

La programmazione di Arduino



G. MARSELLA
UNIVERSITÀ DEL SALENTO



Cenni di Programmazione per Arduino

Per programmazione si intende un insieme di operazioni ordinate (istruzioni) per raggiungere un obiettivo

Istruzioni – Esecuzione – Obiettivo

Cenni di Programmazione per Arduino

Algoritmo

L'Algoritmo è costituito da una lista finita di istruzioni da eseguire finalizzate al raggiungimento di un obiettivo in un tempo finito.

L'arte di "usare e miscelare" uno o più algoritmi per raggiungere uno scopo si chiama Programmazione.

Cenni di Programmazione per Arduino

Algoritmo

Ci sono vari modi per rappresentare un algoritmo, a seconda della vastità e comprensività del lavoro che si deve affrontare

Cenni di Programmazione per Arduino

Metodo di sviluppo del Software

Definire il problema

Sviluppare l'algoritmo

Se necessario implementate il progetto

Provare e verificare l'implementazione

Effettuare la manutenzione

Cenni di Programmazione per Arduino

Struttura di un programma:

Direttive per il compilatore: # include <libreria>

Parte dichiarativa globale : variabili,...

Parte di preparazione: setup() {blocco codice}

Parte esecutiva: loop() {

Parte dichiarativa locale: variabili,...

blocco codice}

// commento alla linea

/* commento

al testo */

Cenni di Programmazione per Arduino

Le variabili sono dei contenitori

Definizione:

tipo nome_variabile;

tipo nome_variabile = valore_iniziale;

Tipo di dati

Cenni di Programmazione per Arduino

Struttura di un programma:

Parte di preparazione: setup() {blocco codice}

```
void setup() {  
    // initialize the LED pin as an output  
    pinMode(ledPin, OUTPUT);  
}
```

Cenni di Programmazione per Arduino

Struttura di un programma:

Parte di preparazione: loop() {blocco codice}

void loop() {

 Parte dichiarativa locale: variabili...

 blocco codice;

 blocco codice;

 blocco codice;

}

Cenni di Programmazione per Arduino

Blocco di codice:

Condizione: if else, case

Cicli: For, while

Funzioni: f(x) { blocco di codice; }

Cenni di Programmazione per Arduino

Condizione: if else

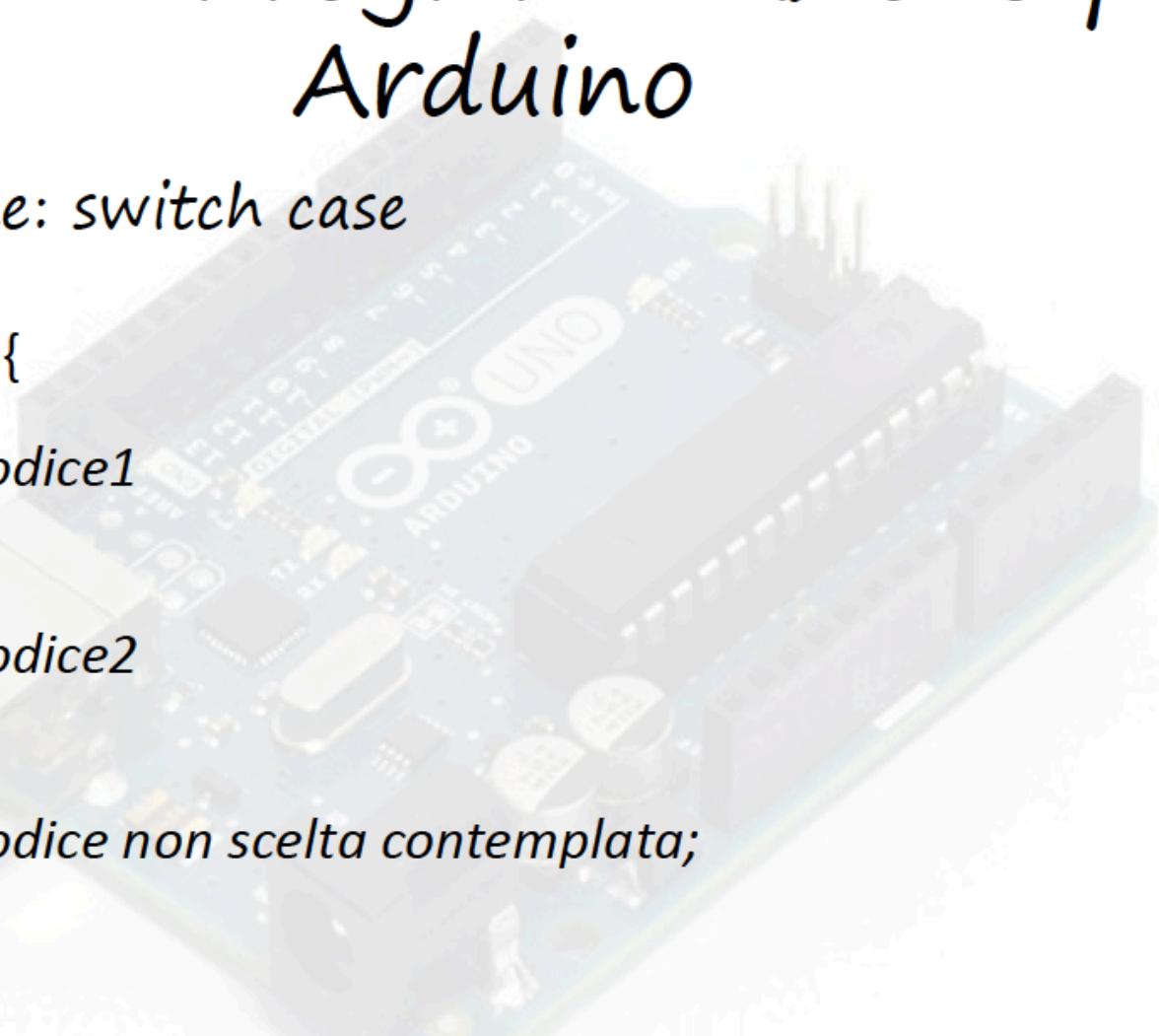
```
if (pinFiveInput < 500) {  
    blocco codice 1;  
}  
else {  
    blocco codice2;  
}  
  
if (x > 0 || y > 0) {  
    blocco codice;  
}
```

[reference](#)

Cenni di Programmazione per Arduino

Condizione: switch case

```
switch (var) {  
    case 1:  
        blocco codice1  
        break;  
    case 2:  
        blocco codice2  
        break;  
    default:  
        blocco codice non scelta contemplata;  
        break;  
}
```



Cenni di Programmazione per Arduino

Cicli: For, While

```
for (cont=valore1; cont < valore2; cont=incr.) {  
    blocco codice;  
}
```

```
while (espressione_boleana) {  
    blocco codice;  
}
```

Cenni di Programmazione per Arduino

Funzioni:

```
Typo_nome_funzione(par1,...parn) {  
    blocco codice;  
    blocco codice;  
    .....  
    return (valore)  
}
```

Cenni di Programmazione per Arduino

Funzioni:

```
int i = 5;
int j = 3;
int k;

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

void loop() {
    k = moltiplicazione(i, j); // in questo punto viene chiamata la funzione
    Serial.println(k);
    delay(500);
}
// Funzione:
int moltiplicazione(int x, int y) {
    int risultato;
    risultato = x * y;
    return risultato;
}
```

Cenni di Programmazione per Arduino

Array

```
int mioArray[6];  
mioArray[0] = 3;  
mio_array[1] =2;
```

```
int mioArray[6] = {2, 4, -8, 3, 2};  
.....  
loop() {  
int i; for (i = 0; i < 5; i = i + 1) {  
Serial.println(mioArray[i]);  
}
```

Software comparison

	Arduino IDE	C/C++	ASM
Semplicità della programmazione	Easy	Moderate	Challenging
librerie	molte	molte	niente
velocità	lento	veloce	veloce
accessibilità delle caratteristiche del processore	Molto limitata	La maggior parte	completamente