

```

def compute_weight_matrix(self, eig_pairs, cum_var_exp):
    """
    compute weight matrix of top principal components conditioned on target
    explained variance.
    (Hint : use cumulative explained variance and target_explained_variance to find
    top components)

    :param eig_pairs: list of tuples containing eigenvalues and eigenvectors,
    sorted by eigenvalues in descending order (the biggest eigenvalue and corresponding eigenvectors first).
    :param cum_var_exp: cumulative explained variance by features
    :return: weight matrix (the shape of the weight matrix is n X k)
    """
    # your code here
    weight_matrix = np.ones((self.feature_size, 1))
    for i in range(len(eig_pairs)):
        if cum_var_exp[i] < self.target_explained_variance:
            weight_matrix[i] = eig_pairs[i][1]

    return weight_matrix

```