28		4	11	18	25		1	8	15	22	29		6	13	20	27		3	10	17	24		1	8	15	22	29		Sa	
Di	1	8	15	22	29		5	12	19	26		2	9	16	23	30		7	14	21	28		4	11	18	25		2	9	16	23	30		Di

11

	Janvier					Février					Mars					Avril					Mai					Juin								
Lu	1	8	15	22	29		5	12	19	26		4	11	18	25		1	8	15	22	29		6	13	20	27		3	10	17	24		Lu	
Ma	2	9	16	23	30		6	13	20	27		5	12	19	26		2	9	16	23	30		7	14	21	28		4	11	18	25		Ma	
Me	3	10	17	24	31		7	14	21	28		6	13	20	27		3	10	17	24		1	8	15	22	29		5	12	19	26		Me	
Je	4	11	18	25		1	8	15	22	29		7	14	21	28		4	11	18	25		2	9	16	23	30		6	13	20	27		Je	
Ve	5	12	19	26		2	9	16	23		1	8	15	22	29		5	12	19	26		3	10	17	24	31		7	14	21	28		Ve	
Sa	6	13	20	27		3	10	17	24		2	9	16	23	30		6	13	20	27		4	11	18	25		1	8	15	22	29		Sa	
Di	7	14	21	28		4	11	18	25		3	10	17	24	31		7	14	21	28		5	12	19	26		2	9	16	23	30		Di	
	Juillet					Août					Septembre					Octobre					Novembre					Décembre								
Lu	1	8	15	22	29		5	12	19	26		2	9	16	23	30		7	14	21	28		4	11	18	25		2	9	16	23	30		Lu
Ma	2	9	16	23	30		6	13	20	27		3	10	17	24		1	8	15	22	29		5	12	19	26		3	10	17	24	31		Ma
Me	3	10	17	24	31		7	14	21	28		4	11	18	25		2	9	16	23	30		6	13	20	27		4	11	18	25		Me	
Je	4	11	18	25		1	8	15	22	29		5	12	19	26		3	10	17	24	31		7	14	21	28		5	12	19	26		Je	
Ve	5	12	19	26		2	9	16	23	30		6	13	20	27		4	11	18	25		1	8	15	22	29		6	13	20	27		Ve	
Sa	6	13	20	27		3	10	17	24	31		7	14	21	28		5	12	19	26		2	9	16	23	30		7	14	21	28		Sa	
Di	7	14	21	28		4	11	18	25		1	8	15	22	29		6	13	20	27		3	10	17	24		1	8	15	22	29		Di	

12

	Janvier					Février					Mars					Avril					Mai					Juin							
Lu	7	14	21	28		4	11	18	25		3	10	17	24	31		7	14	21	28		5	12	19	26		2	9	16	23	30		Lu
Ma	1	8	15	22	29		5	12	19	26		4	11	18	25		1	8	15	22	29		6	13	20	27		3	10	17	24		Ma

13

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	6 13 20 27	3 10 17 24	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Lu
Ma	7 14 21 28	4 11 18 25	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Ma
Me	1 8 15 22 29	5 12 19 26	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Me
Je	2 9 16 23 30	6 13 20 27	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Je
Ve	3 10 17 24 31	7 14 21 28	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Ve
Sa	4 11 18 25	1 8 15 22 29	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Sa
Di	5 12 19 26	2 9 16 23	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Lu
Ma	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Ma
Me	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Me
Je	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Je
Ve	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Ve
Sa	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Sa
Di	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Di

14

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	5 12 19 26	2 9 16 23	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Lu
Ma	6 13 20 27	3 10 17 24	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Ma
Me	7 14 21 28	4 11 18 25	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Me
Je	1 8 15 22 29	5 12 19 26	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Je
Ve	2 9 16 23 30	6 13 20 27	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Ve
Sa	3 10 17 24 31	7 14 21 28	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Sa
Di	4 11 18 25	1 8 15 22 29	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Lu
Ma	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Ma
Me	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Me
Je	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Je
Ve	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Ve
Sa	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Sa
Di	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Di

15

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	4 11 18 25	1 8 15 22 29	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Lu
Ma	5 12 19 26	2 9 16 23	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Ma
Me	6 13 20 27	3 10 17 24	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Me
Je	7 14 21 28	4 11 18 25	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Je
Ve	1 8 15 22 29	5 12 19 26	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Ve
Sa	2 9 16 23 30	6 13 20 27	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Sa
Di	3 10 17 24 31	7 14 21 28	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Lu
Ma	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Ma
Me	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Me
Je	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Je
Ve	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Ve
Sa	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Sa
Di	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Di

16

	Janvier	Février	Mars	Avril	Mai	Juin	
Lu	3 10 17 24 31	7 14 21 28	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	Lu
Ma	4 11 18 25	1 8 15 22 29	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	Ma
Me	5 12 19 26	2 9 16 23	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	Me
Je	6 13 20 27	3 10 17 24	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	Je
Ve	7 14 21 28	4 11 18 25	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	Ve
Sa	1 8 15 22 29	5 12 19 26	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	Sa
Di	2 9 16 23 30	6 13 20 27	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	Di

	Juillet	Août	Septembre	Octobre	Novembre	Décembre	
Lu	3 10 17 24 31	7 14 21 28	4 11 18 25	2 9 16 23 30	6 13 20 27	4 11 18 25	Lu
Ma	4 11 18 25	1 8 15 22 29	5 12 19 26	3 10 17 24 31	7 14 21 28	5 12 19 26	Ma
Me	5 12 19 26	2 9 16 23 30	6 13 20 27	4 11 18 25	1 8 15 22 29	6 13 20 27	Me
Je	6 13 20 27	3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	Je
Ve	7 14 21 28	4 11 18 25	1 8 15 22 29	6 13 20 27	3 10 17 24	1 8 15 22 29	Ve
Sa	1 8 15 22 29	5 12 19 26	2 9 16 23 30	7 14 21 28	4 11 18 25	2 9 16 23 30	Sa
Di	2 9 16 23 30	6 13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26	3 10 17 24 31	Di

Quelques explications.

Le calendrier perpétuel de ces pages suit le **calendrier grégorien**. Il porte le nom de son instigateur, le pape Grégoire XIII.

Ce calendrier n'a pas été adopté par tous les pays la même année !

Avant, le calendrier julien faisait fois. Il a été établi sous Jules César en -46.

Dans les pays catholiques, dépendant de Rome, le passage du calendrier julien au calendrier grégorien s'est fait la nuit du 4 au 5 octobre 1582. Pour corriger pour le décalage des jours, le 5 octobre s'est retrouvé un 15 octobre, donc les 10 jours du 5 au 14 octobre 1582 n'existent pas.

(Thérèse d'Avila est morte durant la nuit du 4 au 15 octobre 1582!)

Pays concernés : Espagne, Portugal, Italie,

En France, le 9 décembre 1582 est suivi par le 20 décembre 1582.

En Autriche, Belgique, Pays-Bas, entre 1583 et 1584.

En Allemagne, en mars 1700, certaines régions avant. Les états catholiques, dès 1585.

En Danemark et Norvège, le 18 février 1700 est suivi par le 1er mars 1700.

En Suisse, beaucoup de variabilité, mais, à partir de 1701 tous l'avaient adopté.

En Angleterre, le 2 septembre 1752 est suivi par le 14 septembre 1752. En 1700 il y avait 11 jours de décalage.

En Russie, le 31 janvier 1918 est suivi par le 14 février 1918.

En Chine, soit en 1912, soit en 1929, soit en 1949.

En Turquie, en 1927.

Certains pays ne l'ont pas adopté !

Il n'y a que **14 calendriers d'année possibles**. 7 pour les années bissextiles et 7 pour les non bissextiles.

Pour les années bissextiles, comme pour les autres, l'année est déterminée par son premier jour.

Il suffit donc d'imprimer que 14 calendriers et d'indiquer pour chaque année à quel calendrier elle correspond.

Tous les **4 siècles**, la séquence des jours se répète exactement (dans le calendrier grégorien).

C'est la raison pour laquelle, seul 4 siècles sont représentés, en commençant par les années 1900 (et donc aussi 2300, 2700, etc.).

Dans un siècle, tous les **28 ans**, la séquence des jours se répète exactement. L'exception étant le début de chaque *siècle*, qui n'est pas multiple de 4, comme 19⁰⁰, 18⁰⁰, 17⁰⁰ et 21⁰⁰, 22⁰⁰, 23⁰⁰.

Ces exceptions viennent du fait que ces années ne sont pas bissextiles, malgré le fait qu'elles sont des multiples de 4.

Le calendrier de l'année 1900 est identique à celui de l'année 1906.

De même pour les couples (1800 ; 1806) ; (1700 ; 1706) ; (2100 ; 2106) ; (2200 ; 2306) ; etc.

Pour les signes chinois, c.f. http://fr.wikipedia.org/wiki/Zodiaque_chinois

Pour les signes du Zodiaque, c.f. <http://fr.wikipedia.org/wiki/Zodiaque>

Bélier		20 mars 19 avril	Lion		22 juillet 22 août	Sagittaire		22 nov. 20 déc.
Taureau		20 avril 20 mai	Vierge		23 août 22 sept.	Capricorne		21 déc. 19 janvier
Gémeaux		21 mai 20 juin	Balance		23 sept. 22 oct.	Verseau		20 janvier 18 février
Cancer		21 juin 21 juillet	Scorpion		23 oct. 21 nov.	Poissons		19 février 19 mars

Détermine le jour de la semaine pour une date donnée1°) **Ajoute** :**A** : le nombre composé des deux derniers chiffres de l'année ;**Q** : l'entier juste en dessous ou égale au quart de A (= la partie entière de $A/4 = PE(A/4)$);**J** : le numéro du jour du mois de cette date (de 1 à 31) ;**M** : un nombre associé au mois de la date, à l'aide du tableau suivant :**C** : une correction due au siècle et à un cas particulier, qui sont décrits aux points 2°) et 3°).Correspondance : Mois → chiffre

Janvier → 0	Février → 3	Mars → 3	Avril → 6
Mai → 1	Juin → 4	Juillet → 6	Août → 2
Septembre → 5	Octobre → 0	Novembre → 3	Décembre → 5

2°) **Si** 1900 ≤ date ≤ 1999, alors **C = 0**. **Si** 2000 ≤ date ≤ 2099, alors **C = 6 (ou C = -1)**.Plus généralement : **C = 6 - 2 fois le reste de la division du siècle par 4.**

année =	15xx	16xx	17xx	18xx	19xx	20xx	21xx	22xx	23xx	24xx	25xx	26xx	27xx	28xx	etc.
C =	0	6	4	2	0	6	4	2	0	6	4	2	0	6	

3°) **Si** la date est janvier ou février d'une année bissextile, alors **ajoute 6 à C (ou soustrait 1 à C)**.

(2000 est une année bissextile, tous les 4 ans, il y a une année bissextile... sauf 1700, 1800, 1900, 2100, ...)

4°) **Divise le résultat (=A+Q+J+M+C) par 7** et garde le reste.5°) **Le reste de la division** donne le jour de la semaine :

0 → dimanche 1 → lundi 2 → mardi 3 → mercredi 4 → jeudi 5 → vendredi 6 → samedi

On peut *simplifier encore un peu les calculs* en soustrayant 28 ou 56 ou 84 ou 112 au nombre **A** représentant les deux derniers chiffres de l'année, car ces trois nombres sont des multiples de 4 et de 7.Rien n'interdit d'utiliser des nombres négatifs, mais : partie entière de $-0,5 = PE(-0,5) = -1$; $PE(-1,75) = -2$; $PE(-2) = -2$.Exemples :17.2.1958 → $A = 58 - 56 = 2$; $Q = PE(2/4) = 0$; $J = 17 = 3 + \text{multiple de } 7$; $M = 3$; $C = 0$ Jour = $A + Q + J + M (+C) = 2 + 0 + 3 + 3 = 1 + \text{multiple de } 7 = \text{lundi}$ 23.2.1964 → $A = 64 - 56 = 8$; $Q = PE(8/4) = 2$; $J = 23 = 2 + \text{multiple de } 7$; $M = 3$; $C = -1$ (car année bissextile.)Jour = $A + Q + J + M + C = 8 + 2 + 2 + 3 - 1 = 14 = 0 + \text{multiple de } 7 = \text{dimanche}$ 11.9.2001 → $A = 01$; $Q = PE(1/4) = 0$; $J = 11 = 4 + \text{multiple de } 7$; $M = 5$; $C = -1$ (car année ≥ 2000.)Jour = $A + Q + J + M + C = 1 + 0 + 4 + 5 - 1 = 9 = 2 + \text{multiple de } 7 = \text{mardi}$

Liste de dates pour s'entraîner :

mercredi	02-01-1963	mardi	12-04-1983	mercredi	19-09-1990
mardi	12-01-1926	jeudi	14-04-1960	dimanche	20-09-1981
vendredi	27-01-1989	lundi	16-04-1956	jeudi	26-09-1963
mercredi	28-01-1942	dimanche	17-04-1966	samedi	29-09-1984
jeudi	06-02-1986	mercredi	21-04-1965	mardi	06-10-1992
vendredi	11-02-1966	samedi	14-05-1988	jeudi	19-10-1989
samedi	13-02-1993	mardi	26-05-1964	vendredi	25-10-1991
samedi	14-02-1959	jeudi	29-05-1952	mercredi	30-10-1963
jeudi	18-02-1988	vendredi	13-06-1969	mercredi	30-10-1974
mardi	27-02-1996	dimanche	28-06-1942	dimanche	03-11-1968
dimanche	01-03-1964	mardi	28-06-1960	lundi	21-11-1955
jeudi	03-03-1983	dimanche	03-07-1988	jeudi	22-11-1934
dimanche	03-03-1996	vendredi	10-07-1959	mercredi	11-12-1957
vendredi	13-03-1936	vendredi	21-07-2000	dimanche	16-12-1979
vendredi	13-03-1998	dimanche	30-07-2000	samedi	21-12-1991
samedi	16-03-1985	samedi	15-08-1959	jeudi	17-02-2000
dimanche	24-03-1946	mercredi	21-08-1957	vendredi	29-11-2002
mardi	24-03-1981	mercredi	03-09-1969	jeudi	19-02-2004
dimanche	29-03-1959	samedi	14-09-1963	dimanche	24-12-2006
jeudi	04-04-1968	vendredi	15-09-1989	vendredi	08-02-2008
jeudi	10-04-1986	jeudi	17-09-1992	jeudi	26-08-2010
mardi	11-04-1967	jeudi	18-09-1969	mercredi	01-08-2012