Prologin 2013

Concours national d'informatique

Épreuve écrite d'algorithmique
Bordeaux, Louvain-la-Neuve & Strasbourg

Samedi 2 mars 2013

Enigma

1 Préambule

Bienvenue à **Prologin**. Ce sujet est l'épreuve écrite d'algorithmique et constitue la première des trois parties de votre épreuve régionale. Sa durée est de 3 heures. Par la suite, vous passerez un entretien (20 minutes) et une épreuve de programmation sur machine (3 heures 30).

Conseils

- Lisez bien tout le sujet avant de commencer.
- **Soignez la présentation** de votre copie.
- N'hésitez pas à poser des questions.
- Si vous avez fini en avance, relisez bien, ou préparez votre présentation pour l'entretien.
- N'oubliez pas de passer une bonne journée.

Remarques

- Le barème est donné à titre indicatif uniquement.
- Indiquez lisiblement vos nom et prénom, la ville où vous passez l'épreuve et la date en haut de votre copie.
- Tous les langages sont autorisés, veuillez néanmoins préciser celui que vous utilisez.
- Ce sont des humains qui lisent vos copies : laissez une marge, aérez votre code, ajoutez des commentaires (**seulement** lorsqu'ils sont nécessaires) et évitez au maximum les fautes d'orthographe, sinon ça va barder.
- Le barème récompense les algorithmes les plus efficaces : écrivez des fonctions qui trouvent la solution le plus rapidement possible.
- Si vous trouvez le sujet trop simple, relisez-le, réfléchissez bien, puis dites-le-nous, nous pouvons ajouter des questions plus difficiles.

2 Sujet

Introduction

La machine Enigma a cinq composantes principales :
- un disque d'entrée, tableau où l'on peut relier certaines lettres à d'autres ;
- trois rotors [1], représentant chacun une permutation de l'alphabet ;
- un réflecteur, dont on dira qu'il convertit A en Z, B en Y, C en X et ainsi de suite.

Dans le problème, on ne considérera qu'un seul rotor, et tous les messages seront en capitales. Quand une lettre est tapée, l'information traverse le circuit et une autre lettre est éclairée, représentant la lettre chiffrée, puis le rotor tourne d'un cran.

Exemple. On configure le disque d'entrée pour qu'il relie W à A, T à S et O à N, ce qui signifie que A sera changé en W et inversement, que S sera changé en T et inversement, etc. Si le rotor contient la permutation suivante :

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
YDUHNMFCRKXZLJSOVTIWBAEPQG
```

alors en tapant A, on aura :

$$A \xrightarrow{\text{disque d'entrée}} W \xrightarrow{\text{rotor}} E \xrightarrow{\text{réflecteur}} V \xrightarrow{\text{rotor à l'envers}} Q \xrightarrow{\text{disque d'entrée à l'envers}} Q.$$

Donc Q sera la première lettre du message chiffré, puis le rotor deviendra :

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
BCDEFGHIJKLMNOPQRSTUVWXYZA
DUHNMFCRKXZLJSOVTIWBAEPQGY
CTGMLEBQJWYKIRNUSHVAZDOPFX
```

car « tourner d'un cran » revient à ajouter une lettre, appliquer la permutation du rotor, puis enfin retirer une lettre. Si l'on tape alors U, le parcours du circuit sera le suivant :

$$U \xrightarrow{\text{disque d'entrée}} U \xrightarrow{\text{rotor}} Z \xrightarrow{\text{réflecteur}} A \xrightarrow{\text{rotor à l'envers}} T \xrightarrow{\text{disque d'entrée à l'envers}} S.$$

Deux particularités de ce circuit :
- une lettre ne pourra jamais être chiffrée par elle-même (cela a causé bien des failles) ;
- pour un instant donné, si x donne y, alors y donne x.

> J'étais tombé sur ce message, et j'ai pensé : « Il manque quelque chose. Qu'est-ce que ce type a fait, son message ne comporte pas un seul L. » En fait, ce gars devait envoyer un message bidon et il avait juste appuyé sur le L, la dernière lettre du clavier. C'était donc la seule lettre qui ne s'était pas allumée. C'était le plus long message en clair qu'on avait eu, et ça nous a donné la composition du rotor. C'est le genre de choses que nous étions entraînés à faire. Chercher instinctivement quelque chose qui n'allait pas ou quelqu'un qui avait fait une bêtise, qui n'avait pas suivi les règles.
>
> – Mavis Lever

1. Enigma est donc une *three-rotor machine*, pas une *two-ring machine*.

Question 1 (2 points)

> Écrire une fonction qui prend en argument un message chiffré et détecte une lettre de l'alphabet qui n'y figure pas (si elle existe ; sinon, retourner le caractère espace).

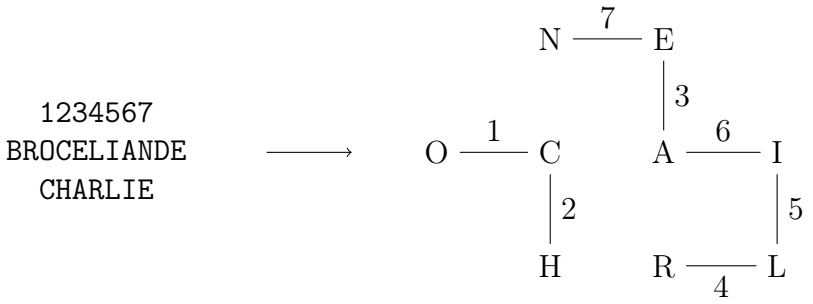
Question 2 (3 points)

> Écrire une fonction qui, à partir d'une configuration du rotor, retourne sa nouvelle configuration après l'avoir tourné d'un cran.

Question 3 (3 points)

> Sachant qu'une lettre n'est jamais codée par elle-même, écrire une fonction qui prend en argument un message m et un chiffré c et détermine les positions convenables [2] pour m dans c, sous la forme d'une liste d'entiers.
> **Exemple.** Si votre fonction est appelée pour chercher `CHARLIE` dans `BROCELIANDE`, elle retournera `[0, 2]` car par exemple, la position n° 3 ne convient pas :
>
> ```
> BROCELIANDE
>    CHARLIE
> ```
>
> car C ne peut pas être codé par lui-même.

Lorsqu'une position pour le message est satisfaisante, il convient de tracer un diagramme appelé *menu*, représentant les paires de lettres liées. Ainsi, la position n° 2 donne le menu suivant :



Question 4 (3 points)

> Écrire une fonction qui à partir d'un chiffré, d'un message et d'une position crée le menu, représenté en mémoire par une matrice 26×26 dont les éléments sont des listes (il peut y avoir plusieurs arêtes entre deux sommets).

Question 5 (4 points)

> Le fonctionnement de la machine Enigma peut être séparé en trois parties : disque d'entrée, rotor + réflecteur + rotor inverse, disque d'entrée inverse. À partir de la configuration d'un rotor [3], écrire un programme qui détermine la permutation rotor + réflecteur + rotor inverse.

2. C'est-à-dire, non impossibles.
3. Une permutation de l'alphabet.

Question 6 (5 points)

> Sachant que le disque d'entrée ne permet que d'échanger des lettres deux à deux [4], si l'on part d'une paire possible pour le disque d'entrée, à l'aide de votre tableau à la question précédente, vous pouvez déduire d'autres paires du disque d'entrée. Écrire un programme qui, à partir d'une position convenable pour un message m dans un chiffré c, teste si une paire aboutit à une contradiction [5].

Cet algorithme, qu'implémenta Alan Turing sur la Bombe, fonctionne mieux lorsque le menu comporte des cycles.

Question 7 (3 points)

> Écrivez un programme qui détermine si votre menu comporte un cycle.

Vous pouvez attaquer les questions suivantes si et seulement si vous avez compris toutes les questions précédentes.

Question bonus 8 (2 points)

Écrivez une fonction qui détermine le nombre de cycles du menu de longueur supérieure ou égale à 3.

Question bonus 9 (4 points)

Vrai ou faux ?

- Le nombre « EINS » apparaissait dans 90 % des messages allemands.
- La plupart du temps, les officiers de l'armée de l'air allemande choisissaient « HITLER » pour configurer leur Enigma.
- Les Anglais demandaient de temps en temps à la Royal Air Force de miner certaines zones spécifiques de la mer du Nord pour que les Allemands s'échangent des messages contenant « MER DU NORD » en chiffré.
- Les Allemands pensaient qu'Enigma était incassable.

Le sujet est sur 23 points, et les questions bonus rapportent au total 6 points, plus 1 point de présentation.

4. On parle aussi de *transpositions*.
5. En partant d'une paire (x, y), on ne peut pas aboutir à une paire (x, z) avec $z \neq y$.