#include <Keypad.h>

#include <Servo.h> // Ajoute la bibliothèque Servo

// Définition des broches et des variables

int compteur = 0; // Compte le nombre de touches appuyées

const int MAX\_TENTATIVES = 3; // Limite de tentatives avant blocage

// Définition de la LED verte

const int led\_verte = 12;

// Définition de la broche du servo-moteur

const int servoPin = 13;

Servo servo; // Crée un objet Servo

// Définition de la matrice du clavier

const byte Ligne = 4;

const byte Colonne = 4;

char hexaBouton[Ligne][Colonne] = {

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'\*','0','#','D'}

};

char code[3]; // Contient les touches tapées par l'utilisateur

byte Ligne\_Pins[Ligne] = {9, 8, 7, 6}; // Broches des lignes du clavier

byte Colonne\_Pins[Colonne] = {5, 4, 3, 2}; // Broches des colonnes du clavier

Keypad mon\_keypad = Keypad(makeKeymap(hexaBouton), Ligne\_Pins, Colonne\_Pins, Ligne, Colonne);

void setup(){

pinMode(led\_verte, OUTPUT); // La LED verte est en sortie

digitalWrite(led\_verte, LOW); // Éteint la LED verte au départ

servo.attach(servoPin); // Attache le servo à la broche 13

servo.write(0); // Position initiale du servo (porte fermée)

}

void loop(){

char keypad\_matrix = mon\_keypad.getKey(); // Récupère la touche appuyée

if (keypad\_matrix) { // Si une touche est appuyée

code[compteur] = keypad\_matrix; // Met la valeur dans la liste

compteur += 1; // Incrémente pour se déplacer dans la liste des valeurs

if (keypad\_matrix == 'A') { // Si on appuie sur A, on réinitialise les valeurs

compteur = 0;

}

if (compteur == 3) { // Si 3 chiffres sont saisis

if (code[0] == '1' && code[1] == '2' && code[2] == '3') { // Si le code est "123"

digitalWrite(led\_verte, HIGH); // Allume la LED verte pour indiquer que la porte est ouverte

servo.write(90); // Ouvre la porte (position 90 degrés)

delay(5000); // Attend 5 secondes avant de refermer

servo.write(0); // Ferme la porte (position 0 degrés)

digitalWrite(led\_verte, LOW); // Éteint la LED verte après la fermeture

}

compteur = 0; // Réinitialise le compteur si l'utilisateur veut retenter

}

}

}