```cpp
void initializeEnemyMap(std::map<std::string, std::tuple<int, int, int, int, int, int>>& enemyMap)
{
    // int w, int h, int d, int points, int left, int up
    /* ********** V E R T I C A L ********** */
    enemyMap["V_L_L"] = std::make_tuple(7, H, 10, 90, 1, -1);
    enemyMap["V_L_R"] = std::make_tuple(7, H, 10, 90, 0, -1);
    enemyMap["V_M_L"] = std::make_tuple(10, H, 10, 70, 1, -1);
    enemyMap["V_M_R"] = std::make_tuple(10, H, 10, 70, 0, -1);
    enemyMap["V_B_L"] = std::make_tuple(L, H, 10, 40, 1, -1);
    enemyMap["V_B_R"] = std::make_tuple(L, H, 10, 40, 0, -1);

    /* ********** H O R I Z O N T A L ********** */
    enemyMap["H_L_U"] = std::make_tuple(L, 4, 10, 90, -1, 1);
    enemyMap["H_L_B"] = std::make_tuple(L, 4, 10, 90, -1, 0);
    enemyMap["H_M_U"] = std::make_tuple(L, 8, 10, 70, -1, 1);
    enemyMap["H_M_B"] = std::make_tuple(L, 8, 10, 70, -1, 0);
    enemyMap["H_B_U"] = std::make_tuple(L, H, 10, 40, -1, 1);
    enemyMap["H_B_B"] = std::make_tuple(L, H, 10, 40, -1, 0);

    /* ********** S Q U A R E ********** */
    enemyMap["L_L_B"] = std::make_tuple(7, 7, 10, 100, 1, 0);
    enemyMap["L_R_B"] = std::make_tuple(7, 7, 10, 100, 0, 0);
    enemyMap["L_L_U"] = std::make_tuple(7, 7, 10, 100, 1, 1);
    enemyMap["L_R_U"] = std::make_tuple(7, 7, 10, 100, 0, 1);

    enemyMap["M_L_B"] = std::make_tuple(10, 10, 10, 80, 1, 0);
    enemyMap["M_R_B"] = std::make_tuple(10, 10, 10, 80, 0, 0);
    enemyMap["M_L_U"] = std::make_tuple(10, 10, 10, 80, 1, 1);
    enemyMap["M_R_U"] = std::make_tuple(10, 10, 10, 80, 0, 1);

    enemyMap["B_L_B"] = std::make_tuple(15, 15, 10, 50, 1, 0);
    enemyMap["B_R_B"] = std::make_tuple(15, 15, 10, 50, 0, 0);
    enemyMap["B_L_U"] = std::make_tuple(15, 15, 10, 50, 1, 1);
    enemyMap["B_R_U"] = std::make_tuple(15, 15, 10, 50, 0, 1);
}
```