

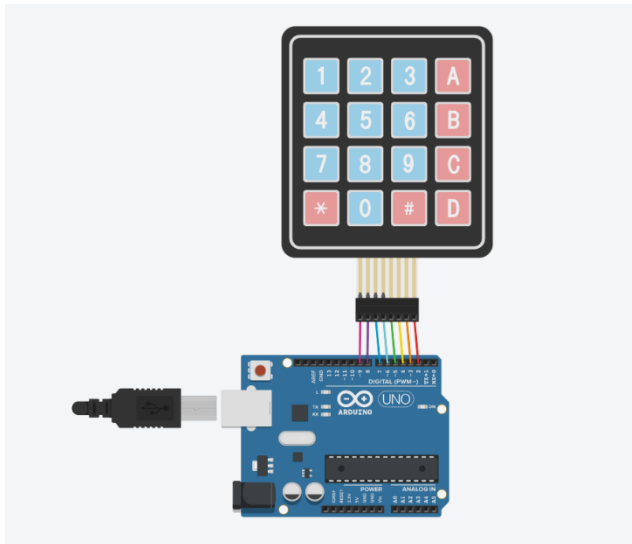
## SEANCE 1 PROJET ARDUINO – ELISA LARTIGUE

Pour cette première séance, je devais m'occuper de verrouillage et de déverrouillage de la boîte puis de l'envoi des informations du pavé numérique sur le terminal. J'ai tout d'abord cherché le fonctionnement du pavé numérique 4x4. Chaque chiffre du pavé numérique est défini comme un numéro de ligne et un numéro de colonne. Les lignes sont branchées sur les entrées 9, 8, 7, 6 et les colonnes 5,4,3,2. Lorsque l'utilisateur appuie sur un chiffre, il crée un contact entre une colonne et une ligne. Par exemple, pour le chiffre 8, la colonne reliée à l'entrée 4 est mise en contact avec la ligne 7.

### TRANSFORMATION DU CODE TAPE EN NOMBRE APPARAISSANT SUR LE TERMINAL

J'ai ensuite cherché à faire apparaître les numéros tapés sur le clavier numérique, sur le terminal.

Montage réalisé :



Montage réalisé sur tinkercad car soucis avec une librairie, qui m'empêche de réaliser le montage en vrai

Code :

```
#include <Keypad.h>

const byte LIGNE = 4; //on a 4 lignes
const byte COLONNE = 4; //on a 4 colonnes

char keys[LIGNE][COLONNE] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

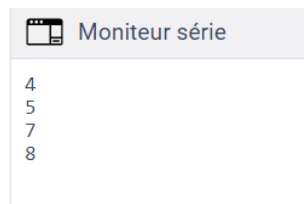
byte colonnePins[COLONNE] = {5, 4, 3, 2}; //chaque pin correspond à une ligne
byte lignePins[LIGNE] = {9, 8, 7, 6}; //chaque pin correspond à une colonne

Keypad keypad = Keypad( makeKeymap(keys), lignePins, colonnePins, LIGNE, COLONNE);

void setup() {
  Serial.begin(9600);
}

void loop() {
  char key = keypad.getKey(); //renvoie la touche sous forme de caractère
  if (key) {
    Serial.println(key);
  }
}
```

### Exemple de résultats :



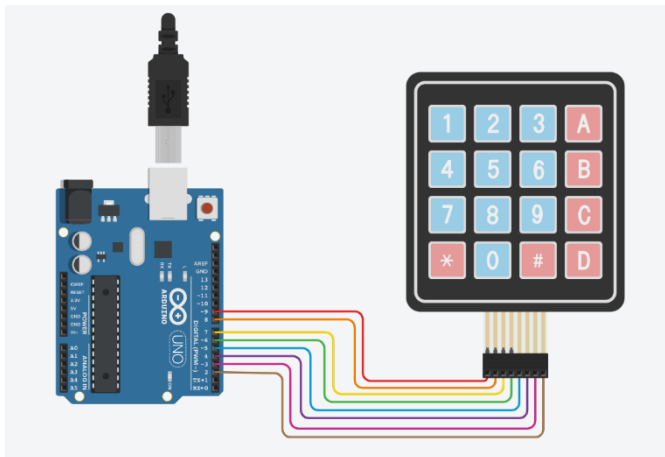
### Problèmes rencontrés :

Le chiffre sur lequel j'appuyais ne correspondait pas au chiffre affiché dans le moniteur série. En effet, c'est comme si j'avais un effet miroir qui se faisait sur la diagonale passant de « \* » à « A ». Quand je tapais sur le « 1 », le « D » apparaissait sur le moniteur série. Une mauvaise compréhension du fonctionnement des lignes et des colonnes m'avait amené à nommer mes lignes et colonnes dans le mauvais sens. J'ai donc interverti le sens des chiffres sur la ligne « byte colonnePins[COLONNE] » et sur la ligne « byte lignePins[LIGNE] ».

### **DEFINITION D'UN CODE ET COMPARAISON AVEC LE CODE ATTENDU**

J'ai ensuite dû créer un code qui permet de comparer un code enregistré au préalable (code attendu) à un code tapé par un utilisateur. J'ai donc défini le « code bon » (dans cet exemple 1234). Lorsque je lance le programme, la phrase : « Entrez votre code : » s'affiche. L'utilisateur tape le code. Celui-ci s'affiche lorsqu'il est complet (4 chiffres rentrés). Dans le cas où ce code correspond au code attendu, la phrase « Code bon ! » s'affiche. Si ça n'est pas le cas, l'utilisateur peut retenter à l'infini.

### Montage :



Montage réalisé sur tinkercad car soucis avec une librairie, qui m'empêche de réaliser le montage en vrai

### Code :

J'ai utilisé la librairie Keypad.h et ai récupéré le code précédent que j'ai complété. J'ai choisi une longueur de 4 chiffres pour le code (#define CODE 4). J'ai choisi le code considéré bon : 1234.

## VOIR PAGE SUIVANTE

```
#include <Keypad.h>
#define LIGNES 4
#define COLONNES 4
#define CODE 4

//Paramètres
const char kp4x4Keys[LIGNES][COLONNES] = {
  {'1', '2', '3', 'A'}, {'4', '5', '6', 'B'}, {'7', '8', '9', 'C'}, {'*', '0', '#', 'D'};
byte lignePin [4] = {9, 8, 7, 6};
byte colonnePin [4] = {5, 4, 3, 2};
char tab[CODE];
char codebon[]="1234";
int i=0;

//Variables
Keypad kp4x4 = Keypad(makeKeymap(kp4x4Keys), lignePin, colonnePin, LIGNES, COLONNES);


void setup() {
  Serial.begin(9600);
  Serial.println("Entrez votre code :");
}

void loop() {
  lecturechiffre();
}

void lecturechiffre(){
  char transformechiffre = kp4x4.getKey(); //récupère le chiffre du keypad
  if (transformechiffre) {
    tab[i]=transformechiffre;//mets les chiffres à la suite des autres
    i=i+1;

    if(i==CODE){
      Serial.println(tab);
      i=0;
      if(strcmp(tab,codebon)==0){ //compare le code entré et celui attendu
        Serial.println("Code bon !");
      }
    }
  }
}
```

## Résultats :

 Moniteur série	Première ligne : Question posée à l'utilisateur.
Entrez votre code : 1234	Deuxième ligne : L'utilisateur rentre 1234.
Code bon ! 4567	Troisième ligne : Le code entré est comparé au code attendu (ici code bon).
7885	Quatrième et cinquième lignes : Deux codes faux entrés. Ils ne correspondent pas au code attendu donc l'utilisateur peut réessayer.

## Problèmes rencontrés :

Transformer le code tapé par l'utilisateur en une chaîne de caractère pour que l'ordinateur puisse le comparer au code attendu. J'ai utilisé la fonction strcmp() pour comparer les deux chaînes de caractères. Cette fonction renvoie un 0 si les deux chaînes sont similaires. Le chiffre sur lequel j'appuyais s'affichait 4 fois au lieu d'une. Seuls les codes à 4 chiffres identiques fonctionnaient (1111 ou 2222 par exemple). Apparition d'un point d'interrogation à la fin du code tapé, que je n'arrive pas à enlever.

#### **Sites web utilisés :**

Comprendre la fonction strcmp() : <https://overiq.com/c-programming-101/the-strcmp-function-in-c/#:~:text=The%20strcmp%20%28%29%20function%20is%20used%20to%20compare,by%20character%20using%20ASCII%20value%20of%20the%20characters.>

Comprendre le pavé numérique : <https://playground.arduino.cc/Code/Keypad/>

Exemple de codes pris à l'appui : <https://www.hackster.io/diy-hacking/arduino-keyless-door-lock-system-with-keypad-and-lcd-bcad2e#:~:text=Arduino%20Keypad%20Lock%20Code,The%20initial%20password%20is%201234.>

<https://www.electronicclinic.com/password-door-lock-security-system-using-arduino-and-keypad/>

VOIR LES LIENS DANS LES COMMENTAIRES SUR GITHUB