We may employ the following algorithm to attempt to color given graph $G = (V, E)$ black and white.

1: Set variable *color* to *black* (arbitrary).

2: Assign to v the color specified by *color*.

3: If $color = black$, set *color* to *white*. Otherwise, set *color* to *black*.

4: For each vertex w adjacent to v, if w is colored and $color(w) = color(v)$, then indicate that no black and white coloring exists. Otherwise, recurse on w (go to step 2, passing w to v).

The algorithm simply performs a depth first search perfoming coloration as nodes are visited. Thus, the time complexity is $O(N + M)$ (the complexity of depth first search).