TD22 : Programmation dynamique

On travaille sur les polygones convexes du plan.
Une *corde* d'un polygone est une droite qui relie deux de ses sommets non consécutifs.
Une *triangulation* d'un polygone est un ensemble de cordes qui ne se coupent pas à l'intérieur du polygone et le divisent en triangle.

Question 1 : Montrer que toute triangulation d'un polygone à n côtés possède $n-3$ cordes et coupe le polygone en $n-2$ triangles.

On souhaite maintenant obtenir une triangulation optimale d'un polygone, c'est-à-dire une triangulation qui minimise la somme d'une certaine mesure sur les triangles (par exemple les périmètres des triangles, mais cela pourrait être autre chose, du moment que c'est positif).
On note $P = (v_0, \ldots, v_{n-1})$ les sommets du polygone dans l'ordre direct, et $w(i, j, k)$ le poids associé au triangle (v_i, v_j, v_k).
On note $t(i, j)$, $0 \leq i < j < n$, la pondération d'une triangulation optimale du polygone $(v_{i-1}, \ldots, v_j)$ (avec $t(i, i) = 0$ pour tout $0 \leq i < n$).

Question 2 : Définir récursivement t. En déduire un algorithme de calcul et sa complexité.

TD22 : Programmation dynamique

On travaille sur les polygones convexes du plan.
Une *corde* d'un polygone est une droite qui relie deux de ses sommets non consécutifs.
Une *triangulation* d'un polygone est un ensemble de cordes qui ne se coupent pas à l'intérieur du polygone et le divisent en triangle.

Question 1 : Montrer que toute triangulation d'un polygone à n côtés possède $n-3$ cordes et coupe le polygone en $n-2$ triangles.

On souhaite maintenant obtenir une triangulation optimale d'un polygone, c'est-à-dire une triangulation qui minimise la somme d'une certaine mesure sur les triangles (par exemple les périmètres des triangles, mais cela pourrait être autre chose, du moment que c'est positif).
On note $P = (v_0, \ldots, v_{n-1})$ les sommets du polygone dans l'ordre direct, et $w(i, j, k)$ le poids associé au triangle (v_i, v_j, v_k).
On note $t(i, j)$, $0 \leq i < j < n$, la pondération d'une triangulation optimale du polygone $(v_{i-1}, \ldots, v_j)$ (avec $t(i, i) = 0$ pour tout $0 \leq i < n$).

Question 2 : Définir récursivement t. En déduire un algorithme de calcul et sa complexité.