ALGORITHMIQUE DES NOMBRES
TRAVAUX PRATIQUES

1. Algorithme du PGCD étendu

Soient x et y deux entiers naturels, Δ leur PGCD. L'identité de Bâchet-Bézout affirme l'existence de deux entiers relatifs u et v tels que :

$$ux + vy = \Delta, \quad |u| \leq y, \quad |v| \leq x. \tag{1}$$

1.1. **Implantation.**

(1) Écrire une fonction récursive pour déterminer les entiers u, v et Δ vérifiant (1).
(2) Écrire une fonction itérative pour faire le même travail.
(3) Comparer les performances de ces deux fonctions.

1.2. **Réflexion.** Soit n un entier premier avec 10. Déterminer l'unique couple (x, y) de résidus modulo n, solution du système :

$$\begin{aligned} 25x + 10y &\equiv 1811 \pmod n \\ 30x + 4y &\equiv 1777 \pmod n \end{aligned}$$

1.3. **Culture.** Le problème 26 du chapitre III du *Sunzi suanjing*, un texte du mathématicien Sun Zi qui aurait vécu autour du cinquième siècle est énoncé en ces termes:

Nous avons des choses dont nous ne connaissons pas le nombre ;

- *si nous les comptons par paquets de trois, le reste est 2 ;*
- *si nous les comptons par paquets de cinq, le reste est 3 ;*
- *si nous les comptons par paquets de sept, le reste est 2.*

Combien y a-t-il de choses ?

(1) Trouvez une solution!
(2) La solution est-elle unique ?

1.4. **Challenge.** Si vous avez répondu aux questions précédentes, vous pouvez vous attaquer au challenge :

Ecrire une version optimisée du calcul du PGCD deux deux entiers non signés de 64 bits.

Date: Avril 2006.

2. Pivot de Gauss

L'algorithme du pivot de Gauss permet de calculer le rang d'une matrice. Une méthode particulièrement efficace dans le cas des matrices sur le corps à deux éléments. De plus, une modification de l'algorithme permet de calculer une base du noyau d'une application linéaire.

```
REDUCTION( m:matrice, nbl, nbc:entier)
variable i, j, r: entier;
debut
   r := nbl;
   i := 0;
   tant que ( i < r )
      j := pivot( m, i, nbc);
      si ( j >=0 )
          alors
            pour chaque k, i < k <  r
               si ( m[k][j] ) alors
                addligne(m, k, i);
               fsi
            fdp
            i := i + 1;
          sinon
            r := r - 1;
            excligne(m, i, r)
          fsi
   ftq
   retourner r;
fin
```

Figure 1. Réduction de Gauss.

(1) Ecrire un programme pour lire une matrice binaire dans un fichier texte. Vous trouverez quelques des exemples de matrices à l'adresse http://langevin.univ-tln.fr/CDE/I43/
(2) Ecrire une fonction pour calculer le rang d'une matrice.
(3) Modifier le calcul de rang pour déterminer une base du noyau d'une application linéaire.