

ΕΡΓΑΣΙΑ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ ΑΙΣΘΗΤΗΡΩΝ

A.M. : 1115201200106

Όνομα : Βασίλειος

Επίθετο : Μαυρομαμάτης

ΠΑΡΟΥΣΙΑΣΗ ΚΩΔΙΚΑ

Η εργασία αποτελείται από το **parent directory Project2019** το οποίο περιέχει το παρών **README.pdf** αρχείο και το directory με το **netbeans project netbeansProject** και τα απαραίτητα υλοποιημένα συστατικά του. Το directory αυτό περιλαμβάνει **4** υλοποιημένα **middlets**. Τα **SensorNode2019** , **SensorNode2019DIF** που αποτελούν τα middlet για τα nodes που παράγουν τις τυχαίες μετρήσεις θερμοκρασίας χρησιμοποιώντας την **BoxMuller-Method** και η δουλειά τους είναι να τις στέλνουν σε ένα aggregator όπου αποφασίζεται αν θα σταλούν προς το sink και τα 2 streams μετρήσεων ή ένα ενοποιημένο stream για εξοικονόμηση ενέργειας. Το **πρώτο** παράγει τιμές με **μέση τιμή 0.0** και **διακύμανση 1.0**, ενώ το **δεύτερο** έχει **τυχαίες τιμές μέσης τιμής και διακύμανσης**.

Το **τρίτο middlet** είναι το **AggregatorNode2019** το οποίο χρησιμοποιεί **Fast Fourier Transform** για να παρατηρήσει τις αποκλίσεις μεταξύ των δειγμάτων και κατ' επέκταση την ύπαρξη ή όχι event. Λαμβάνει ένα set 8 μετρήσεων απτον κάθε κόμβο και αν δει οτι ξεπερνιέται η τιμή της μεταβλήτης **THRESHOLD** που δίνουμε εμεις τιμή (default = 2) τότε στέλνει και τις 2 ροές και ξανατρέχει το FFT, αλλιώς στέλνει μια ενοποιημένη ροή και για τα επόμενα 8 set μετρήσεων στέλνει ενοποιημένα στην καταβόθρα.

Το **τέταρτο middlet** είναι το **SinkNode2019** το οποίο λαμβάνει τις μετρήσεις απ'τον aggregator κόμβο μαζί με μία υπόδειξη για το αν πρόκειται για ενοποιημένες ή όχι ροές, και τυπώνει τα αντίστοιχα δεδομένα που έλαβε μαζί με αυτή την υπόδειξη.

Και στα 4 middlets παρουσιάζονται (περιορισμένα λόγω καθυστέρησης I/O) διαγνωστικά μηνύματα στο stdout.

Επίσης και τα 4 middlets, τρέχουν επ άπειρων και πρέπει να κλείσουν χειροκίνητα.

COMPILING + EXECUTION

1)Το compiling έχει ήδη πραγματοποιηθεί και το αποτέλεσμα του είναι 2 αρχεία build.xml και emulator_demo.jar.

2)Μετακινούμε το parent directory Project2019 στο path που είναι εγκατεστημένο το Sun directory (πχ C:\Program Files\Sun\SunSPOT\sdk).

3)Ανοίγουμε το **SunSPOT Manager** → **solarium**

4)Στο παράθυρο SpotView που ανοίγει ανοίγουμε την καρτέλα **emulator** → **New virtual spot** και δημιουργούμε 4 αισθητήρες στο grid.

5)Σε όλους τους κόμβους κάνουμε δεξί κλικ στο εικονίδιο τους → **display application output** → **in internal frame** για να μπορούμε να βλέπουμε μηνύματα stdout.

6)Σε όλους τους κόμβους κάνουμε πάλι δεξί κλικ στο εικονίδιο τους → **deploy middlet bundle** → πηγαίνουμε στο path με το netbeans project που αντιγράψαμε στο 2ο βήμα και φορτώνουμε το αρχείο **emulator_demo.jar** που έχει παραχθεί από building του netbeans project.

7)Στον αισθητήρα με διεύθυνση **0a00.020f.0000.1004**

κάνουμε δεξί κλικ στο εικονίδιο → run middlet → **SinkNode2019**.

8)Στον αισθητήρα με διεύθυνση **0a00.020f.0000.1003**

κάνουμε δεξί κλικ στο εικονίδιο → run middlet → **AggregatorNode2019**.

9)Στους υπόλοιπους 2 αισθητήρες με διευθύνσεις **0a00.020f.0000.1002** και **0a00.020f.0000.1001** κάνουμε δεξί κλικ στο εικονίδιο → run middlet → **SensorNode2019** και

ξεκινάμε έτσι τους κόμβους που θα στέλνουν τις μετρήσεις.

10) Τέλος για να τρέξουμε και τον δεύτερο τύπο sensor , κλείνουμε τελείως το SpotView παράθυρο και επαναλαμβάνουμε τα βήματα 4 έως 9 φορτώνοντας το **SensorNode2019DIF** .