ΕΡΓΑΣΤΗΡΙΟ Α’

ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΔΙΚΤΥΩΝ ΚΑΙ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ

Π19204 – ΓΕΩΡΓΙΟΣ ΣΕΪΜΕΝΗΣ

# ­­­Άσκηση 1

### Εισαγωγή

Παρακάτω θα εξηγηθούν οι ενέργειες που ακολουθήθηκαν για να λυθεί η πρώτη εργαστηριακή άσκηση (Άσκηση 1) του μαθήματος. **Στόχος της άσκησης ήταν να τροποποιηθούν οι κατάλληλες γραμμές κώδικα, ώστε να εκτυπώνεται το όνομα του χρήστη, αντί για το «Hello World».**

### Βήμα 1ο: Δημιουργία των Κατάλληλων Αρχείων

Το πρώτο βήμα που πρέπει να ακολουθήσουμε, προτού ανοίξουμε το Cooja, είναι να τροποποιήσουμε το αρχείο C κώδικα που εκτελείται. **Γι’ αυτό, θα πάμε στην διεύθυνση «/home/user/contiki-2.7/examples»** και φτιάχνουμε έναν νέο φάκελο, που θα τον πούμε Ε1. **Μέσα σε αυτόν τον φάκελο σιγουρευόμαστε ότι υπάρχουν τα κατάλληλα αρχεία:**

**Graphical user interface, text, application

Description automatically generated**

Εικόνα 1: Παρουσία των κατάλληλων αρχείων

### Βήμα 2ο: Τροποποίηση Κώδικα

Τροποποιούμε κατάλληλα τον κώδικα για να εμφανίζονται μονάχα το όνομά μας κατά την εκτέλεσή του. Αρκεί να αλλάξουμε την γραμμή στην οποία γράφεται το printf();

Graphical user interface, text, application, email

Description automatically generated

Εικόνα 2: Τροποποίηση της κατάλληλης γραμμής κώδικα.

### Βήμα 3ο: Άνοιγμα Cooja και Προσθήκη Κόμβου

Για να δούμε εν δράσει την τροποποίηση της γραμμής αυτής, θα πρέπει να ανοίξουμε το Cooja και να προσθέσουμε έναν κόμβο με το Script αυτό. **Έχοντας φτιάξει, λοιπόν, ένα νέο simulation στο Cooja, πατάμε την προσθήκη ενός Sky Mote. Πάνω σε αυτό το Sky Mote, βάζουμε το Script «ergastirio1.c».** Όταν βρει το αρχείο, μπορούμε να πατήσουμε “compile” για να γίνει η μεταγλώττιση του αρχείου.

**Graphical user interface, application

Description automatically generated**

Εικόνα 3: Αποτέλεσμα πατήματος του κουμπιού compile.

Αφότου προσθέσουμε τον κόμβο, μπορούμε να δούμε την εκτέλεση του προγράμματος, στην οποία, πράγματι, εμφανίζεται το μήνυμα που θέλουμε. Για να δούμε την εκτέλεση του προγράμματος, πατάμε το κουμπί «start».

Graphical user interface

Description automatically generated

Εικόνα 4: Επιτυχής εκτέλεση του προγράμματός μας.

# Άσκηση 1α

### Εισαγωγή

Μετά την πραγματοποίηση της άσκησης 1, δόθηκε μία άλλη άσκηση με ελάχιστα διαφορετικό στόχο. Σε αυτήν την άσκηση θα πρέπει να γράψουμε κώδικα ώστε να ανάψουν LEDs μέσα στους κόμβους μας. **Με την χρήση χρονομετρητών, θα πρέπει να ανάψουν μετά από ένα χρονικό διάστημα τα δύο LED με διαφορετικά χρώματα.** Συγκεκριμένα, για 2 δευτερόλεπτα μόνο το κόκκινο LED, ύστερα για 4 δευτερόλεπτα μόνο το μπλε LED, και τέλος, για ένα δευτερόλεπτο και τα δύο (παράλληλα θα εμφανίζονται και μηνύματα).

### Βήμα 1ο: Δημιουργία των Κατάλληλων Αρχείων

Θα ακολουθήσουμε την ίδια διαδικασία με πριν, κάνοντας τα δύο απαραίτητα αρχεία, όπως και πριν. Αυτήν τη φορά, ας το ονομάσουμε «ergastirio1a.c».

Graphical user interface

Description automatically generated

Εικόνα 5: Τα κατάλληλα αρχεία για την εκτέλεση.

### Βήμα 2ο: Τροποποίηση Κώδικα

Πάμε να δούμε την λογική πίσω από τα ζητούμενα της άσκησης. Για αρχή, θέλουμε τις κατάλληλες βιβλιοθήκες του Contiki, ώστε να έχουμε χρήσιμα εργαλεία για την εκτέλεση του προγράμματος. **Από την στιγμή λοιπόν, που έχουμε να κάνουμε με LEDs, χρειαζόμαστε την “leds.h” και για τους χρονομετρητές, το struct “etimer”.**

Graphical user interface, text, application, email

Description automatically generated

Εικόνα : Εισαγωγή των κατάλληλων μεταβλητών και βιβλιοθηκών.