1. Show that \( P \) is closed under union, concatenation, complement, and the star operation.
   
   Hint (for star): Use dynamic programming.

2. EXACT3COVER (X3C) is the following problem: given a positive integer \( m \), and a sequence of subsets \( S_1, S_2, ..., S_r \subseteq U = \{1, 2, ..., m\} \), such that \( |S_i| = 3 \) for all \( i \), determine if there is a subset \( T \subseteq \{1, 2, ..., r\} \) such that \( \bigcup_{j \in T} S_j = U \) and \( \forall i, j \in T \) with \( i \neq j \) we have \( S_i \cap S_j = \emptyset \) (Such a set \( T \) of \( m/3 \) disjoint sets whose union is \( U \) is called an exact cover of \( U \)).

   Formulate X3C as a language. You can assume this X3C language is NP-hard and that all the theorems listed in the notes are true. Prove that X4C (formulate it as a language in a similar manner), described below, is NP-hard:

   EXACTCOVERBY4SETS (X4C) is the following problem: given a positive integer \( m \), and a sequence of subsets \( S_1, S_2, ..., S_r \subseteq U = \{1, 2, ..., m\} \), such that \( |S_i| = 4 \) for all \( i \), determine if there is a subset \( T \subseteq \{1, 2, ..., r\} \) such that \( \bigcup_{j \in T} S_j = U \) and \( \forall i, j \in T \) with \( i \neq j \) we have \( S_i \cap S_j = \emptyset \).

3. Consider the following problem, called SHORTEST SIMPLE \( s - t \) PATH: Given a directed graph \( G = (V, E, w) \), where \( w(e) \) is defined as a (possibly negative) integer for each edge \( e \in E \), vertices \( s, t \in V \), and a positive integer \( K \), answer YES if there is a simple \( s - t \) path of total weight at most \( K \). An \( s - t \) path starts at \( s \) and ends at \( t \), a path is simple if it does not repeat any vertices, and the total weight of a path is the sum of the weights of its edges.

   Prove that the SHORTEST SIMPLE \( s - t \) PATH problem is NP-hard. Simplest reduction is from HAMPATH.

4. Consider the following problem, called DOMINATING-SET: Given a graph \( G = (V, E) \), and an integer \( k \), is there a set of vertices \( A \subseteq V \) such that \( |A| = k \) and every vertex of \( V \) is either in \( A \) or has a neighbour in \( A \).

   Formulate DOMINATING-SET as a language and prove that this language is NP-hard. Hints: If you plan to use 3SAT, use three vertices for each variable and one vertex for each clause. If you plan to use VERTEX-COVER, add \( m + 2 \) vertices, where \( m \) is the number of edges in the original instance.

   Of course, there are correct solutions which ignore the hints.