ALGORITHME D'EUCLIDE

On rappelle l'algorithme d'Euclide :

$$r_k = q_{k+1}\, r_{k+1} + r_{k+2}\ ,\ k \geq 0$$

$$r_0 = a \in \mathbb{N}\ ,\ r_1 = b \in \mathbb{N}^*\ ,\ 0 \leq r_{k+2} < r_{k+1} \quad \text{pour } k \geq 0$$

Le dernier reste non nul dans l'algorithme est d = pgcd(a,b).

Sous Maple, le quotient dans la division euclidienne de a par b se calcule par iquo(a,b) et le reste par irem(a,b).

1) Algorithme, version itérative.
Ecrire sous Maple, à l'aide d'une boucle une suite de commande calculant d .
Rajouter un test pour contrôler que a est dans $\mathbb{N}$ et b dans $\mathbb{N}^*$ (type(a,integer) rend vrai si a est entier et faux sinon et type(a,posint) rend vrai ssi a est dans $\mathbb{N}^*$).

2) Ecrire à l'aide d'un test un algorithme récursif calculant pgcd(a,b) en utilisant

$$\text{pgcd}\,(a,b) = \begin{cases} \text{pgcd}(b,a) \text{ si } a < b \\ b \text{ si } b \mid a \quad \text{(condition d'arrêt)} \\ \text{sinon pgcd}(b,r) \text{ où } a = bq + r \text{ est la division euclidienne de } a \text{ par } b \text{ avec } 0 < r < b \end{cases}$$

3) On souhaite trouver une relation de Bezout, c.a.d des entiers u et v tels que $ua + bv = d$.
On construit pour cela des suites $(u_k)_{k \geq 0}$ et $(v_k)_{k \geq 0}$ telles que $u_k\, a + v_k\, b = r_k$. Déterminer u_0, u_1, v_0 et v_1 puis établir les relations de récurrence reliant u_{k+2} , u_{k+1} et u_k d'une part, et v_{k+2} , v_{k+1} et v_k d'autre part.
Adapter l'algorithme itératif pour calculer au fur et à mesure les u_k et v_k et déterminer des réels u et v tels que $ua + bv = d$.

4) Adapter l'algorithme récursif, pour construire des relations $\alpha_k\, r_k + \beta_k\, r_{k+1} = d$: lorsque $r_{n+1} = 0$, on a $d = r_n$ d'où $\alpha_n = 1$ et $\beta_n = 0$. Etablir les relations de récurrence $\alpha_{k-1} = \beta_k$ et $\beta_{k-1} = \alpha_k - q_k\, \beta_k$. Pour $k = 0$, on obtient $\alpha_0\, r_0 + \beta_0\, r_1 = d$ c.a.d $\alpha_0 a + \beta_0\, b = d$.