**Worksheet 9**  
**EEL 4705**  
**Emerging Logic Devices – K-Map based Mapping**  
(To convert AND/OR Logic to Majority Logic)

**Question 1:** Making use of the Algorithm and the K-Maps depicted, reduce the following function into a Majority Logic function. Each of the three functions \( (f_1, f_2, f_3) \) will be only from the Library of K-Map patterns depicted above:

- \( n = a\overline{b}c + \overline{a}b\overline{c} + \overline{a}bc + abc \)
- Function needs to be broken in the form \( n = \text{Maj} (f_1, f_2, f_3) \)
- Find an admissible pattern for \( f_1 \) from the above library.
  
For finding \( f_1 \), set \( \Psi_1 \) is obtained as follows: if a minterm of \( n \) is not a minterm of \( f_1 \), add this minterm to \( \Psi_1 \).

Similarly, for finding \( f_2 \), set \( \Psi_0 \) is obtained as follows: if a maxterm of \( n \) is not a maxterm of \( f_1 \), add this maxterm to \( \Psi_0 \).

- A suitable pattern for \( f_2 \) is then determined using new \( \Psi_1 \) and \( \Psi_0 \) (from the above library).

Furthermore, to determine \( f_3 \), \( \Psi_1 \) and \( \Psi_0 \) are updated again as follows: if a minterm (maxterm) of node \( n \) is not a minterm (maxterm) of both \( f_1 \) and \( f_2 \), add this minterm (maxterm) to \( \Psi_1 \) (\( \Psi_0 \)).
**Question 2:** Perform the AND/OR mapping of the same expression \( n = \overline{a} \cdot \overline{b} \cdot \overline{c} + a \cdot b \cdot c + a \cdot \overline{b} \cdot c + a \cdot b \cdot c \). Then see the difference in the number of majority gates used for K-map method and AND/OR method.