

## FFT

1. Rappeler la définition de la base canonique de  $\mathbb{F}_2[X]$
2. En déduire que multiplier deux entiers en binaire revient à multiplier deux polynômes sur  $\mathbb{F}_2$ .
3. Montrer que pour tout  $P$ , on peut écrire  $P(X) = Q_1(X^2) + XQ_2(X^2)$ .

On définit la transformée de fourier (DFT) d'un polynôme de coefficients  $(x_k)_{0 \leq k \leq N}$  par le vecteur des

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} nk}.$$

4. Montrer que

$$X_k = \sum_{m=0}^{N/2-1} x_{2m} e^{-\frac{2\pi i}{N/2} mk} + e^{-\frac{2\pi i}{N} k} \sum_{m=0}^{N/2-1} x_{2m+1} e^{-\frac{2\pi i}{N/2} mk} = E_k + e^{-\frac{2\pi i}{N} k} O_k \quad \text{pour } k = 0, \dots, \frac{N}{2} - 1,$$

5. En cherchant une expression de  $X_{k+\frac{N}{2}}$  utilisant la périodicité de l'exponentielle complexe, en déduire un algorithme calculant la DFT en  $\mathcal{O}(N \log N)$ .
6. Montrer que  $DFT(P * Q) = DFT(P) \otimes DFT(Q)$ , la deuxième expression étant le produit point à point.
7. En déduire un algorithme calculant le produit de deux polynômes de degré  $N - 1$  en  $\mathcal{O}(N \log N)$

**Soupe d'hier et d'aujourd'hui.**

Dans cet exercice on va chercher une méthode efficace pour calculer le  $n$ -ème nombre de Fibonacci  $F_n$ .

1. Donner un algorithme calculant  $F_n$ , sans questions d'efficacité.
2. Montrer que si  $n \geq 1$  :

$$\begin{pmatrix} F_n \\ F_{n+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n \begin{pmatrix} F_0 \\ F_1 \end{pmatrix}$$

3. Montrer que  $\mathcal{O}(\log n)$  multiplications matricielles suffisent à calculer  $F_n$
4. Montrer que chaque calcul implique  $\mathcal{O}(n)$  bits.
5. Conclure que le temps de calcul de la procédure est en  $\mathcal{O}(n^2 \log(n))$ .
6. En supposant que 2 entiers sur  $n$  bits peuvent être multipliés  $\mathcal{O}(M(n))$ , peut on prouver que la procédure nécessite un temps en  $\mathcal{O}(M(n))$  ?

### Tset

Des capteurs équipés d'un émetteur/récepteur radio sont déployés sur une région plane. La position de chaque capteur est identifiée par ses coordonnées cartésiennes  $(x, y)$ . Les capteurs doivent transmettre des données à une station centrale située aux coordonnées  $(0, 0)$ , qui est elle aussi équipée du même émetteur/récepteur radio. Tous les émetteurs/récepteurs possèdent une portée effective  $r$ , ce qui signifie que deux radios peuvent communiquer si et seulement si leur distance est au plus  $r$ . Cependant, les capteurs et la station de base forment un réseau : ainsi, deux dispositifs qui ne peuvent pas communiquer directement peuvent tout de même communiquer indirectement par l'intermédiaire d'un ou plusieurs autres dispositifs servant de relais. Voir l'exemple ci-dessous.

Nous avons quatre capteurs  $a, b, c, d$ , ainsi qu'une station de base  $S$ . Les cercles représentent la portée de chaque radio. Le capteur  $a$  peut communiquer directement avec la station de base, et le capteur  $b$  peut également communiquer avec la station de base grâce à  $a$ , qui agit comme relais. Les capteurs  $c$  et  $d$  peuvent communiquer entre eux mais pas avec  $S$ .

1. Faites un dessin de la situation.
2. Écrire un algorithme  $\text{CHECK-CONNECTIVITY}(X, Y, r)$  qui, étant données les coordonnées de tous les capteurs stockées dans les tableaux  $X$  et  $Y$  tels que le capteur  $i$  se trouve aux coordonnées  $(X[i], Y[i])$  ainsi que la portée de communication  $r$ , retourne **true** si tous les capteurs peuvent transmettre leurs données à la station de base, ou **false** si un ou plusieurs capteurs ne le peuvent pas. Analyser également la complexité de votre solution.
3. Écrire un algorithme  $\text{MINIMAL-CONNECTIVITY-RANGE}(X, Y, t)$  qui, étant données les coordonnées de tous les capteurs stockées dans les tableaux  $X$  et  $Y$ , ainsi qu'un seuil de précision  $t$ , retourne la portée radio minimale  $r$  garantissant une connectivité complète. Le rayon obtenu  $r$  peut être une approximation du rayon minimal réel  $\bar{r}$  à une précision  $t$  près, c'est-à-dire telle que

$$|r - \bar{r}| \leq t.$$

Analyser la complexité de  $\text{MINIMAL-CONNECTIVITY-RANGE}$ .

### Plus petit produit de chiffres

Étant donné un entier positif  $n$ , et un entier  $t$ , donnez un algorithme renvoyant le plus petit nombre plus grand que  $n$  dont aucun des chiffres n'est 0 et dont le produit des chiffres est divisible par  $t$ . S'il n'en existe pas, renvoyez  $-1$ .