

Comandi Arduino Display LCD e Pullup

Scrivere sul display LCD la scritta "AVE STUDENTS" seguita dal numero di secondi trascorsi dall'accensione di Arduino.

```
#include <LiquidCrystal.h>

//Creazione oggetto LiquidCrystal con uscite
// specificate dall'esercizio

// Esempio esercizio: Scrivere sul display LCD la scritta "AVE STUDENTS"
// seguita dal numero di secondi trascorsi dall'accensione di Arduino.

/*
 * Pin R/W dell LCD ⇒ pin GND
 * Pin V0 dell LCD ⇒ pin GND
 * Pin RS dell LCD ⇒ pin digitale 2
 * Pin E  dell LCD ⇒ pin digitale 3
 * Pin D4 dell LCD ⇒ pin digitale 4
 * Pin D5 dell LCD ⇒ pin digitale 5
 * Pin D6 dell LCD ⇒ pin digitale 6
 * Pin D7 dell LCD ⇒ pin digitale 7
 */
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

void setup(){
    // Impostiamo colonne e righe
    // = n.caratteri e righe
    // es. 16 caratteri su 2 righe
    lcd.begin(16, 2);
    // Pulizia display
    lcd.clear();
    /* Dopo avere eseguito:
    [][][][][][][][][][][][][][]
    [][][][][][][][][][][][][][]
    */
    lcd.print("AVE ");
    /* Dopo avere eseguito:
    [A][V][E][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ]
    [ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ]
    */
}
```

```

    */
    lcd.print("STUDENTS");
        /* Dopo avere eseguito:
    [A][V][E][ ][S][T][U][D][E][N][T][S][ ][ ][ ]
    [ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ]
    */
}

void loop(){
    // ISTRUZIONE: setCursor(col, row) - La numerazione parte da 0
    lcd.setCursor(0, 1); // 1° colonna - 2° riga
    // stampa il numero di secondi trascorsi dal reset di Arduino
    lcd.print(millis()/1000);
}
// Impostazione di una certa cella
lcd.setCursor(7,1)
/*
[ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ][ ]
[ ][ ][ ][ ][ ][qui][ ][ ][ ][ ][ ][ ][ ][ ][ ]
*/

```

Creare un simbolo a forma di busta chiusa (disposto in verticale) che oscilla da destra a sinistra sul display LCD

```

// Creazione carattere in byte (qui è la busta)
byte Busta[8] =
{ B11111, B11001, B10101, B10011, B10101, B11001, B11111, B00000};

LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

void setup{
    // Creo il carattere "busta"
    lcd.createChar(0, Segno);
    // massimo 8 caratteri = tutti i segni supportati da un quadrante
    del display (punto compreso)
    lcd.begin(16,2);
    lcd.clear();
    // Scrivi a partire dal primo carattere
    lcd.write(byte(0));
}

void loop(){

```

```

        // Mostro il carattere busta scrollando tra dx e sx
        int i;
        // Questo ciclo sposta il testo a destra
        for (i = 0; i < 16; i++)
        {
            // Sposta il contenuto del display di una colonna a destra
            lcd.scrollDisplayRight();
            delay(150); // piccolo ritardo
        }

        /*
        [][][][][][][][][][][][][]
        [][][][][][][][][][][][][]
        */

        /*
        [Busta][][][][][][][][][][]
        [][][][][][][][][][][][][]
        */
        // Delay
        /*
        [][Busta][][][][][][][][][]
        [][][][][][][][][][][][][]
        */
        // Delay
        /*
        [][][Busta][][][][][][][][]
        [][][][][][][][][][][][][]
        */
        ....
        // Delay
        /*
        [][][][] [][][][][][][][] [Busta]
        [][][][] [][][][][][][][]
        */
        ....

        // Questo ciclo sposta il testo a sinistra
        for (i = 0; i < 16; i++)
        {
            // Sposta il contenuto del display di una colonna a sinistra
            lcd.scrollDisplayLeft();

```

```

    delay(150); // aspetta un attimo
  }
}

// Delay
/*
[] [] [] [] [] [] [] [] [] [] [] [] [] [] []
[] [] [] [] [] [] [] [] [] [] [] [] [] [] [Busta]
*/
// Delay
/*
[] [] [] [] [] [] [] [] [] [] [] [] [] [] []
[] [] [] [] [] [] [] [] [] [] [] [] [] [] [Busta] []
*/
....
/*
[] [] [] [] [] [] [] [] [] [] [] [] [] [] []
[Busta] [] [] [] [] [] [] [] [] [] [] [] [] [] []
*/

```

Creare un programma che sia in grado di leggere il testo digitato sul Serial Monitor e lo mostri sul display LCD. Poichè la linea di invio del serial monitor accetta al massimo 69 caratteri mentre il display utilizzato può mostrare solamente 32 caratteri inserire i necessari controlli che evitino di mostrare la sequenza digitata oltre il trentaduesimo carattere.

```

LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

void setup(){
  Serial.begin(9600);
  lcd.begin(16,2);
  lcd.leftToRight(); // da sinistra a destra (classico)
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Digita: ");
  // riga dopo
  lcd.setCursor(0,1);
  lcd.print("del testo ... ");
}

```

```

Serial.println("Digita del testo (max 32 char) ... ");

}

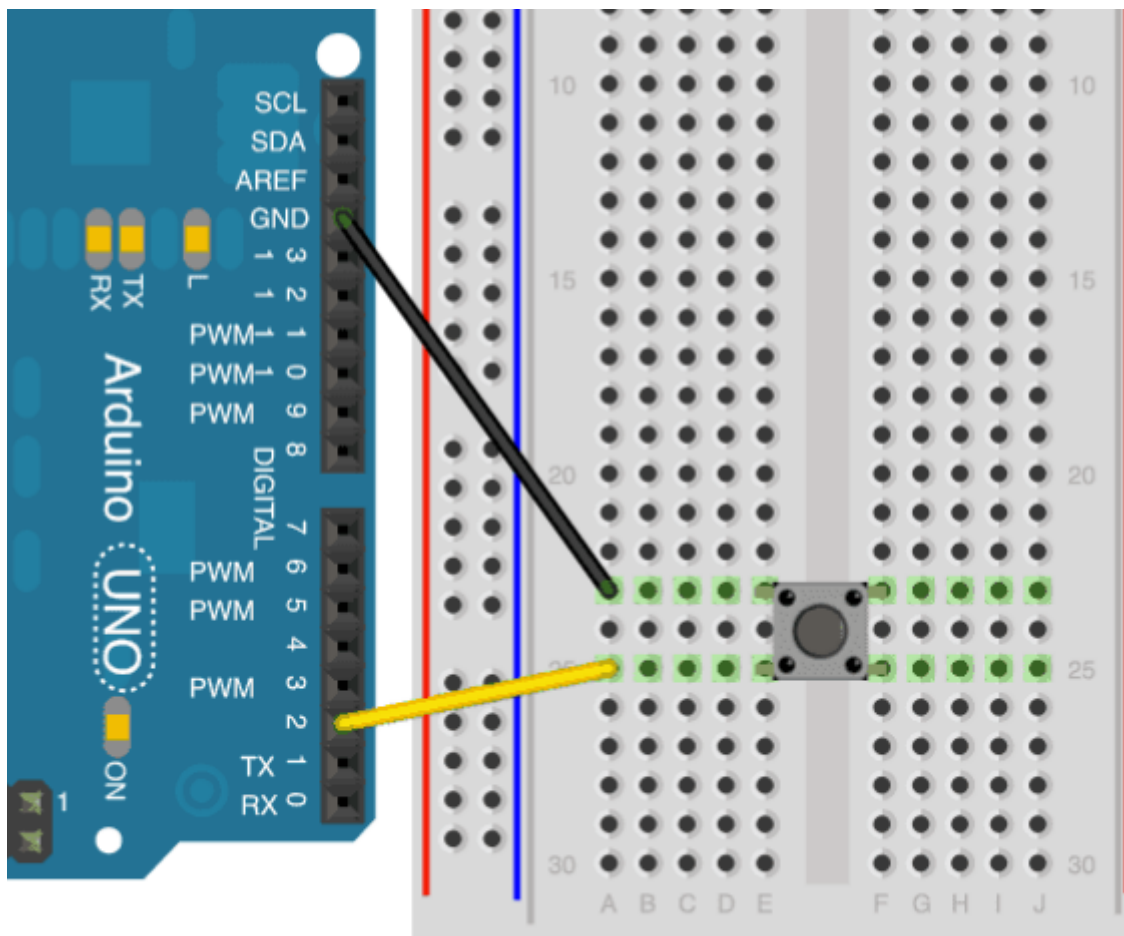
void loop()
{
  char ch;
  int r=0, c=0;
  if (Serial.available()) // Se ci sono caratteri sul Serial Monitor
  {
    // aspetta un attimo affinché arrivi l'intero messaggio
    delay(100);
    lcd.clear(); // pulisce lo schermo
    while (Serial.available() > 0)
    {
      if (r!=2)
      {
        lcd.setCursor(c++,r);
        lcd.write(Serial.read());
        if (c==16) {c=0; r++;}
      }
      else
        Serial.read(); // butto via i caratteri in +
    }
  }
}

```

Pulsante pullup



Esempio base di utilizzo pullup:



```
void setup(){
    Serial.begin(9600);
    // pin di input del pullup
    pinMode(2, INPUT_PULLUP);
    //impostazione output
    pinMode(13, OUTPUT);
}

````arduino
void loop() {46
 // leggi il valore del pullup
 int sensorVal = digitalRead(2);
 Serial.println(sensorVal);

 // HIGH = non è premuto
 // LOW = premuto

 if (sensorVal == HIGH) {
 digitalWrite(13, LOW);
 } else {66
```

```
digitalWrite(13, HIGH);
}
```

## Schema

Esempio di codice:

<pre>#include &lt;LiquidCrystal.h&gt;  LiquidCrystal lcd (12, 11, 5, 4, 3, 2);  void setup() {   lcd.begin (16, 2);    lcd.print ("Hello, world!"); }  void loop() {   lcd.setCursor (0, 1);    lcd.print (millis() / 1000); }</pre>	<p>← Inclusione della libreria che ci permette di usare comandi molto semplici.</p> <p>← Creazione dell'oggetto "lcd". Tra le parentesi bisogna indicare il pin RS, il pin Enable e i quattro pin dei dati partendo dal D4 fino al D7. (RS, Enable, D4, D5, D6, D7)</p> <p>← Specifica quante colonne e righe possiede il nostro display</p> <p>← Il comando "print" stampa a sul display la stringa "Hello, world!"</p> <p>← Questo comando sposta il cursore alla colonna 0 della riga 1. Le righe e colonne partono da 0. Il cursore indica l'inizio della stampa del testo.</p> <p>← Provate a indovinare cosa fa il programma. (!!!Aiuto!!! millis() restituisce i millisecondi trascorsi dall'avvio del programma)</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Elenco funzioni libreria "LiquidCrystal.h":

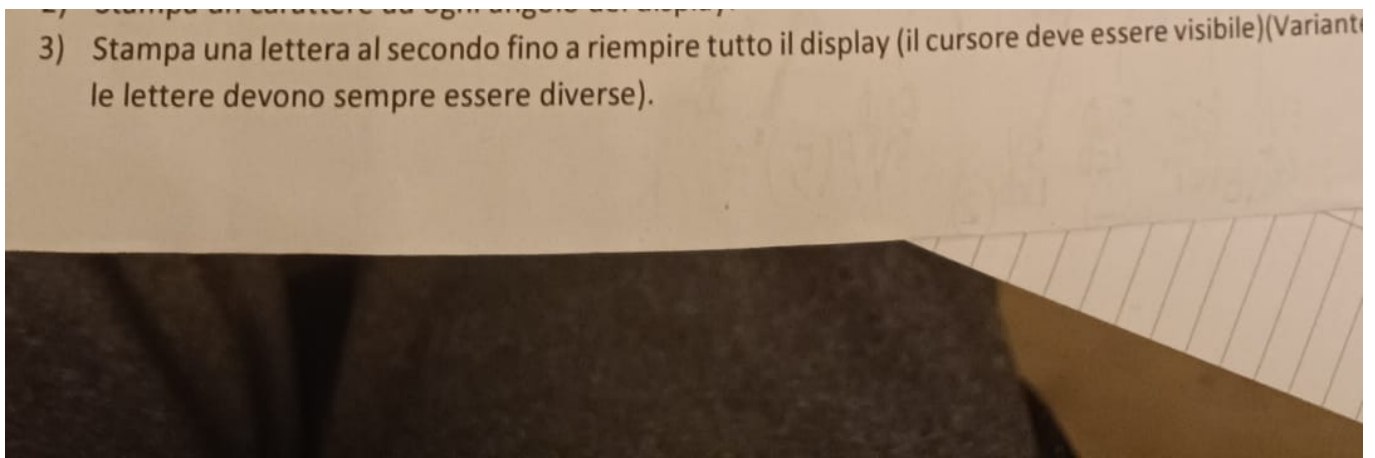
<b>LiquidCrystal</b> ( RS, Enable, D4, D5, D6, D7)	Crea la variabile "LiquidCrystal" e indica come l'LCD è stato collegato.	<b>noBlink()</b>	Disattiva il lampeggio del cursore
<b>begin</b> (colonne, righe)	Serve per indicare la dimensione del LCD	<b>display()</b>	Accende il display
<b>clear()</b>	Cancella tutto il testo presente	<b>noDisplay()</b>	Spegne il display
<b>home()</b>	Sposta il cursore in alto a destra	<b>scrollDisplayLeft()</b>	Sposta il testo di uno spazio a sinistra
<b>setCursor</b> (colonna, riga)	Permette di spostare il cursore dove vogliamo	<b>scrollDisplayRight()</b>	Sposta il testo di uno spazio a destra
<b>write</b> (carattere)	Stampa un carattere	<b>autoscroll()</b>	Man mano inseriamo caratteri sposta il testo stampato di uno spazio.
<b>print</b> (testo)	Stampa dati e stringhe	<b>noAutoscroll()</b>	Disabilita lo scroll automatico
<b>cursor()</b>	Visualizza il cursore	<b>leftToRight()</b>	Imposta la scrittura da sinistra a destra
<b>noCursor()</b>	Nasconde il cursore	<b>rightToLeft()</b>	Imposta la scrittura da destra a sinistra
<b>blink()</b>	Fa lampeggiare il cursore	<b>createChar</b> (numero, vettore di byte)	Ci permette di creare al massimo 8 caratteri.

1) Dopo aver testare il codice sopra riportato svolgere i seguenti esercizi nell'ordine preferito.

Esercizi:

- 1) Stampa del testo e fallo lampeggiare (Aiuto usa clear()) e ristampa il testo dopo un delay adeguato)
- 2) Stampa un carattere ad ogni angolo del display.





## Esercizi del foglio

Stampa del testo e fallo lampeggiare (Aiuto usa clear()) e ristampa il testo dopo un delay adeguato:

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // Inizializza l'oggetto lcd

void setup() {
 lcd.begin(16, 2); // Imposta le dimensioni del display
}

void loop() {
 lcd.clear(); // Cancella il display
 lcd.print("Testo lampeggiante"); // Stampa il testo
 delay(1000); // Aspetta 1 secondo

 lcd.clear(); // Cancella il display
 delay(1000); // Aspetta 1 secondo
}
```

2. Stampa un carattere ad ogni angolo del display:

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
 lcd.begin(16, 2);
}
```

```

void loop() {
 lcd.clear();
 lcd.setCursor(0, 0); // Angolo in alto a sinistra
 lcd.write('A');
 lcd.setCursor(15, 0); // Angolo in alto a destra
 lcd.write('B');
 lcd.setCursor(0, 1); // Angolo in basso a sinistra
 lcd.write('C');
 lcd.setCursor(15, 1); // Angolo in basso a destra
 lcd.write('D');
 delay(1000); // Aspetta 1 secondo
}

```

Stampa una lettera al secondo fino a riempire tutto il display (il cursore deve essere visibile)>

```

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

char letters[] = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"; // Array di lettere
int letterIndex = 0; // Indice per l'array di lettere

void setup() {
 lcd.begin(16, 2);
 lcd.cursor(); // Mostra il cursore
}

void loop() {
 lcd.clear();
 for (int i = 0; i < 16 * 2; i++) { // Itera per tutte le posizioni del
display
 lcd.setCursor(i % 16, i / 16); // Imposta la posizione del cursore
 lcd.write(letters[letterIndex]); // Stampa la lettera corrente
 letterIndex = (letterIndex + 1) % 26; // Passa alla prossima lettera
diversa
 delay(1000); // Aspetta 1 secondo
 }
}

```

**Link utili:**

- Pullup: [Circuit design Copy of PULL\\_UP Arduino - Tinkercad](#)
- Display con pullup: [LCD & Button cases - should work as a switch - Using Arduino / Displays - Arduino Forum](#)
-