Corrigé : détection d'erreurs par codage CRC (X psi 2003)

Partie I. Bit de parité

Question 1.

```python
def ou_exclusif(x, y):
    return (x + y) % 2
```

Question 2. On peut calculer b_n à l'aide de la formule : $b_n = b_0 \oplus b_1 \oplus \cdots \oplus b_{n-1}$:

```python
def bit_parite(b):
    return sum(b) % 2
```

Si le message reçu contient un nombre impair de bits égaux à 1, il y a eu un nombre impair d'erreurs de transmissions, donc au moins une ! En revanche, un nombre pair d'erreurs de transmissions ne sera pas détecté.

Partie II. Le codage CRC

Question 3.

```python
def degre(b):
    k = 0
    while k < len(b) and b[k] == 0:
        k += 1
    return len(b) - 1 - k
```

Question 4.

```python
def plus(b, c, i, j, l):
    for k in range(l):
        b[i+k] = ou_exclusif(b[i+k], c[j+k])
```

Question 5.

a. Si le mot a été transmis sans erreur, le message reçu est le polynôme $\mathrm{T(X)} = \mathrm{X}^k\mathrm{P(X)} \oplus \mathrm{R(X)}$; or celui-ci est par définition divisible par G(X) ; ainsi, $\big(\mathrm{X}^k\mathrm{P(X)} \oplus \mathrm{R(X)}\big) \bmod \mathrm{G(X)} = 0$.

b. Réciproquement, si on note $\widetilde{\mathrm{T}}(\mathrm{X})$ le polynôme associé au message reçu, posons $\mathrm{E(X)} = \mathrm{T(X)} \oplus \widetilde{\mathrm{T}}(\mathrm{X})$; le message est donc transmis sans erreur si et seulement si $\mathrm{E(X)} = 0$. Or il est tout à fait possible d'avoir $\mathrm{E(X)} \neq 0$ sans que l'erreur soit détectée ; il suffit que E(X) soit divisible par G(X). On verra néanmoins qu'un choix judicieux de G(X) rend cette situation très improbable.

Question 6.

a. Puisque G(X) divise T(X), si G(X) ne divise pas E(X), il ne divise pas non plus $\widetilde{\mathrm{T}}(\mathrm{X})$, et donc $\widetilde{\mathrm{T}}(\mathrm{X}) \bmod \mathrm{G(X)} \neq 0$; l'erreur est détectée.

b. Une erreur sur un seul bit correspond à $\mathrm{E(X)} = \mathrm{X}^i$ avec $i \in [\![0, n+k]\!]$; si G(X) n'est pas un monôme, E(X) n'est pas divisible par G(X) et l'erreur est détectée.

c. Supposons que G(X) soit divisible par $(\mathrm{X}+1)$, et soit E(X) une erreur non détectée par le CRC. Alors G(X) divise E(X) et donc $(\mathrm{X}+1)$ aussi. On en déduit que 1 est racine de E(X) : $\mathrm{E}(1) = 0$. Mais ceci ne peut avoir lieu que si l'erreur contient un nombre pair de 1. Ainsi, toute erreur portant sur un nombre impair de bits est détectée.

Question 7.

a. Un paquet d'erreurs de longueur ℓ correspond à un polynôme $E(X) = X^{i+\ell-1} + \cdots + X^i = X^i F(X)$ avec $\deg F = \ell - 1$. Supposons que $G(X)$ divise $E(X)$. Si le coefficient constant de $G(X)$ n'est pas nul, $G(X)$ est premier avec X^i donc $G(X)$ divise $F(X)$. Puisque $\deg G(X) = k$, on a : $\ell - 1 \geqslant k$, soit $\ell > k$. En contraposant, on en déduit que tout paquet d'erreur de longueur $\ell \leqslant k$ est détecté.

b. Un paquet d'erreurs de longueur $k+1$ correspond à un polynôme $E(X) = X^{i+k} + \cdots + X^i = X^i F(X)$ avec $\deg F = k$. Si cette erreur n'est pas détectée, $G(X)$ divise $F(X)$, et puisqu'ils ont même degré, $F(X) = G(X)$. Il y a donc un seul paquet d'erreurs non détecté parmi les 2^{k-1} possibles (correspondants au choix des coefficients de $X^{i+1}, X^{i+2}, \ldots, X^{i+k-1}$ dans $E(X)$), donc une probabilité égale à $\dfrac{1}{2^{k-1}}$.

c. Un paquet d'erreurs de longueur $k+p$, avec $p \geqslant 2$, correspond à un polynôme $E(X) = X^{i+k+p-1} + \cdots + X^i = X^i F(X)$, avec $\deg F = k + p - 1$. Si cette erreur n'est pas détectée, $F(X) = G(X)Q(X)$, avec $\deg Q = p - 1$. De plus, X ne divise pas $F(X)$, donc le coefficient constant de $Q(X)$ n'est pas nul. Ainsi, $Q(X) = X^{p-1} + \cdots + 1$; ce qui donne 2^{p-2} polynômes possibles. La probabilité que cette erreur ne soit pas détectée est donc égale à : $\dfrac{2^{p-2}}{2^{k+p-2}} = \dfrac{1}{2^k}$.

Remarque. Un polynôme générateur souvent utilisé est le CRC-16 : $G(X) = X^{16} + X^{15} + X^2 + 1 = (X+1)(X^{15} + X + 1)$. D'après ce qui précède ; les erreurs en nombre impair sont détectées, tous les paquets d'erreurs de longueur inférieure ou égale à 16 sont détectés ; la probabilité de détecter un paquet d'erreurs de longueur 17 est égale à $1 - \dfrac{1}{2^{15}} \approx 99,997\,\%$; la probabilité de détecter un paquet d'erreurs de longueur supérieure ou égale à 18 est égale à $1 - \dfrac{1}{2^{16}} \approx 99,998\,\%$.

Question 8.

a. Nous allons effectuer les calculs dans un tableau auxiliaire c correspondant à un polynôme $C(X)$ initialement égal au polynôme $X^k P(X)$, et tant que $\deg C > k$, on remplace $C(X)$ par $C(X) \oplus X^{\deg C - k} G(X)$.

```
def crc(b, g):
    n, p = len(b), len(g)
    k = degre(g)
    c = b + [0] * k       # calcul de P.X^k dans c
    while degre(c) >= k:
        plus(c, g, n+k-1-degre(c), p-1-k, k+1)
    return c[-k:]
```

b. Le coût spatial de cette fonction est lié à la création du tableau `c` ; c'est donc un $\Theta(n+k)$.
Le coût de la fonction `plus` est proportionnel à son dernier argument et celui de la fonction `degre` est dominé par la taille de son argument. Sachant que la boucle conditionnelle est exécutée au plus n fois, le coût temporel est un $O(n(k+n))$.

Question 9. L'amélioration demandée utilise la remarque suivante : lors du calcul des différentes sommes, seuls $k+1$ bits de b sont utilisés ; ce sont ceux-ci que nous allons stocker dans le registre. Il s'agit donc de parcourir le tableau b par paquets de $k+1$ bits en procédant ainsi :

– si le bit de poids fort est un 0, on se contente de décaler le registre d'un cran vers la droite ;
– si le bit de poids fort est un 1, on ajoute g et on décale d'un cran vers la droite.

L'utilisation d'un registre circulaire permet de réaliser l'opération de décalage en coût constant.

```
def crc1(b, g):
    n, p = len(b), len(g)
    k = degre(g)
    r, d = b[:k+1], 0
    for i in range(n):
        if r[d] == 1:                  # addition de r et de g
            for i in range(k+1):
                r[(d+i) % (k+1)] = ou_exclusif(r[(d+i) % (k+1)], g[p-1-k-i])
        if k + i + 1 < n:              #
            r[d] = b[k+i+1]            # décalage du registre
        d = (d+1) % (k+1)              #
    return r[-k:]
```

Au passage nous avons aussi gagné en complexité temporelle puisque cette dernière est maintenant un $O(nk)$.

Circuits dédiés

Question 10. Pour tout $i \in [\![0, n+k-2]\!]$,

$$\mathrm{R}_{i+1}(\mathrm{X}) = \Big(\mathrm{X}(b_0\mathrm{X}^i + b_1\mathrm{X}^{i-1} + \cdots + b_{i-1}\mathrm{X} + b_i) + b_{i+1}\Big) \bmod \mathrm{G}(\mathrm{X}) = \Big(\mathrm{X}\mathrm{R}_i(\mathrm{X}) + b_{i+1}\Big) \bmod \mathrm{G}(\mathrm{X}).$$

Posons $\mathrm{R}_i(\mathrm{X}) = \alpha_0\mathrm{X}^{k-1} + \alpha_1\mathrm{X}^{k-2} + \cdots + \alpha_{k-1}$.
Alors $\mathrm{X}\mathrm{R}_i(\mathrm{X}) + b_{i+1} = \alpha_0\mathrm{X}^k + \alpha_1\mathrm{X}^{k-1} + \cdots + \alpha_{k-1}\mathrm{X} + b_{i+1}$ donc $\deg\Big(\mathrm{X}\mathrm{R}_i(\mathrm{X}) + b_{i+1}\Big) \oplus \alpha_0\mathrm{G}(\mathrm{X}) \leqslant k-1$ et par conséquent :

$$\Big(\mathrm{X}\mathrm{R}_i(\mathrm{X}) + b_{i+1}\Big) \oplus \alpha_0\mathrm{G}(\mathrm{X}) = \Big(\mathrm{X}\mathrm{R}_i(\mathrm{X}) + b_{i+1}\Big) \bmod \mathrm{G}(\mathrm{X}) = \mathrm{R}_{i+1}(\mathrm{X}).$$

On notera en particulier que : $\mathrm{R}_{n-1+k}(\mathrm{X}) = \mathrm{X}^k\mathrm{P}(\mathrm{X}) \bmod \mathrm{G}(\mathrm{X})$; cette formule permet le calcul par récurrence du CRC.

Question 11.

a. Les valeurs successives prises par le tableau $[r_0, r_1, r_2, r_3, r_4]$ définissent une suite de polynômes $\Big(\widetilde{\mathrm{R}}_i(\mathrm{X})\Big)_{0 \leqslant i \leqslant n+4}$ débutant ainsi :

$$\begin{aligned}
\widetilde{\mathrm{R}}_0(\mathrm{X}) &= b_0 \\
\widetilde{\mathrm{R}}_1(\mathrm{X}) &= b_0\mathrm{X} + b_1 \\
\widetilde{\mathrm{R}}_2(\mathrm{X}) &= b_0\mathrm{X}^2 + b_1\mathrm{X} + b_2 \\
\widetilde{\mathrm{R}}_3(\mathrm{X}) &= b_0\mathrm{X}^3 + b_1\mathrm{X}^2 + b_2\mathrm{X} + b_3 \\
\widetilde{\mathrm{R}}_4(\mathrm{X}) &= b_0\mathrm{X}^4 + b_1\mathrm{X}^3 + b_2\mathrm{X}^2 + b_3\mathrm{X} + b_4
\end{aligned}$$

Après cette étape r_0 a pris la valeur de b_0 donc la dernière étape du parcours du circuit revient à calculer :

$$\widetilde{\mathrm{R}}_5(\mathrm{X}) = (b_1\mathrm{X}^4 + b_2\mathrm{X}^3 + b_3\mathrm{X}^2 + b_4\mathrm{X} + b_5) \oplus (b_0\mathrm{X}^4 + b_0\mathrm{X}^2 + b_0)$$

Puisque $b_0 \oplus b_0 = 0$, on peut aussi écrire :

$$\widetilde{\mathrm{R}}_5(\mathrm{X}) = (b_0\mathrm{X}^5 + b_1\mathrm{X}^4 + b_2\mathrm{X}^3 + b_3\mathrm{X}^2 + b_4\mathrm{X} + b_5) \oplus (b_0\mathrm{X}^5 + b_0\mathrm{X}^4 + b_0\mathrm{X}^2 + b_0),$$

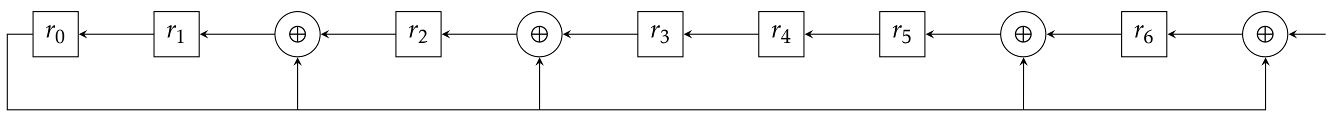
soit :

$$\widetilde{\mathrm{R}}_5(\mathrm{X}) = (b_0\mathrm{X}^5 + b_1\mathrm{X}^4 + b_2\mathrm{X}^3 + b_3\mathrm{X}^2 + b_4\mathrm{X} + b_5) \oplus \Big(b_0\mathrm{G}(\mathrm{X})\Big) = (b_0\mathrm{X}^5 + b_1\mathrm{X}^4 + b_2\mathrm{X}^3 + b_3\mathrm{X}^2 + b_4\mathrm{X} + b_5) \bmod \mathrm{G}(\mathrm{X}).$$

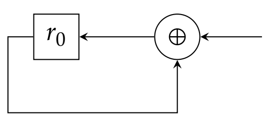
Plus généralement, si on note $\widetilde{\mathrm{R}}_i(\mathrm{X}) = \alpha_0\mathrm{X}^4 + \alpha_1\mathrm{X}^3 + \alpha_2\mathrm{X}^2 + \alpha_3\mathrm{X} + \alpha_4$, alors :

$$\widetilde{\mathrm{R}}_{i+1}(\mathrm{X}) = \Big(\mathrm{X}\widetilde{\mathrm{R}}_i(\mathrm{X}) + b_{i+1}\Big) \oplus \Big(\alpha_0\mathrm{G}(\mathrm{X})\Big)$$

donc $\widetilde{\mathrm{R}}_i(\mathrm{X}) = \mathrm{R}_i(\mathrm{X})$ et en particulier, $\widetilde{\mathrm{R}}_{n+4}(\mathrm{X})$ est le polynôme associé au CRC.
Ainsi, le circuit associé au polynôme générateur $\mathrm{G}(\mathrm{X}) = \mathrm{X}^7 + \mathrm{X}^5 + \mathrm{X}^4 + \mathrm{X} + 1$ est :



b. Considérons le polynôme générateur $\mathrm{G}(\mathrm{X}) = \mathrm{X} + 1$. Il correspond au circuit suivant :



Autrement dit, le CRC est ici égal à : $b_0 \oplus b_1 \oplus \cdots \oplus b_{n-1}$; c'est le bit de parité.