



**ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

Ενσωματωμένα Συστήματα

Ενότητα: ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ ARDUINO

Δρ. Μηνάς Δασυγένης

mdasyg@ieee.org

Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών

Εργαστήριο Ψηφιακών Συστημάτων και Αρχιτεκτονικής Υπολογιστών

<http://arch.ict.e.uowm.gr/mdasyg>

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ψηφιακά Μαθήματα του Πανεπιστημίου Δυτικής Μακεδονίας**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Περιεχόμενα

1.Σκοπός της άσκησης.....	4
2.Τι είναι το arduino.....	4
3.Εγκατάσταση του Arduino IDE.....	4
4.Προγραμματισμός του Arduino.....	6
5.Ηλεκτρονικά του Arduino.....	8
6.Κατάλογος εξαρτημάτων (Inventory List).....	10
7.Αποσφαλμάτωση του Arduino.....	12
8.Κανόνες εργαστηρίου.....	12

1. Σκοπός της άσκησης

- Γνωριμία με την ενσωματωμένη πλατφόρμα arduino, τα περιφερειακά και τη δομή του προγραμματισμού. Εκτέλεση καθοδηγούμενων ασκήσεων.



2. Τι είναι το arduino

Το arduino είναι ένα εργαλείο που μας επιτρέπει να κατασκευάσουμε υπολογιστικά συστήματα που μπορούν να αισθανθούν και να ελέγξουν το φυσικό κόσμο πολύ πιο εύκολα από ότι αν χρησιμοποιούσαμε έναν τυπικό υπολογιστή γραφείου. Είναι μια αρχιτεκτονική που βασίζεται σε ανοιχτό κώδικα, μια πλακέτα μικρο-επεξεργαστή και ένα αναπτυξιακό περιβάλλον για τη συγγραφή προγράμματος για την πλακέτα.

Το arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη διαλογικών λειτουργιών, με είσοδο από μια πληθώρα πηγών (διακόπτες, αισθητήρες,..) και έλεγχο φυσικών αντικειμένων (φώτα, κινητήρες,..). Το arduino μπορεί να είναι αυτόνομο ή να επικοινωνεί με άλλα arduino ή υπολογιστικά συστήματα. Εγκατάσταση του Arduino IDE.

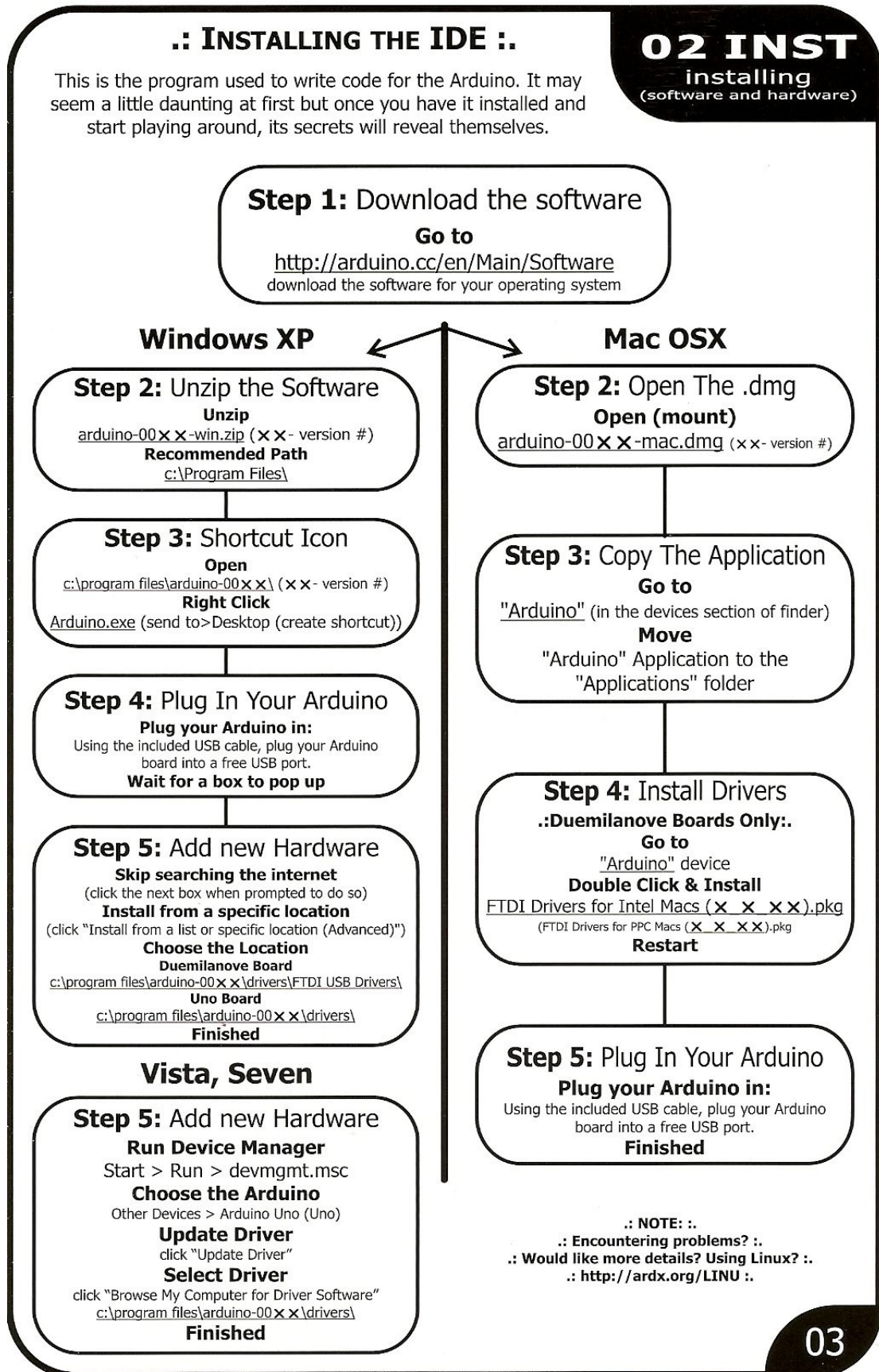
Μερικά από τα πλεονεκτήματα του arduino είναι:

- χαμηλό κόστος
- ανεξαρτήτου ΛΣ (το IDE εκτελείται σε linux, windows, mac)
- απλό μοντέλο προγραμματισμού
- υλικό που μπορεί να προσαρμοστεί στις ανάγκες με πληθώρα περιφερειακών (arduino shields)

3. Εγκατάσταση του Arduino IDE

Αρχικά πρέπει να εγκαταστήσετε το software σε περίπτωση που δεν είναι ήδη εγκατεστημένο στον υπολογιστή που θα εργαστείτε. **Αν το arduino IDE είναι ήδη εγκατεστημένο στον υπολογιστή σας τότε παραλείψτε αυτό το βήμα.**

Μεταβείτε στη [σελίδα](#) και ακολουθήστε το παρακάτω σχεδιάγραμμα για να ολοκληρώσετε την εγκατάσταση.



4. Προγραμματισμός του Arduino

Για να προγραμματίσουμε το arduino χρησιμοποιούμε τη γλώσσα C με κάποιες τροποποιήσεις. Οι παρακάτω 2 σελίδες περιέχουν μια περίληψη της δομής και των εντολών που υποστηρίζει το arduino IDE.

ARDUINO PROGRAMMING IN BRIEF

The Arduino is programmed in the C language. This is a quick little primer targeted at people who have a little bit of programming experience and just need a briefing on the idiosyncracies of C and the Arduino IDE. If you find the concepts a bit daunting, don't worry, you can start going through the circuits and pick up most of it along the way. For a more in-depth intro, the Arduino.cc website is a great resource.

STRUCTURE

Each Arduino program (often called a sketch) has two required functions (also called routines).

`void setup(){ }`
All the code between the two curly brackets will be run once when your Arduino program first runs.

`void loop(){ }`
This function is run after setup has finished. After it has run once it will be run again, and again, until power is removed.

SYNTAX

One of the slightly frustrating elements of C is its formatting requirements (this also makes it very powerful). If you remember the following you should be alright.

`//` (single line comment)
It is often useful to write notes to yourself as you go along about what each line of code does. To do this type two forward slashes and everything until the end of the line will be ignored by your program.

`/* */` (multi line comment)
If you have a lot to say you can span several lines as a comment. Everything between these two symbols will be ignored in your program.

`{ }` (curly brackets)
Used to define when a block of code starts and ends (used in functions as well as loops).

`;` (semicolon)
Each line of code must be ended with a semicolon (a missing semicolon is often the reason for a program refusing to compile).

VARIABLES

A program is nothing more than instructions to move numbers around in an intelligent way. Variables are used to do the moving.

int (integer)
The main workhorse, stores a number in 2 bytes (16 bits). Has no decimal places and will store a value between -32,768 and 32,767.

long (long)
Used when an integer is not large enough. Takes 4 bytes (32 bits) of RAM and has a range between -2,147,483,648 and 2,147,483,647.

boolean (boolean)
A simple True or False variable. Useful because it only uses one bit of RAM.

float (float)
Used for floating point math (decimals). Takes 4 bytes (32 bits) of RAM and has a range between -3.4028235E+38 and 3.4028235E+38.

char (character)
Stores one character using the ASCII code (ie 'A' = 65). Uses one byte (8 bits) of RAM. The Arduino handles strings as an array of char's.

.:For a full programming reference visit:
<http://ardx.org/PROG>

03 PROG programming primer

MATH OPERATORS

Operators used for manipulating numbers. (they work like simple math).

= (assignment) makes something equal to something else (eg. x = 10 * 2 (x now equals 20))
% (modulo) gives the remainder when one number is divided by another (ex. 12 % 10 (gives 2))
+ (addition)
- (subtraction)
* (multiplication)
/ (division)

COMPARISON OPERATORS

Operators used for logical comparison.

== (equal to) (eg. 12 == 10 is FALSE or 12 == 12 is TRUE)
!= (not equal to) (eg. 12 != 10 is TRUE or 12 != 12 is FALSE)
< (less than) (eg. 12 < 10 is FALSE or 12 < 12 is FALSE or 12 < 14 is TRUE)
> (greater than) (eg. 12 > 10 is TRUE or 12 > 12 is FALSE or 12 > 14 is FALSE)

CONTROL STRUCTURE

Programs are reliant on controlling what runs next, here are the basic control elements (there are many more online).

```
if(condition){ }  
else if( condition ){ }  
else { }
```

This will execute the code between the curly brackets if the condition is true, and if not it will test the else if condition if that is also false the else code will execute.

```
for(int i = 0; i <  
#repeats; i++){ }
```

Used when you would like to repeat a chunk of code a number of times (can count up i++ or down i-- or use any variable)

DIGITAL

```
pinMode(pin, mode);
```

Used to set a pin's mode, pin is the pin number you would like to address 0-19 (analog 0-5 are 14-19). The mode can either be INPUT or OUTPUT.

```
digitalWrite(pin, value);
```

Once a pin is set as an OUTPUT, it can be set either HIGH (pulled to +5 volts) or LOW (pulled to ground).

```
int digitalRead(pin);
```

Once a pin is set as an INPUT you can use this to return whether it is HIGH (pulled to +5 volts) or LOW (pulled to ground).

ANALOG

The Arduino is a digital machine but it has the ability to operate in the analog realm (through tricks). Here's how to deal with things that aren't digital.

```
int analogWrite(pin, value);
```

Some of the Arduino's pins support pulse width modulation (3, 5, 6, 9, 10, 11). This turns the pin on and off very quickly making it act like an analog output. The value is any number between 0 (0% duty cycle ~0v) and 255 (100% duty cycle ~5 volts).

```
int analogRead(pin);
```

When the analog input pins are set to input you can read their voltage. A value between 0 (for 0 volts) and 1024 (for 5 volts) will be returned.

5. Ηλεκτρονικά του Arduino

Το arduino μόνο του δε μπορεί να κάνει κάτι χρήσιμο. Χρησιμοποιούμε λοιπόν μια σειρά από ηλεκτρονικά εξαρτήματα. Τα βασικά ηλεκτρονικά στοιχεία που χρησιμοποιούνται παρουσιάζονται στις δύο επόμενες σελίδες. **Διαβάστε με προσοχή το “Things to watch out for”, ώστε να μη δημιουργηθεί ζημιά.**

ELECTRONICS IN BRIEF

No previous electronic experience is required to have fun with this kit. Here are a few details about each component to make identifying, and perhaps understanding them, a bit easier. If at any point you are worried about how a component is used or why it's not working the internet offers a treasure trove of advice, or we can be contacted at help@oomlout.com

COMPONENT DETAILS

LED

(Light Emitting Diode)



What it Does:

Emits light when a small current is passed through it. (only in one direction)

Identifying:

Looks like a mini light bulb.

No. of Leads:

2 (one longer, this one connects to positive)

Things to watch out for:

- Will only work in one direction
- Requires a current limiting resistor

More Details:

<http://ardx.org/LED>

Diode



What it Does:

The electronic equivalent of a one way valve. Allowing current to flow in one direction but not the other.

Identifying:

Usually a cylinder with wires extending from either end. (and an off center line indicating polarity)

No. of Leads:

2

Things to watch out for:

- Will only work in one direction (current will flow if end with the line is connected to ground)

More Details:

<http://ardx.org/DIOD>

Resistors



What it Does:

Restricts the amount of current that can flow through a circuit.

Identifying:

Cylinder with wires extending from either end. The value is displayed using a color coding system (for details see next page)

No. of Leads:

2

Things to watch out for:

- Easy to grab the wrong value (double check the colors before using)

More Details:

<http://ardx.org/RESI>

Transistor



What it Does:

Uses a small current to switch or amplify a much larger current.

Identifying:

Comes in many different packages but you can read the part number off the package. (P2N2222AG in this kit and find a datasheet online)

No. of Leads:

3 (Base, Collector, Emitter)

Things to watch out for:

- Plugging in the right way round (also a current limiting resistor is often needed on the base pin)

More Details:

<http://ardx.org/TRAN>

Hobby Servo



What it Does:

Takes a timed pulse and converts it into an angular position of the output shaft.

Identifying:

A plastic box with 3 wires coming out one side and a shaft with a plastic horn out the top.

No. of Leads:

3

Things to watch out for:

- The plug is not polarized so make sure it is plugged in the right way.

More Details:

<http://ardx.org/SERV>

DC Motor



What it Does:

Spins when a current is passed through it.

Identifying:

This one is easy, it looks like a motor. Usually a cylinder with a shaft coming out of one end.

No. of Leads:

2

Things to watch out for:

- Using a transistor or relay that is rated for the size of motor you're using.

More Details:

<http://ardx.org/MOTO>

COMPONENT DETAILS (CONT.)

Piezo Element



What it Does:
A pulse of current will cause it to click. A stream of pulses will cause it to emit a tone.
Identifying:
In this kit it comes in a little black barrel, but sometimes they are just a gold disc.

No. of Leads:
2
Things to watch out for:
- Difficult to misuse.
More Details:
<http://ardx.org/PIEZ>

IC (Integrated Circuit)



What it Does:
Packages any range of complicated electronics inside an easy to use package.
Identifying:
The part ID is written on the outside of the package. (this sometimes requires a lot of light or a magnifying glass to read).

No. of Leads:
2 - 100s (in this kit there is one with 3 (TMP36) and one with 16 (74HC595))
Things to watch out for:
- Proper orientation. (look for marks showing pin 1)
More Details:
<http://ardx.org/ICIC>

Pushbutton



What it Does:
Completes a circuit when it is pressed.
Identifying:
A little square with leads out the bottom and a button on the top.

No. of Leads:
4
Things to watch out for:
- these are almost square so can be inserted 90 degrees off angle.
More Details:
<http://ardx.org/BUTT>

Potentiometer



What it Does:
Produces a variable resistance dependant on the angular position of the shaft.
Identifying:
They can be packaged in many different form factors, look for a dial to identify.

No. of Leads:
3
Things to watch out for:
- Accidentally buying logarithmic scale.
More Details:
<http://ardx.org/POTE>

Photo Resistor

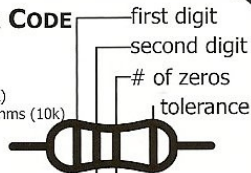


What it Does:
Produces a variable resistance dependant on the amount of incident light.
Identifying:
Usually a little disk with a clear top and a curvy line underneath.

No. of Leads:
2
Things to watch out for:
- Remember it needs to be in a voltage divider before it provides a useful input.
More Details:
<http://ardx.org/PHOT>

RESISTOR COLOR CODE

Examples:
green-blue-brown - 560 ohms
red-red-red - 2 200 ohms (2.2k)
brown-black-orange - 10 000 ohms (10k)



0 - Black	5 - Green	20% - none
1 - Brown	6 - Blue	10% - silver
2 - Red	7 - Purple	5% - gold
3 - Orange	8 - Grey	
4 - Yellow	9 - White	

LEAD CLIPPING

Some components in this kit come with very long wire leads. To make them more compatible with a breadboard a couple of changes are required.

LEDs:
Clip the leads so the long lead is ~10mm (3/8") long and the short one is ~7mm (9/32").




Resistors:
Bend the leads down so they are 90 degrees to the cylinder. Then snip them so they are ~6mm (1/4") long.




Other Components:
Other components may need clipping. Use your discretion when doing so.




6. Κατάλογος εξαρτημάτων (Inventory List)



Για την υλοποίηση των εργαστηριακών ασκήσεων θα χρησιμοποιήσετε μια σειρά από ηλεκτρονικά στοιχεία. Παρατίθεται η πλήρης λίστα με τα στοιχεία που πρέπει να έχετε.




Σε περίπτωση που λείπει κάτι, ενημερώστε τους υπευθύνους άμεσα.




Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	Arduino Uno	
1	Breadboard	
1	Arduino and Breadboard Holder	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	74HC595 Shift Register (16 pins)	
2	2N2222 transistors	
2	Diode Small Signal - 1N4148	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	DC Motor with wires	
1	Small Servo	
1	5V Relay SPDT Sealed	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	TMP36 - Temperature Sensor	
1	Flex Sensor 2.2"	
1	SoftPot Membrane Potentiometer - 50mm	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	USB Cable A to B	
30	Jumper Wires Standard 7" M/M Pack of 30	
1	Mini Photocell	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	LED - RGB Clear Common Cathode 3 color LED	
10	LED - Basic Red 5mm	
10	LED - Basic Yellow 5mm	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
1	Trimpot 10K with Knob	
1	Buzzer - PC Mount 12mm 2.048kHz	
2	Momentary Push Button Switch - 12mm Square	

Ποσότητα	Είδος (Περιγραφή)	Εικόνα
25	Resistor 330 Ohm 1/6th Watt PTH	
25	Resistor 10k Ohm 1/6th Watt PTH	
1	Male Headers - Break Away Headers - Straight	

7. Αποσφαλμάτωση του Arduino

Πολλές φορές θέλουμε να αποσφαλματώσουμε το κύκλωμά μας ή να επιβεβαιώσουμε ότι ένα κομμάτι του λειτουργεί σωστά. Για να το πετύχουμε αυτό, χρησιμοποιούμε τη σειριακή επικοινωνία σε συνδυασμό με εντολές εκτύπωσης στο σειριακό τερματικό.

1. Εμφανίζουμε τη σειριακή οθόνη, πατώντας το εικονίδιο με το μεγεθυντικό φακό στη εργαλειοθήκη με τα εικονίδια (τέρμα δεξιά).
2. Ρυθμίζουμε την ταχύτητα σε 9600 bps
3. Στο sketch του arduino στο `setup()` τοποθετούμε τη γραμμή `Serial.begin(9600);`
4. Στο σημείο που θέλουμε να εμφανίσουμε κάποια τιμή, δίνουμε `Serial.println(value);` ή `Serial.print(value)` (το `ln` κάνει και αλλαγή γραμμής).

Με αυτόν τον τρόπο μπορούμε να αποσφαλματώσουμε βήμα-προς-βήμα όλο το κύκλωμα. Εκτυπώνουμε τις τιμές από τις εισόδους, και τις τιμές που στέλνουμε στις εξόδους και έτσι προσδιορίζουμε το πρόβλημα.

Κατανομή των εργαστηριακών ασκήσεων ARDUINO:

- 1ο εργαστήριο στο arduino.
 - Ασκήσεις: CIRC-01 , CIRC-02, CIRC-03, CIRC-04, CIRC-05
 - Μέγιστος Χρόνος Ολοκλήρωσης: 145 λεπτά
- 2ο εργαστήριο στο arduino.
 - Ασκήσεις: CIRC-06 , CIRC-07, CIRC-08, CIRC-09
 - Μέγιστος Χρόνος Ολοκλήρωσης: 135 λεπτά
- 3ο εργαστήριο στο arduino.
 - Ασκήσεις: CIRC-10 , CIRC-11, CIRC-12, CIRC-13, CIRC-14
 - Μέγιστος Χρόνος Ολοκλήρωσης: 135 λεπτά
- 4ο εργαστήριο στο arduino
 - Ασκήσεις: ARDUINO FINAL
 - Μέγιστος Χρόνος Ολοκλήρωσης: 120 λεπτά

8. Κανόνες εργαστηρίου

1. Πριν χρησιμοποιήσετε οποιοδήποτε εξοπλισμό, θα πρέπει να ακουμπήσετε μια γειωμένη επιφάνεια (π.χ. το κουτί ενός υπολογιστή) για να απομακρυνθεί ο στατικός ηλεκτρισμός. Αυτό θα πρέπει να το επαναλαμβάνετε κατά τακτά χρονικά διαστήματα (π.χ. κάθε 20-30 λεπτά).
2. Απαγορεύεται η απομάκρυνση οποιουδήποτε εξοπλισμού από το εργαστήριο.
3. Να αναφέρετε αμέσως οποιοδήποτε πρόβλημα (χαμένο εξάρτημα, δυσλειτουργία εξαρτήματος) στους υπευθύνους.
4. Απαγορεύεται να σημειώνετε ή να τροποποιείτε μόνιμα τα εξαρτήματα.
5. **Κατά την είσοδο στο εργαστήριο:**
 - a. Ενημερώνετε τον υπεύθυνο για τον εξοπλισμό που θα χρησιμοποιήσετε.
 - b. Υπογράφετε την παραλαβή του εξοπλισμού και των εξαρτημάτων που το συνοδεύουν.
6. **Κατά την έξοδο από το εργαστήριο:**

- a. θα πρέπει να παραδώσετε τακτοποιημένο τον εξοπλισμό (μέσα στις πλαστικές σακούλες/κουτιά), ακριβώς όπως σας παραδόθηκε.
 - b. Ο πάγκος να είναι καθαρός, η οθόνη, ο υπολογιστής, το πληκτρολόγιο και το mouse να είναι τακτοποιημένα.
 - c. Ο υπεύθυνος ελέγχει τον εξοπλισμό που του παραδίδετε και υπογράφει το φύλλο παραλαβής.
7. Πάντα να έχετε κλειστή την τροφοδοσία κατά τη σύνδεση ή αποσύνδεση εξαρτημάτων από μια πλακέτα.
8. Απαγορεύονται χυμοί, νερά, καφέδες, τρόφιμα στο εργαστήριο. Μπορείτε να τα αφήνετε έξω από το εργαστήριο.
9. Μην ασκείτε υπερβολική πίεση κατά τη συναρμολόγηση ενός κυκλώματος. Μπορείτε να χρησιμοποιείτε γειτονικές επαφές/connection points αν δείτε ότι ένα εξάρτημα δεν τοποθετείται χωρίς πίεση.
10. Κάποια εξαρτήματα, όπως η μεμβράνη πίεσης ή το ποτενσιόμετρο επαφής, είναι πολύ ευαίσθητα. Για να τα τοποθετήσετε στο breadboard πρέπει να τα πιάσετε από πολύ χαμηλά (δίπλα στις επαφές).
11. Οι αντιστάσεις και οι δίοδοι που έχουν λυγισμένες επαφές να τις αφήνετε σε αυτή τη μορφή και να μην τις ισιώνετε, διαφορετικά υπάρχει κίνδυνος να κοπούν.
12. Σε περίπτωση που δεν ακολουθήσετε τις υποδείξεις ασφαλείας και προστασίας τόσο του εαυτού σας όσο και του εξοπλισμού, ή αν δεν είστε προσεκτικοί θα υπάρχουν κυρώσεις.
13. Ο υπεύθυνος του εργαστηρίου έχει τον τελευταίο λόγο. Οι υποδείξεις του θα πρέπει να λαμβάνονται σοβαρά υπόψιν.