

Premessa

Prima di iniziare, è fondamentale fissare alcuni concetti base della programmazione per Arduino:

- Arduino viene programmato nell'ambiente di sviluppo open source **Arduino IDE**.
- Salvo alcune eccezioni (le direttive, riconoscibili per la presenza di un #), le istruzioni per Arduino sono **comandi**.
- Tutti i comandi terminano con un **punto e virgola**.
- Il linguaggio di programmazione è il **C**.
- Le formulazioni di alcune istruzioni sono state create per semplificare la programmazione di Arduino, l'IDE si fa carico di trasformarle in istruzioni del C.
- Il linguaggio usato per Arduino può essere arricchito, mediante un insieme di file detto **libreria**, con nuove proprietà e funzioni specifiche per un determinato dispositivo.
- L'arricchimento viene effettuato mediante la **direttiva**:
`#include nomeLibreria.h`
 Che sia una direttiva lo si capisce, come già detto, dalla presenza del simbolo # (per le direttive non è previsto il punto e virgola).
- Se si usa una libreria, occorre specificare (nella modalità indicata dalla libreria stessa) il nome del dispositivo a cui si potranno applicare le proprietà e le funzioni integrate.
- Un programma, detto **sketch**, viene trasformato in un linguaggio comprensibile per Arduino prima di caricarlo su di esso. Ciò avviene in due fasi:
 - la **precompilazione** in cui vengono eseguite le direttive; ad esempio sono inglobate nel programma tutte le proprietà e funzioni (dette **metodi**) messe a disposizione dalle librerie. In questa fase vengono anche inglobati quei comandi semplificativi, creati appositamente per la programmazione di Arduino e di cui si è detto in precedenza;
 - la **compilazione** vera e propria, cioè la trasformazione del codice binario (negli 0 e 1) familiare per Arduino.
- Se nel selettore della scheda dell'IDE di Arduino non dovesse comparire la WiFi UNO R4, è necessario procedere all'installazione del pacchetto della scheda. Per fare ciò, poiché fa parte del Core, bisogna aprire la "Gestione schede" nel seguente modo:
 - dal menu Strumenti, se si utilizza la versione 1.8.19 dell'IDE
 - in alto a sinistra, se invece si utilizza la versione 2.x dell'IDE

successivamente bisogna cercare UNO R4 WiFi Boards ed installare la versione che si desidera utilizzare, preferibilmente la più aggiornata.

Può essere utile seguire passo passo le indicazioni su

<https://docs.arduino.cc/tutorials/uno-r4-wifi/r4-wifi-getting-started/>



In **File > Esempi** sono presenti moltissimi esempi per questa scheda; qualora non fossero ancora presenti dopo l'installazione del pacchetto, è necessario chiudere e riavviare l'IDE. La scheda viene venduta con un esempio precaricato (*MatrixIntro.ico*).

Lecture consigliate:

- <https://docs.arduino.cc/tutorials/uno-r4-wifi/r4-wifi-getting-started/>
- <https://www.maffucci.it/2023/07/25/usare-la-matrice-led-di-arduino-uno-r4-wifi/>
- <https://www.hackster.io/glowascii/programming-the-arduino-uno-r4-wifi-s-led-matrix-2f9c8d>

Le librerie

È possibile potenziare il proprio Arduino aggiungendo alcune librerie.

Le librerie sono insiemi di codici già scritti e testati che rendono più facile ed intuitivo l'utilizzo di sensori ed attuatori, l'uso del monitor seriale ed altro, mettendo a disposizione nuove funzionalità.

Alcune librerie sono già integrate nell'IDE, altre vanno aggiunte, altre installate: innumerevoli sono quelle disponibili e scaricabili da internet.

A volte nel programma in cui si include una libreria è necessario effettuare la “richiesta” di far convergere su una variabile, detta **oggetto**, le nuove funzioni e capacità messe a disposizione della libreria stessa; in questi casi la scelta del nome dell'oggetto, possibilmente esplicativo, spetta al progettista purché sia univoco.

Le direttive `#include <nomeLibreria.h>` e `include "nomeLibreria.h"` inducono Arduino ad utilizzare la libreria specificata.

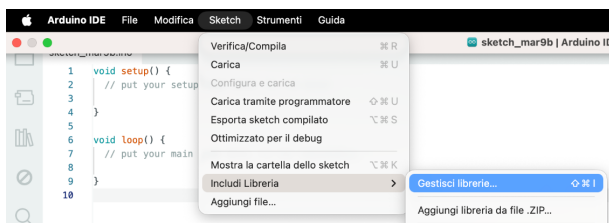
La prima direttiva va usata se la libreria si trova nella cartella predefinita di Arduino, la seconda invece se essa è posizionata nella stessa cartella del progetto.

Esse vanno poste all'inizio del programma e prima del `setup`.

Durante la compilazione, include nel firmware sia il nostro codice che quello della libreria.

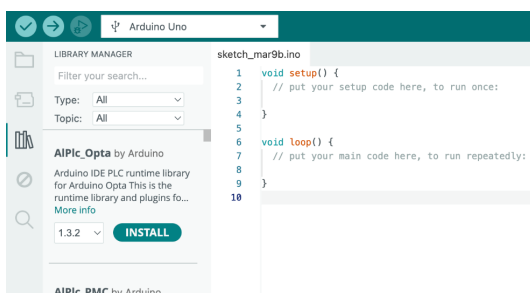
Per prima cosa sarebbe bene controllare che la libreria desiderata sia fra quelle già disponibili nell'IDE: cliccando in corrispondenza della voce di menu **Sketch/#include libreria** verrà mostrato un elenco delle librerie già disponibili; se comparisse la libreria desiderata, basterebbe cliccare su di essa ed il codice di inclusione verrebbe aggiunto automaticamente al nostro programma.

Se invece la libreria di cui abbiamo bisogno non dovesse comparire fra quelle già disponibili, bisognerebbe controllare se si trova fra quelle che è possibile aggiungere dalla voce di menu **Sketch / #include libreria / Gestisci librerie**: cliccando su tale voce si aprirà una finestra con un elenco di librerie pronte per l'installazione.



Se compare la libreria cercata e viene mostrato un menu a discesa, si può scegliere la versione della libreria e cliccare su **Installa**. Una volta che l'installazione sarà terminata, comparirà la dicitura **INSTALLED** (o INSTALLATA) accanto al nome della libreria.

Nella versione 2.x dell'IDE di Arduino, si può procedere cliccando sulla icona della barra laterale sinistra raffigurante libri come mostrato in figura. Si aprirà direttamente il gestore di librerie (Library Manager).



Utilizzo della matrice LED integrata

Per poter gestire la matrice di LED è necessario:

1. includere la libreria **Arduino_LED_Matrix**
#include <Arduino_LED_Matrix.h>
Non è necessario installarla; è già incorporata!
2. definire un oggetto con un nome di nostra scelta, ad es. **MatriceLED**
ArduinoLEDMatrix MatriceLED;
dove *ArduinoLEDMatrix* è il costruttore indicato dalla libreria
3. inizializzare l'oggetto nel setup
MatriceLED.begin();

Per cui, in definitiva, il codice di avvio sarà del tipo:

```
#include <Arduino_LED_Matrix.h>
ArduinoLEDMatrix MatriceLED;
void setup() {
  MatriceLED.begin();
}
```