Consider the difference hierarchy D_iP, which is defined recursively as

- $D_1P = NP\ and$
- $D_iP = \{A \mid A = B \cap \overline{C} \text{ for } B \text{ in } NP \text{ and } C \text{ in } D_{i-1}P\}$

Now consider the statement which is given below:

$Z = \{\langle G_1, k_1, G_2, k_2 \rangle \mid G_1 \text{ has a } k_1 - \text{clique and } G_2 \text{ doesn't have a } k_2 - clique\}$

The above given statement (Z) can be written in the form:

$Z = \{\langle G_1, k_1, G_2, k_2 \rangle \mid \langle G_1, k_1 \rangle \text{ in } CLIQUE \text{ and } \langle G_2, k_2 \rangle\ in\ \overline{CLIQUE}\}$

• Suppose in DP, an arbitrary language is defined as $A = B \cap \overline{C}$. Any language is reducible in polynomial to $CLIQUE$ if they will be in NP.

• So, $B\ and\ C$ is polynomial reducible to $CLIQUE$. Hence, there exists a polynomial reduction function $S(B)$ and $S(C)$ which is used to reduce $B\ and\ C$ respectively.

• Both of the above functions output a coding like $\langle G, k \rangle$, where k is defined as the clique size and G is defined as a graph. So, the reduction ($S(w)$) of both the function can be generated as $S(w) = S_B(w), S_c(w)$, which comprises a well definition of element of $Z \cup \overline{Z}$.

Suppose w is in $B \cap \overline{C}$ then it shows that $S_B(w)$ is in $CLIQUE$ and $S_c(w)$ is in $\overline{CLIQUE}$. So that $S(w)$ will not be in Z. Hence, language A will contain w if and only if $S(w)$ in Z. As, $S(B)$ and $S(C)$ are polynomial and also $S(w)$ **shows polynomial behavior**. Therefore, A **is polynomial reducible to** Z**. Hence it can be said that** Z **is complete for DP.**