1 Présentation / Éléments de cryptographie

1 / Thème abordé

1.1 Problématique, situations d'accroche

Comment assurer la confidentialité d'un message dans un monde où la notion de réseau est devenue incontournable ? Quelles situations nécessitent de crypter les données ? Quelles implications sociétales engendre ce type de démarche ?

De nombreuses situations d'accroche sont susceptibles de motiver les élèves pour conduire ce travail ; elles sont proposées sous forme de situations-problèmes que les élèves vont tenter de résoudre avec un peu de bon sens, puis avec l'aide des concepts apportés dans cette ressource.

1.2 Frontières de l'étude et prolongements possibles

Le niveau mathématique peut rapidement devenir conséquent ; il ne s'agit pas ici d'introduire un cours d'arithmétique mais de sensibiliser les élèves aux démarches de sécurité et de chiffrement.

On peut cependant évoquer le cryptage moderne (à clé publique par exemple) et donner les moyens de le mettre en œuvre, comme PGP ou GPG pour les courriels ou encore le protocole de sécurisation SSL des navigateurs web.

Ce scénario, orienté TP, peut se prolonger au cours de l'année par un mini-projet : l'élève devra être capable de démontrer son autonomie face à une situation de cryptage de données.

2 / Objectifs pédagogiques

2.1 Disciplines impliquées

Essentiellement les Mathématiques (arithmétique élémentaire), et éventuellement les STI (protocoles sur les réseaux) voire les Sciences Physiques (génération de nombres aléatoires).

2.2 Prérequis

Un peu d'algorithmique et de programmation.

2.3 Éléments du programme

Contenus

- Représentation numérique de l'information.
- Algorithmique : algorithmes simples.
- Langage et programmation : types de données, fonctions, correction d'un programme (partie projet).
- Architectures matérielles : réseaux, transmission de l'information.

Compétences et capacités

Décrire et expliquer une situation, un système ou un programme :

- Coder un nombre, un caractère au travers d'un code standard, un texte sous forme d'une liste de valeurs numériques.
- Comprendre et expliquer un algorithme.

Concevoir et réaliser une solution informatique en réponse à un problème :

- Concevoir un algorithme comme réponse à un problème.
- Programmer un algorithme.

Collaborer efficacement au sein d'une équipe dans le cadre d'un projet :

- Réaliser un système de chiffrement en travail d'équipe.

Communiquer à l'écrit et à l'oral :

- Présenter à la classe le mini-projet ou l'étude historique sur les codes secrets.

Faire un usage responsable des sciences du numérique :

- Mettre en place un dispositif permettant d'échanger des courriels chiffrés à partir d'une procédure géné-

rique, sécuriser un réseau wifi en respectant la loi.

3 / Modalités de mise en œuvre

3.1 Durée prévue pour la partie se déroulant en classe

Trois heures pour la partie TP et la synthèse. Le mini-projet sera réalisé sur plusieurs séances.

3.2 Type de l'animation

La partie TP sera conduite en demi-groupe. Suivra une synthèse sur le cryptage et ses implications sociétales. Cette partie peut se réaliser en classe entière.

Le mini-projet sera conduit par des petits groupes de deux ou trois élèves.

3.3 Projet

On peut orienter l'élève vers la réalisation d'un mini-projet, réalisé en petits groupes, autour du cryptage :

- Construire un programme en Python ou en Java cryptant et décryptant un message par substitution mono-alphabétique.
- Modéliser la machine Enigma avec un programme (cryptage, décryptage).
- Construire un programme (cryptant et décryptant) utilisant le chiffre de Vernam (clé secrète, loi de groupe, "Ou exclusif").
- Conduire une analyse fréquentielle d'un message crypté par substitution mono-alphabétique voire polyalphabétique (avec un peu plus d'arithmétique dans ce cas).
- Communiquer avec des courriels en PGP : installation et paramétrage des logiciels, création des clés publiques et privées, envoi de la clé publique sur un serveur de clés, mise en place de stratégies pour conserver la clé privée en sûreté.
- Ajouter des fonctionnalités Wi-fi à un petit réseau et le sécuriser.

3.4 Recherches documentaires

Des recherches seront à conduire sur Gilbert Vernam, Claude Shannon et Joseph Mauborgne.

Le mini-projet nécessitera aussi de nombreuses recherches documentaires pour préciser la problématique et donner des pistes de résolution.

3.5 Production des élèves

Un mini-projet qui pourra prendre la forme d'un dossier, d'un diaporama, d'un programme interactif, d'une exposition…

3.6 Évaluation

Les élèves présentent le mini-projet. On pourra s'inspirer de la pratique d'évaluation des TPE ainsi que de la grille des compétences utilisée pour l'épreuve comptant pour le baccalauréat.

4 / Outils

Langages de programmation (Python ou Java par exemple), navigateurs Web, routeur wifi, logiciel de courrier électronique prenant en charge le chiffrement GPG.

5 / Auteur

Philippe Jonin, professeur de Mathématiques, académie de Nantes

2 Éléments de cryptographie
3 Rendre un élève capable de crypter ses données

1 / Mise en situation

On propose d'introduire la problématique du chiffrement à travers une série de questions qui peuvent être présentées, étudiées et discutées en classe entière (les réponses ne sont pas toutes simples et n'apparaîtront complètement qu'au terme du parcours).

- M. A., mathématicien, est chercheur dans le domaine de la cryptographie. Il vient de mettre en place un nouvel algorithme de cryptage. Peut-il le publier sans compromettre la sécurité des futurs utilisateurs[1]?
- M. B. reçoit un mail d'un vieil ami qui le supplie de lui envoyer de l'argent pour faire face à une situation urgente. Peut-il s'assurer de l'intégrité du message et de l'identité de l'émetteur ? (notion de signature numérique d'un message).
- M. C. et M^me^ D. collaborent pour préparer un sujet d'examen, mais n'habitent pas dans la même ville. Comment peuvent-ils faire pour échanger leurs projets de sujets en toute sécurité ? (stratégie de chiffrement des courriels).
- M. E et M^me^ F. se connaissent depuis longtemps et correspondent par courriel. M. E habite en France et Mme F dans un pays où l'usage d'Internet est fortement encadré. Ont-ils intérêt à utiliser des courriels chiffrés ? (ouverture vers la stéganographie)
- M. G. et M^me^ H. sont dans la même entreprise. Ils échangent des courriels. Ont-ils le droit d'échanger des courriels chiffrés ?
- M. J., titulaire d'un BTS, est administrateur réseau dans un lycée. Un élève lui demande s'il pourrait installer un accès wifi. Comment M. J. peut-il répondre à cette demande s'il souhaite d'une part assurer la confidentialité des messages qui seront échangés et d'autre part répondre au cadre légal imposé par la loi Hadopi 2?
- M^me^ K. a remarqué que le code secret d'une carte bancaire était codé sur 4 chiffres. Elle en déduit que dans sa ville de 100 000 habitants il est très probable qu'au moins deux personnes aient le même code ! Vrai ou faux ? Si c'est vrai, s'agit-il d'une faille de sécurité ?
- M. L. souhaite faire des achats sur le web. Prend-il des risques en tapant le numéro de sa carte bancaire ?

Le choix d'une diversité de situations et d'approches tient compte du fait qu'on s'adresse à des élèves dont les profils et les choix d'orientation sont très variés ; il est important que tous les élèves se sentent concernés.

Pédagogiquement parlant, la liste de situations peut être proposée en dialogue (« en direct ») avec la classe, ou après un temps de recherche court (15 minutes) en petits groupes.

À l'issue de ce temps introductif, la terminologie peut être présentée : code secret (ou cryptogramme), chiffrement, cryptologie, cryptographie, clé secrète, cryptanalyse, etc.

2 / Étude active et détaillée de deux exemples

On privilégiera d'entrée un travail actif de décryptage-cryptage sur des exemples pour construire la première partie de ce scénario. L'idée est de donner les bases de la démarche de cryptage. La durée présumée de cette phase est de deux heures.

2.1 Le cryptosystème de César et le codage affine.

Ce cryptosystème consiste en une substitution mono-alphabétique et est donc sensible aux attaques fréquentielles et même aux attaques exhaustives puisque le nombre réduit de clés permet de toutes les tester.

Activité de l'élève : découverte du cryptosystème de César, notion de clé de cryptage.

On propose aux élèves de décrypter par tâtonnement un message crypté par le code de César. Une étude du texte de Suétone qui traite de ce codage permettra d'amorcer ces recherches.

On peut aussi inciter les élèves à utiliser deux disques concentriques représentant chacun l'alphabet.

1 Allusion au principe de Kerckhoffs.

On demande ensuite de construire une feuille de tableur permettant de crypter un message en prenant comme paramètre de la fonction de cryptage, la clé du système, qui est ici le décalage de la substitution. On exigera de bien scinder les différentes étapes : d'abord la numérisation du message clair en code ASCII, puis son cryptage par la fonction de codage et enfin la restitution du message final (on transforme à nouveau les codes ASCII en lettres).

On réfléchira alors aux implications d'un tel système : combien y a-t-il de codages possibles ? Est-il sûr ? Est-il possible de trouver directement la clé en analysant la fréquence des lettres et en tenant compte de la langue utilisée ? Comment adapter la feuille pour décrypter ? Pourquoi le Rot 13 (cas particulier où le décalage est de 13) a-t-il connu un tel succès ?

Une alternative, pour des élèves performants ou qui auraient déjà abordé le sujet (en seconde en enseignement de MPS par exemple), est de travailler sur le codage affine : la fonction de codage est ici du type $f(x) = (ax + b) \pmod{26}$ où x est le code ASCII correspondant à la lettre que l'on souhaite crypter. La clé est donc cette fois du type (a,b). Le paramètre a est-il libre de prendre toute valeur ? Combien y a-t-il de clés possibles ?

Le problème du décryptage est ici plus pointu et les outils mathématiques un peu plus complexes (statistique descriptive, arithmétique du pgcd).

Il peut être intéressant de faire travailler les élèves en petits groupes de façon à faciliter la gestion de l'hétérogénéité : on peut constituer des groupes homogènes ou au contraire repérer des personnes «ressource» que l'on placera dans chaque groupe.
Le travail en groupe permettra aussi de travailler dans les deux sens : on s'échange des messages cryptés comme autant de petits défis à relever.
Le niveau mathématique peut devenir très conséquent et il n'est pas prévu de développer un cours d'arithmétique ; il est donc préférable d'engager une démarche d'exploration et de découverte. Par exemple la recherche de la fonction réciproque de cryptage dans le cas d'un codage affine est hors de portée dans son cas général. On peut néanmoins poser le problème en termes mathématiques (résoudre une équation modulo n) et en chercher expérimentalement une solution.

La construction de la feuille de tableur va poser des problèmes techniques : elle nécessite de connaître quelques fonctions (CODE, MOD, CAR). Une petite fiche les détaillant pourra en conséquence être proposée. Il faudra aussi prévoir, avant le début de la séance, le nombre d'élèves susceptibles de travailler cette alternative de façon à les regrouper. Les limites du tableur vont aussi rapidement apparaître : difficultés pour traiter de gros volumes et gérer les entrées-sorties. C'est là une opportunité de basculer au langage de programmation comme le Python.

2.2 Le cryptosystème de Che Guevara.

Le cryptosystème de Che Guevara utilise une substitution alphabétique (au travers d'une table de transcodage) et lui associe une clé secrète générée aléatoirement dont la taille est égale à celle du message. Ce système de cryptage, de type Vernam, est relativement sûr et insensible aux attaques fréquentielles. Il n'est pourtant pas exempt de défauts (ce qui explique qu'il soit peu utilisé) : il faut générer une nouvelle clé pour chaque message et surtout transmettre cette clé sans risquer la moindre fuite.

On peut ici associer les sciences physiques lors de la génération de la clé (en lien avec des phénomènes physiques ayant un caractère aléatoire comme la désintégration radioactive ou l'agitation thermique). Il faudra alors prévoir une séance supplémentaire pour générer cette clé.

Activité de l'élève :

On construit la séquence comme une énigme à résoudre. On a retrouvé un brouillon de message qu'une personne projetait d'envoyer. Ce brouillon fait apparaître un certain nombre d'étapes du cryptage et est accompagné d'une table de transcodage. On se propose d'étudier la façon dont le cryptogramme est construit, de détailler ses étapes de façon à construire un algorithme de cryptage correspondant.

Un travail en petits groupes peut à nouveau être proposé. Tout comme dans l'activité précédente le niveau mathématique est susceptible de poser problème si l'on souhaite comprendre toutes les implications d'un codage de type Vernam : pourquoi la clé doit-elle être choisie aléatoirement et sa taille égale à celle du message ?
On peut y répondre en proposant de travailler de façon expérimentale et inciter à construire des contre-exemples (une clé de très petite taille que l'on répète). On pourra alors montrer que le cryptogramme devient sensible aux attaques fréquentielles.

On souhaite mettre en avant la structure algorithmique et ne pas être bloqué par des aspects techniques. On ne procédera donc pas systématiquement au passage sur la machine (tableur ou même programme).

3 / Synthèse sur la cryptographie (une heure)

3.1 La loi :

Le chiffrement des données est autorisé par l'article 30 de la loi 2004-575 du 21 juin 2004.

On trouvera de nombreux détails sur le site gouvernemental de l'ANSSI.

3.2 Les principes fondamentaux, les applications:

Une courte synthèse est à construire avec les élèves. Elle reprendra des aspects évoqués lors des activités. On pourra la structurer autour des notions suivantes :

- Cryptographie à clé secrète (symétrique) : le principe de Kerckhoffs,
- Cryptographie à clé publique (dite asymétrique) : on pourra sans difficulté illustrer son principe (échange de messages écrits sur des petits papiers au moyen de boîtes avec deux cadenas) sans pour autant détailler les fondements mathématiques trop difficiles à ce niveau.
- Les applications de la cryptographie : cartes bleues, cryptage Wifi, paramétrer le cryptage d'un routeur Wi-fi, envoyer des courriels en PGP, utiliser un navigateur web en mode sécurisé (SSL).
- Quel niveau de sécurité peut-on espérer ? Peut-on casser un cryptage ? Une nouvelle fois le niveau mathématique permettant de répondre à ce type de questions est élevé et dépend du cryptosystème étudié. Il s'agit seulement de sensibiliser l'élève à ce questionnement.

3.3 Les problèmes sociétaux induits.

La cryptographie induit de nombreux problèmes sociétaux que l'on peut évoquer sous la forme d'un débat, rythmé par des questions du type :

Pourquoi crypter ?

- Mettre en place un système d'authentification (sur l'Internet, les réseaux wifi, les téléphones portables ou sans fil) permettant de savoir qui est « au bout du fil ».
- Empêcher la copie de logiciels, de données audio ou vidéo (cryptage de type CSS par exemple).
- Sécuriser les données stratégiques d'une entreprise.
- Protéger sa vie privée (dossier médicaux, courriels ...)

Est-on libre d'utiliser, de développer, de diffuser des moyens de cryptologie ?

- La loi LCEN du 21 juin 2004 et son décret d'application du 2 mai 2007 (consolidé en 2010, notamment les articles 30, 31 et 36).
- Distinction selon les usages et le cryptosystème utilisé (symétrique, asymétrique, longueur de la clé, taille du groupe multiplicatif).
- La loi HADOPI 2 et la législation française sur la diffusion d'Internet sans fil.

Le chiffrement libre favorise-t-il les activités criminelles ?

- … mais son interdiction serait-elle de nature à limiter ces activités ?
- Comment agir contre un criminel qui sévit à partir d'un pays où la législation est différente ?

Bien distinguer les notions de légalité et d'impunité.

On s'appuie sur des études de situations que les élèves côtoient régulièrement et qui seront formulées sous forme de questions motivantes :

- Je souscris, de France, à un service basé aux Pays-Bas qui me fournit un accès par tunnel VPN (réseau privé virtuel) à divers serveurs de téléchargement (eux mêmes situés ailleurs). On m'assure que la liaison sera sécurisée au moyen d'un cryptage autorisé par la loi française. Si je télécharge par ce moyen des contenus protégés par un droit d'auteur est-ce que je suis dans l'illégalité ? Quels sont les risques de me faire repérer ?
- La société (néerlandaise) qui offre ce service est-elle dans l'illégalité ?
- Qu'en est-il si la société est localisée en Argentine ? (pays où la législation sur le droit d'auteur est beaucoup moins développée).

Démocratie et droit à la vie privée …

- Mise en place d'un système de vote électronique et mesure des risques associés.

4 / Des mini-projets

On oriente ensuite l'élève vers la réalisation d'un mini-projet, réalisé en petits groupes, autour du cryptage :

- Construire un programme en Python ou en Java cryptant et décryptant un message par substitution mono-alphabétique.
- Modéliser la machine Enigma avec un programme (cryptage, décryptage).
- Construire un programme cryptant et décryptant utilisant le chiffre de Vernam (clé secrète, loi de groupe, "Ou exclusif").
- Analyse fréquentielle d'un message crypté par substitution mono-alphabétique voire polyalphabétique (avec un peu plus d'arithmétique dans ce cas).
- Communiquer avec des courriels en PGP (installation, paramétrage des logiciels, création des clés publiques et privées, automatisation du chiffrement/déchiffrement).

5 / Références

Sites de Cryptologie:

- http://www.securite-informatique.gouv.fr/autoformations/cryptologie/co/Cryptologie.html
- http://www.cryptage.org (ce site comporte quelques publicités mais offre une bonne introduction à la cryptologie, au niveau des élèves).
- http://www.apprendre-en-ligne.net/crypto/menu/index.html

Livres:

- «Histoire des codes secrets» par Simon Singh. ISBN Poche: 978-2253150978
- «Les Codes secrets Décryptés». Didier Müller. Edition City. ISBN: 978-2-35288-544-3
- «Cryptographie : Théorie et pratique». Douglas Stinson; Edition Vuibert. ISBN: 978-2711748006

Textes de lois :

- La loi LCEN
  http://www.assemblee-nationale.fr/12/dossiers/economie_numerique.asp
  (articles 30, 31 et 36)
- Voir aussi :
  http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=LEGITEXT000005789847&dateTexte=#LEGIARTI000006421577
- L'ANSSI :
  http://www.ssi.gouv.fr/archive/fr/reglementation/regl_crypto.html
- La loi Hadopi 2 :
  http://www.legifrance.gouv.fr/affichTexte.do?idTexte=JORFTEXT000021208046&categorieLien=id
  (en particulier l'article 8 sur la sécurisation des accès wifi).

Courriels chiffrés:

- GNUPG: http://www.gnupg.org/ (installable sur Windows, MacOs, déjà présent sous Linux)
- Mozilla Thunderbird (http://www.mozillamessaging.com/fr/thunderbird/) et son extension Enigmail (http://enigmail.mozdev.org/home/index.php.html)

Aller plus loin; des ouvertures:

- Le hasard en sciences physiques; générer des nombres vraiment aléatoires:
  http://www2.cndp.fr/themadoc/radioactivite/caract_alea_ficheprof.htm
- La technologie OTP (One -Time Password): http://en.wikipedia.org/wiki/One-time_password
- La cryptographie quantique:
  http://www.geneve.ch/chancellerie/communiques/2007/20071011.asp ou bien
  http://www.unige.ch/communication/archives/2007/swissQuantum.html

4 Annexe : scénario, exercices …

1 / Le cryptage par substitution mono-alphabétique

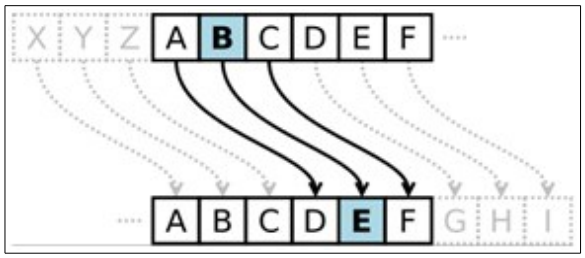
1.1 Le principe sur un exemple célèbre : le chiffre de César.

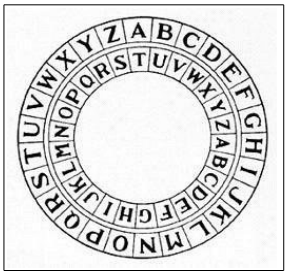
En cryptographie, le chiffrement par décalage, aussi connu comme le chiffre de César est une méthode de chiffrement très simple utilisée par Jules César dans ses correspondances secrètes.

Suétone, Vie de César LVI, 8

Extant et ad Ciceronem, item ad familiares domesticis de rebus, in quibus, si qua occultius perferenda erant, per notas scripsit, id est sic structo litterarum ordine, ut nullum uerbum effici posset: quae si qui inuestigare et persequi uelit, quartam elementorum litteram, id est D pro A et perinde reliquas commutet.

On possède enfin de César des lettres à Cicéron, et sa correspondance avec ses amis sur ses affaires domestiques. Il y employait, pour les choses tout à fait secrètes, une espèce de chiffre qui en rendait le sens inintelligible (les lettres étant disposées de manière à ne pouvoir jamais former un mot), et qui consistait, je le dis pour ceux qui voudront les déchiffrer, à changer le rang des lettres dans l'alphabet, en écrivant la quatrième pour la première, c'est-à-dire le D pour l'A, et ainsi de suite.





Ses caractéristiques :

- Ce système est mono-alphabétique.
- Les lettres gardent la même position au sein du cryptogramme.
- Il y a un nombre réduit de clés (le décalage) ce qui permet de toutes les tester et rend ce système peu sûr.
- Le message est par ailleurs sensible aux analyses fréquentielles[2] et donc facile à casser … si on sait dans quelle langue il est composé ! En revanche, si le message est composé dans une langue non-européenne et non connue à l'avance, le travail devient considérablement plus difficile.[3]

1.2 Approche avec un tableur

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Clef	6																
2																		
3	Message en clair	L	A	S	P	E	C	I	A	L	I	T	E	I	S	N	W	X
4	Message numérisé en ascii décalé	11	0	18	15	4	2	8	0	11	8	19	4	8	18	13	22	23
5	Message crypté en ascii décalé	17	6	24	21	10	8	14	6	17	14	25	10	14	24	19	2	3
6	Cryptogramme final	R	G	Y	V	K	I	O	G	R	O	Z	K	O	Y	T	C	D
7																		

On scinde bien chacune des étapes : le message clair, le message clair numérisé en ASCII (décalé de façon à ce que A = 0) le message crypté en ASCII et enfin le cryptogramme.

Voici les formules utilisées :

Clef	6
Message en clair	I
Message numérisé en ASCII décalé	=CODE(B3)-65
Message crypté en ASCII décalé	=MOD(B4+$B$1;26)
Cryptogramme	=CAR(B5+65)

2 L'analyse fréquentielle est une méthode particulière de *cryptanalyse* (tentative de déchiffrement sans avoir la clé).

3 L'usage de transformations (comme le « javanais » qui « pollue » la statistique d'emploi des lettres) peut aussi compliquer notablement la cryptanalyse du code.

1.3 Algorithme et programme en Python ou Java (pour un mini-projet).

Principe de l'algorithme :

Pour simplifier le travail et ne pas induire une faille de sécurité trop importante, on se limite dans un premier temps à un message tout en majuscules, sans accent, avec peu de séparateurs : on pourra ainsi utiliser le caractère @ pour coder un espace.

La clé correspond alors au décalage ; elle est comprise entre 1 et 25.

Chaque caractère est codé par son code ASCII auquel on a enlevé le code ASCII de "A" (65) de façon à ce que A soit codé par 0, B par 1…(ceci permet de clarifier la notion de modulo 26).

Principe du chiffrement : caractère crypté = caractère initial + k (modulo 26).

Principe du déchiffrement : caractère décrypté = caractère – k +26 (modulo 26).

Principe de l'algorithme de cryptage :

- Saisir le texte en clair.
- Saisir la clé de cryptage.
- Numériser le texte en retranchant le code ASCII de "A", ainsi A sera codé par 0, B par 1, ...
- Pour chaque caractère du texte en clair, appliquer le chiffrement et construire le cryptogramme (prévoir aussi le cas de l'espacement).
- Affichage du cryptogramme.

En Python 3.2.2 cela peut donner :

```
# Le chiffre de César simple en Python 3.2.2
import os
def Chiffrement(caractere,clef):
    if caractere==-33:                          # Cas du caractère d'espacement
        return -1                               # -1 + 65 = 64 : cela donnera le caractère @
    else:
        if caractere>=0 and caractere<=25:
            return (caractere+clef)%26          # Réalise le décalage dans la cas d'une majuscule non accentuée, retourne un entier
        else:
            return caractere                    # Pas de chiffrage en dehors des majuscules non accentuées et de l'espace
MessageClair = input('Texte à crypter ?')       # On demande le message à coder, c'est une chaîne de caractères.
clef = int(input('Clef de cryptage ?'))         # On demande la clef de cryptage, c'est un entier
MessageInter =[]                                # Cette variable list contiendra le message en ascii décalé de 65 (code ASCII de A).
MessageInter = [ord(i)-65 for i in MessageClair]
MessageCode=''                                  # chaîne de caractères qui contiendra le cryptogramme.
print ("Message initial en ascii décalé", MessageInter)
for i in MessageInter:
  MessageCode=MessageCode+chr(Chiffrement(i,clef)+65)
print ("Voici le message codé", MessageCode)
os.system("pause")
```

En JAVA (Ici Java's Cool) cela peut donner :

```
// Le chiffre de César avec Java's Cool
// Définition de la fonction de chiffrement
```

```
int Chiffrement(int car,int x){
        if (car==-33) return -1 ;                    // -1 + 65 = 64, cela donnera le caractère @
        if ((car>=0) && (car<=25) ) {                // Les majuscules non accentuées...
        return (car+x)%26;
        } else {
        return car;                                  // Pas de chiffrage si en dehors des majuscules non accen-
tuées
        }
}
void main(){
    String MessageClair;                             // Déclaration des variables
    int MessageInter [] ;
    MessageInter = new int [100];                    //Taille limitée à 100 pour le message à crypter
    int Clef,i;
    String MessageCode="";
    println("Texte à crypter ?");                    // Demande du message et de la clé de cryptage
    MessageClair=readString();
    println("Le message à crypter est"+ "\n"+ MessageClair);
    println("Clef de cryptage 1?");
    Clef =readInt();
        // Traitement
    println("La longueur du message à crypter est");
    println(MessageClair.length());
        // On sépare chaque caractère du message et on met son code ascii décalé de 65 dans le tableau
    for (i=0;i<=MessageClair.length()-1;i++){
        MessageInter[i]=(int)(MessageClair.charAt(i)-65);
        }
                                                     //Affichage pour vérification des codes ascii décalés avant le
cryptage.
    println("Le message non crypté en ascii décalé est");
    for(i=0;i<=MessageClair.length()-1;i++) {
        print(MessageInter[i]+" ");
        }
    println("\n");
                                                     // Cryptage du message et création du cryptogramme
    for (i=0;i<=MessageClair.length()-1;i++) {
        MessageCode=MessageCode+ (char)(Chiffrement(MessageInter[i],Clef)+65);
    }
        // Affichage du résultat.
    println("Le cryptogramme avec la clef" + "\n" + Clef + " est" + "\n" + MessageCode);
}
```

Variante : on pourra, pour des élèves rapides et performants, proposer un traitement plus complexe : coder les espaces et les signes typographiques avec le caractère @, remplacer les minuscules par les majuscules correspondantes, les accents par les lettres sans accent associées (on utilisera par exemple une table de transcodage).

Remarque : que se passe-t-il pour une clé égale à 13 ? (système nommé Rot 13, destiné à dissimuler les textes aux regards indiscrets)

1.4 Une alternative : le codage affine ; l'arithmétique en plus !

Le principe du chiffrement :

On ne décale plus les lettres d'un nombre fixe. On utilise maintenant une fonction affine pour réaliser ce décalage.

Appelons x le code ASCII (modifié de façon à ce que A = 0 ; B = 1) d'une des lettres du message en clair.

On chiffre cette lettre en calculant le reste de la division euclidienne par 26 de l'expression affine $ax + b$ (soit $ax + b$ mod 26).

Exemple : prenons comme exemple la fonction de cryptage f(x) = 17 x + 3 et pour lettre, la lettre G : son code ASCII est de 71 qui donne après modification 6.

f(6) = 105 = 4* 26 + 1. Le reste de la division est 1. Cette lettre G sera donc chiffrée par le nombre y = 1.

Le passage au tableur n'est pas plus compliqué, la clé est un couple (a;b).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Clefs	17	3													
2																
3	Message en clair	L	A	S	P	E	C	I	A	L	I	T	E	I	S	N
4	Message numérisé en ascii avec décalage	11	0	18	15	4	2	8	0	11	8	19	4	8	18	13
5	Message crypté en ascii avec décalage	8	3	23	24	19	11	9	3	8	9	14	19	9	23	16
6	Cryptogramme final	I	D	X	Y	T	L	J	D	I	J	O	T	J	X	Q

La formule de la ligne 3 devient simplement : =MOD(B4*$B$1+$C$1;26)

(la clé est entrée dans les cases B1 et C1)

Un déchiffrement plus délicat : l'arithmétique en secours, l'expérimentation en soutien !

Pour déchiffrer le message, il faut trouver l'antécédent x de y par l'application qui, à un entier x compris entre 0 et 25, associe le reste de $17x+3$ dans la division par 26.

On a donc: $17x+3\equiv y[26]$

Pour isoler x, nous sommes confronté à deux obstacles : le 3 et le 17.

Le problème du 3 est simple: $17x\equiv y-3[26]$ (On peut procéder autrement, en ajoutant 23 par exemple...).

Pour le 17, le problème est plus complexe (on cherche l'inverse de 17 modulo 26, c'est à dire un nombre c tel que: 17c admette 1 pour reste dans la division par 26).

De façon théorique, le théorème de Bachet-Bézout indique qu'il est possible de trouver c puisque 17 est premier avec 26. L'algorithme d'Euclide étendu permet alors de trouver une solution.

Mais on peut sans problème tenter expérimentalement de chercher un tel entier c ; deux lignes d'un tableur vont donner une solution :

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
17	8	25	16	7	24	15	6	23	14	5	22	13	4	21	12	3	20	11	2	19	10	1	18	9	0

23 convient (on a utilisé la formule `=MOD(17*A1;26)`)

L'équation peut alors se traiter: $17x\equiv y-3[26]$ équivaut à $x\equiv(y-3)*23[26]$ soit $x\equiv 23y-69[26]$.

On remarque aussi que $-69=26*(-3)+9$ et donc $-69\equiv 9[26]$. Une clé de décryptage est donc $(23;9)$.

Les élèves curieux pourront trouver d'autres solutions: combien y en a-t-il ? Quelle est leur forme?

Tous les couples (a;b) sont-il possibles pour crypter ? Combien de couples sont alors possibles?

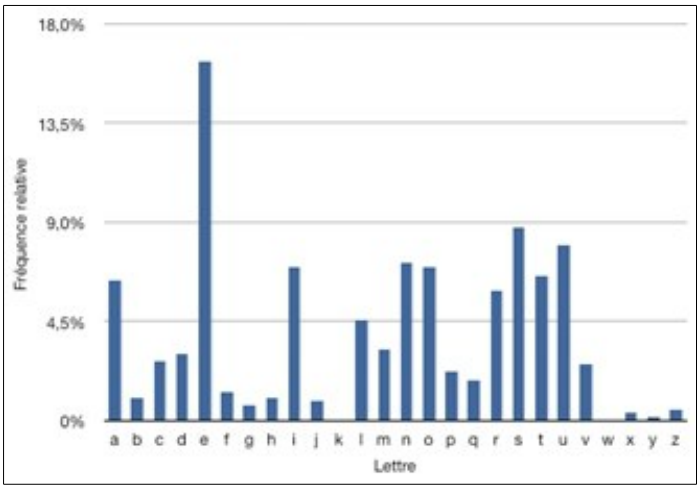
Une réponse partielle avec un tableur :

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Clefs	4	5															
2																		
3	Message en clair	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
4	Message numérisé en ascii avec décalage	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
5	Message crypté en ascii avec décalage	5	9	13	17	21	25	3	7	11	15	19	23	1	5	9	13	17
6	Cryptogramme final	F	J	N	R	V	Z	D	H	L	P	T	X	B	F	J	N	R
7																		

Cette évolution du code de César est-elle beaucoup plus sûre ? Combien existe-t-il de clés possibles ? (312...)

Autant d'interrogations permettant de moduler le niveau de cette partie…

2 / Analyse fréquentielle



On a analysé ci-dessus la fréquence d'apparition de lettres dans un texte en français.

Un chiffrement de César ne fait que décaler cette distribution, ce qui rend son cassage aisé par analyse fréquentielle … pour autant que l'on sache dans quelle langue le message est écrit.

Un exemple d'analyse fréquentielle est visible ici :

http://www.cryptage.org/analyse-frequentielle.html

Remarques :

- Que donne cette analyse avec le roman de Perec «La disparition»?
- L'analyse fréquentielle est-elle pertinente pour des textes de petite taille ?
- La distribution des fréquences varie-t-elle d'une langue à l'autre?
- Un message crypté selon le système de Vernam est-il sensible à ce type d'analyse ?

3 / Le code de Che Guevara (un code à clé secrète de type Vernam)

3.1 Première étape : on utilise une substitution de lettres (par des chiffres) selon le tableau suivant (appelé aussi table de transcodage)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
6	38	32	4	8	30	36	34	39	31	78	72	70	76	9	79	71	58	2	0	52	50	56	54	1	59

Le mot CRYPTAGE devient donc : 32 58 1 79 0 6 36 8

3.2 Seconde étape : on découpe le message obtenu en blocs de cinq chiffres.

CRYPTAGE devient alors : 32581 79063 68

3.3 Troisième étape : une addition modulo 10

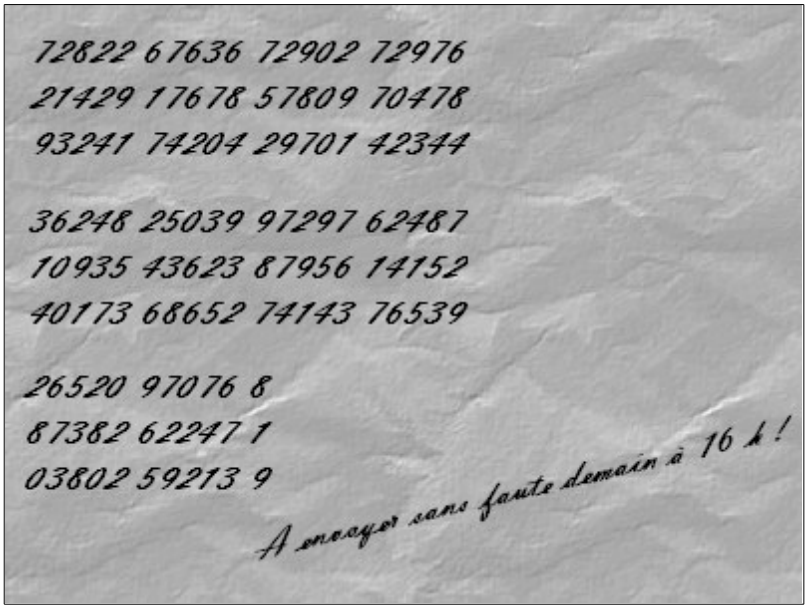
On construit une clé secrète aléatoire, de taille égale à celle du message, que l'on découpe en bloc de 5 chiffres.

Prenons comme clé 12681 (pour le premier bloc). On additionne modulo 10 cette clé aux blocs précédents :

32581 + 12681 = 44162 etc.

3.4 Les documents élèves : pour motiver une recherche.

Document 1 : un brouillon de codage ?



Document 2 : une table de transcodage.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
6	38	32	4	8	30	36	34	39	31	78	72	70	76	9	79	71	58	2	0	52	50	56	54	1	59

Comment à partir de ces deux documents, reconstituer les étapes et construire un algorithme ?

Adresse à consulter : http://www.cryptage.org/chiffre-che-guevara.html

http://mauro-israel.over-blog.com/article-principes-simples-de-cryptographie-expliques-72433552.html

4 / Le principe de Kerckhoffs

Le concept de clé : c'est un paramètre de l'algorithme de chiffrement. Il peut donc être changé sans modifier l'algorithme. Un même algorithme peut alors être employé par plusieurs personnes qui utilisent des clés différentes.

Par exemple, pour le chiffre de César, on modifie le nombre de positions par lequel on décale chaque lettre dans l'alphabet. En ce sens, ce nombre peut être considéré comme une clé.

Le principe de Kerckhoffs s'énonce ainsi : « La sécurité d'un cryptosystème à clé secrète doit résider dans le secret de la clé. Les algorithmes utilisés doivent pouvoir être rendus publics »

Voir ici pour plus de détails :

http://www.securite-informatique.gouv.fr/autoformations/cryptologie/co/cryptologie_CH01_SCH01_U02.html

5 / Le chiffrement à clé publique

Le chiffrement à clé publique (ou chiffrement asymétrique) est un système de chiffrement reposant sur deux clés (et non une seule et même clé comme pour les cryptosystèmes à clé secrète), la **clé publique** et la **clé privée**. Le principe est assez simple :

- chacun publie sa propre clé publique à tous ses correspondants
- pour écrire à M. X, on chiffre le message avec la clé publique de X qui déchiffre avec sa clé privée (qu'il est le seul à posséder et qui est protégée par un long mot de passe ou une phrase secrète).

En somme,

Le système de chiffrement (asymétrique) ne nécessite aucune transmission de mot de passe !

Le **transfert d'informations par courrier électronique chiffré** (mail crypté) est assez facile à mettre en œuvre avec les logiciels libres GNUPG (http://www.gnupg.org) et Mozilla Thunderbird (http://www.mozillamessaging.com/fr/thunderbird).

Le logiciel GNUPG (GNU Privacy Guard, en abrégé GPG), sert en « sous-main » du logiciel de courrier et implémente les algorithmes dits de chiffrement asymétrique ; selon le principe de Kerkhoffs, ces algorithmes ne sont pas secrets (ils sont déduits des algorithmes RSA et DES).

Le logiciel de courrier Mozilla Thunderbird, simple d'emploi, est très stable et fort répandu. Au moyen d'une extension (Enigmail, ici : http://enigmail.mozdev.org/home/index.php.html), l'usage du chiffrement se fait facilement et sans perte de temps.

Enigmail permet en outre d'attacher des **signatures numériques** aux messages, qui contribuent à empêcher qu'une tierce personne usurpe votre identité pour envoyer des messages.