

# TP3 en Algorithmique

10 février 2009

1. *Algorithme des carrés successives.* Implémenter une fonction double `puiss(double base, unsigned int exp)` pour calculer la puissance  $\text{base}^{\text{exp}}$ . Utiliser l'algorithme des carrés successives décrit à grands traits ci-dessus. En particulier, n'utiliser pas la fonction `pow` de `math.h`.

Pour calculer  $b^e$ , on se base sur l'écriture binaire de  $e$

$$e = \sum_{i=0}^k \varepsilon_i 2^i$$

où  $\varepsilon_i = 0$  ou  $1$ . Donc

$$b^e = b^{\sum_i \varepsilon_i 2^i} = \prod_i (b^{2^i})^{\varepsilon_i},$$

où chaque élément du produit est le carré du précédent.

2. Écrire des fonctions de lecture et affichage de polynômes à coefficients entiers, codés comme des tableaux, de degré maximale `MAX` de votre choix. Écrire une fonction

```
multpol(int res[], int *degres, int a[], int b[], unsigned int dega,
        unsigned int deg b)
```

qui remplit dans `res` et son degré `degres` le résultat de la multiplication des deux polynômes `a` et `b` (de degrés `dega` et `degb`). Si le degré de la multiplication est plus élevé que `MAX`, on tranche les coefficients jusqu'à `MAX`, en fixant `degres = MAX` (en fait, on implémente la multiplication dans l'anneau  $\mathbb{Z}[X]/(X^{\text{MAX}+1})$ ). On suppose `res[]` un tableau déjà alloué.

Enfin, implémenter la fonction `puisspol` qui calcule la puissance d'un polynôme, en utilisant `multpol`.

3. Écrire une fonction

```
double maximum(double tab[], int taille)
```

renvoyant le maximum nombre dans `tab`.

4. Écrire une fonction double `second_max(double tab[], int taille)` renvoyant le deuxième maximum nombre dans `tab`. Par exemple sur `tab = {1, 3, 3, 5, 7}`, `second_max` renvoie 5.
5. Écrire une fonction

```
void maxmin(double tab[], unsigned int taille, unsigned int *max, unsigned
            int *min)
```

qui mémorise dans `max` (resp. dans `min`) l'index de la cellule de `tab` avec la valeur maximale (resp. minimale). Cherchez une fonction qui fait  $3 \cdot \text{taille} / 2$  comparaisons de valeurs (conseil : calculer `max` et `min` en même temps. Plutôt que comparer chaque élément de `tab` avec `max` et puis `min`, à un coût de 2 comparaisons par élément, comparer les éléments de `tab` par paires...).