arr [] split (**int** value,**boolean** axis){

arr [] ans=arr[4]//O(1)

**if**(axis){

low\_value=first\_X//O(1)

high\_value=max\_X//O(1)

ans [1]=low\_median//O(1)

ans[4]=high\_median//O(1)

**while**(low\_value.get\_x<value AND

value <= high\_value.get\_x)//O(|C|) when C is the number of the elements in the small collection

low\_value=low\_value.next

high\_value=high\_value.prev

**if**(low\_value.get\_x<value)//O(1)

ans[2]=high\_value//O(1)

ans[3]=high\_value.next//O(1)

**else**

ans[2]=low\_value.prev

ans[3]=low\_value

}

**else**{//same run time

low\_value=first\_Y

high\_value=max\_Y

ans [1]=low\_median

ans[4]=high\_median

**while**(low\_value.get\_Y<value AND value <= high\_value.get\_Y)

low\_value=low\_value.next

high\_value=high\_value.prev

**if**(low\_value.get\_Y<value)

ans[2]=high\_value

ans[3]=high\_value.next

**else**

ans[2]=low\_value.prev

ans[3]=low\_value

}

RETURN ans

}

הסבר

1. הפונקציה split מקבלת value וaxis , תפקידה הוא להחזיר 2 אוספים של נקודות ,אוסף אחד ממינימום ועד value ואוסף שני מvalue ועד המקסימום. הפונקציה split מחזירה מערך ans המכיל את תחומי האוספים ,כלומר האוסף הראשון (הקטן מvalue) תחום בans[1] ועד ans[2] . האיברים שבמערך האלה הם מצביעים ל container שהוא איבר במבנה הנותנים הקיים. בעצם על ידי ניצול מבנה הנתונים הקיים אפשר לעבור איבר איבר ברשימה על ידי המעבר בין ה containers . בהתאמה ans[3] וans[4] מכילים את התחום השני וגם כאן אילו מצביעים על containers מהמבנה הקיים.
2. המעבר על האיברים מתבצע על ידי המעבר בתוך מבנה הנתונים שבנינו כיוון הפונקצית הSplit מחזירה מצביעים לאיברים קיימים ושמקושרים בינהם.
3. זמן ריצה: בפונקציה שבנינו ישנם 2 if שמחליטים באיזה ציר אנחנו , בכל אחד מהif יש לולאת while . הלולאה מתחילה להתקדם משני גם מהמקסימום וגם מהמינימום של המבנה נתונים עד אשר תנאי הלולאה לא מתקיים . בשיטה זאת אנו נמשיך להיות בלולאה עד אשר המצביעים לאוסף הקטן יותר ימצאו מה שאפשר לזמן הריצה של פונקציה זאת להיות O(|C|) . בנוסף אפשר לראות את ניתוח הזמן בפסאודו קוד.