**דוח מטלה 2**

**מגישים:** שי לוי אלעד מוצני

**מבנה המחלקות:**

* **S\_I:**

מחלקת עזר לאלגוריתם A. אובייקט שלוקח את הsignal ששייך לMAC ספציפי מהקלט, ומצמיד לו את הindex מהdata (איפה הוא נמצא בטבלה).

בנאי: מקבל signal,index ובונה אובייקט S\_I

S\_IbubbleSort: הפונקציה מקבלת רשימה של אובייקטים S\_I וממיינת אותן בסדר יורד לפי הsignal.

הוספנו גם getters,setters,toString.

* **M\_S:**

מחלקת עזר לקלט באלגוריתם B. המחלקה בונה אובייקט M\_S שמקבלת MAC ואת האות המתאים לו.

בנאי: מקבל כתובת MAC ואות ובונה אובייקט חדש M\_S בעזרת המשתנים שהוכנסו.

בנאי שמקבל רק כתובת MAC: מקבל כתובת MAC ומאתחל את האות כך שיהיה -120 כדי לטפל בכתובת MAC שלא נמצאת בdata.

הוספנו גם getters,setters,toString.

* **Point3D:**

מחלקת עזר לאלגוריתמים. המחלקה בונה נקודה לפי הנתונים.

בנאי ריק: שם בכל הקואורדינטות את הערך -1.0

בנאי: מקבל 3 ערכים של נקודה ויוצר אובייקט Point3D חדש.

הוספנו גם getters,setters,toString.

* **Algorithms:**

המחלקה לוקחת נקודות עם MAC ואותות ומחשבת מיקום משוערך לפי שני האלגוריתמים שבמטלה.

בנאי ריק: קורא את הdata.

getWpoint: האלגוריתם הראשון. מקבל כתובת MAC ורמת דיוק ומחשב את המיקום המשוערך של נקודת הWIFI.

CenterW: פונקציית עזר לשני האלגוריתמים. הפונקציה מקבלת מערך נקודות (Point3D) ומערך של משקלים ( ) ומחשבת ממוצע משוכלל של הנתונים.

GetWlocation: הפונקציה מקבלת מערך של אובייקט M\_S ומשערת את המיקום של המשתמש בעזרת האלגוריתם הבא:

הפונקציה מחלצת מהמידע שקיבלנו רשימה של רשימות שבכל שורה יש רשימה של אובייקט M\_S שמתאימה לקלט.

אם רשימה יוצאת ריקה, הפונקציה מאתחלת את האובייקט עם -120 באות. הפונקציה שולחת כל תת רשימה לפונקציה indeofclosest שמקבלת אות ורשימה של אובייקט M\_S ומחזירה את האינדקס של האובייקט M\_S שהכי קרוב לאות בקלט. לאחר מכן, הפונקציה שולחת כל MAC לאלגוריתם A כדי שנוכל לקבל את המיקום שלו. האלגוריתם מכין מערך נקודות ומערך משקלים ושולח לפונקציה CenterW בשביל לקבל את הנקודה המשוערכת.