Agorithme Euclidien

TP-4, Module I41, Licence Informatique
université du sud Toulon-Var

Avril 2007

Pour cette séance de TPs, on utilise des nombres signés de 64 bits :

```
typedef long long nombre;
```

1 Analyse de l'algorithme d'Euclide.

1. Implanter une fonction `nombre pgcd( nombre a, nombre b)` qui calculer et retourne le PGCD des entiers a et b.
2. On note $i(a,b)$ le nombre d'itération de l'algororithme d'Euclide engendré par le calcul de $\textsc{pgcd}(a,b)$. Implanter une fonction `nombre pgcdcpt( nombre a, nombre b, int *cpt)` qui retourne $\textsc{pgcd}(a,b)$ et calcule le nombre d'itérations.
3. On note $\mu(a)$ la moyenne des $i(a,b)$ et $m(a)$ le nombre maximal d'itérations $(1 \leq b \leq a)$. Représenter graphiquement ces fonctions.
4. Déterminer des estimations empiriques des fonctions $m(a)$ et $\mu(a)$.
5. Déterminer deux entiers a et b tel que $i(a,b) = 64$.

2 Euclide Étendu : version récursive.

1. Implanter une fonction récursive :

   ```
   nombre pgcd( nombre a, nombre b, nombre *u, nombre *v)
   ```

   qui calcule des entiers u et v tels que : $\textsc{pgcd}(a,b) = au + bv$
2. Modifier cette fonction pour tracer les quotients et restes successifs engendrés par l'algorithme d'Euclide.