**Init()**

האלגוריתם:

-נאתחל את מבנה הנתונים ריק- כלומר שלושה עצים ריקים(כלומר המצביע לעצים), ושלושה int-ים.

סיבוכיות זמן האלגוריתם:

-O(1)- מספר סופי וקבוע של פעולות

לכן בסה"כ סיבוכיות האלגוריתם O(1)

**StatusType AddCourse (void \*DS, int courseID, int numOfClasses)**

האלגוריתם:

1. נוודא את תקינות הקלט ואם לא תקין נחזיר ערך מתאים
2. נוודא שהקורס אינו קיים במבנה שלנו על ידי בדיקה אם המספר קורס מופיע בעץ הקורסים אם מופיע נחזיר שגיאה
3. נוסיף לעץ הקורסים צומת בעלת מפתח זהה למספר קורס וערך של מצביע למבנה "המערך הכפול" שיצרנו.

* הסבר על מבנה "המערך הכפול" שיצרנו:

ביצירת המצביע אנו יוצרים שני מערכים בגודל מספר הכיתות. מאתחלים את המערך הראשון להכיל זמן צפיה לכל שיעור להיות 0, יוצרים רשימה של כל השיעורים שיש בקורס (רשימת השיעורים הלא נצפים) ושומרים מצביע לכל איבר ברשימה במערך השני.

1. הוספת צומת לעץ השיעורים הלא נצפים עם מפתח של מספר הקורס וערך של התחלת רשימת השיעורים הלא נצפים.
2. עדכון משתנים נוספים של המבנה כגון מספר השיעורים הכולל, מספר השיעורים הלא נצפים.

סיבוכיות זמן האלגוריתם:

1. בדיקה של משתנים o(1)
2. בדיקת ערך קיים בעץ AVL ראינו בהרצאה o(log(n)) כאשר n-מספר הקורסים הקיימים כבר במבנה
3. הוספת צומת לעץ AVL כנ"ל o(log(n)) יצירת "המערך הכפול" o(m) כאשר m=מספר השיעורים שיש בקורס, משום שאנחנו מקצים ומאתחלים מערך בגודל m ויוצרים רשימה בגודל m .
4. הוספת צומת לעץ AVL כנ"ל o(log(n))
5. עדכון של מספר משתנים סופי o(1)

* **סה"כ:** o(m+log(n)) כאשר n=מספר הקורסים הקיימים במבנה, m=מספר השיעורים שיש לקורס המוסף.

סיבוכיות מקום האלגוריתם:

אין קריאות רקורסיביות, רק הקצה של שני מערכים בגודל m ויצירה של רשימה מקושרת באורך m לכן

* **סה"כ:** O(m) כאשר m= מספר השיעורים בקורס המוסף

**StatusType RemoveCourse(void \*DS, int courseID)**

האלגוריתם:

1. נוודא את תקינות הקלט ואם הוא לא תקין נחזיר ערך מתאים
2. נמצא את הקורס בעץ הקורסים אם הוא לא נמצא נחזיר ערך מתאים
3. במידה ומצאנו מקודם את הצומת, ניגש למערך השיעורים של הקורס שנמצא בצומת הזאת, עבור כל שיעור שנצפה נשחרר אותו מעץ השיעורים הנפצים.
4. עבור כל שיעור שלא נצפה כבר, נשחרר את האיבר ברשימת השיעורים הלא נצפים, שאליו יש מצביע במערך זה.
5. נשחרר את "המערך הכפול".
6. במידה והיו שיעורים שלא נצפו נשחרר את הצומת של הקורס שהייה בה מצביע לרשימת השיעורים שלא נצפו של אותו קורס.
7. נחזיר הצלחה.

סיבוכיות זמן האלגוריתם:

1. O(1) -בדיקת מספר משתנים סופית
2. O(log(n)) כאשר n מספר הקורסים שיש במבנה- חיפוש בעץ AVL
3. O(m\*log(M)) כאשר m מספר ההרצאות של הקורס וM מספר ההרצאות הכולל במערכת- המקרה הכי גרוע יהיה כאשר כל אחד מהשיעורים שבמנה יהיה בעל זמן צפיה חיובי ממש כלשהו. ולכן לכל שיעור בקורס שנרצה להוציא נאלץ להוציא את האיבר שלו מעץ השיעורים הנצפים וזה יהיה בסיבוכיות של O(logM) מכיוון שזוהי הוצאה מעץ AVL שבמקרה הכי גרוע יש לו M איברים.
4. O(m)-לכל היותר m שיעורים שלא נצפו לכן שחרור של רשימה בגודל m
5. O(m)- שחרור של 2 מערכים בגודל m.
6. O(logn)-הוצאה מעץ(AVL) השיעורים הלא נצפים אשר לכל היותר בגודל n
7. O(1)- פעולה אחת

logM>=logn מכיוון שמספר ההרצאות במערכת בהכרח גדול יותר ממספר הקורסים ולכן

* **בסה"כ**: הסיבוכיות זמן תהיה בסדר גודל של O(mlogM) כנדרש

סיבוכיות מקום של האלגוריתם:

לא נדרשת הקצאה של מקום או פעולה רקוסיבית לכן סיבוכיות מקום: O(1)

**StatusType WatchClass(void \*DS, int courseID, int classID, int time)**

האלגוריתם:

1. נוודא את תקינות הקלט ואם לא תקין נחזיר ערך מתאים
2. נוודא שהקורס קיים במבנה. אם לא נחזיר ערך מתאים
3. נוודא שלקורס הנ"ל יש מספר שיעור כזה על ידי בדיקת כמות מספר הכיתות. אם אין נחזיר ערך מתאים
4. נמצא את הקורס בעץ הקורסים, ונוסיף לזמן הצפיה שלו את הזמן המבוקש.
5. במידה וצפו בו לפני כן, נוציא את הצומת המתאימה לשיעור מעץ השיעורים הנצפים
6. במידה ולא צפו בו לפני כן, נמחק את באיבר ברשימת השיעורים הלא נצפים של אותו קורס בעזרת המצביע לאיבר זה השמור במערך השני.
   1. אם זה היה השיעור הלא נצפה האחרון, נוציא את הצומת של הקורס מעץ הכיתות הלא נצפות
7. נכניס לעץ הכיתות הנצפות את הכיתה הזאת לאחר עדכון זמן הצפייה בה.
8. נחזיר הצלחה (במידה ולא הייתה שגיאת הקצאה)

סיבכיות זמן האלגוריתם:

1. O(1) – בדיקה של מספר סופי של משתנים
2. O(log(n))- חיפוש הקורס בעץ(AVL) הקורסים. כאשר n=מספר הקורסים הקיימים כבר במבנה.
3. O(log(n))- חיפוש הקורס בעץ הקורסים שוב ובדיקה של גודל המערכים שלו בעזרת בדיקת משתנה הגודל של מבנה "המערך הכפול"
4. O(log(n))- שוב חיפוש הקורס בעץ הקורסים ושינוי ערך במערך אחד שנמצא ב"מערך הכפול" בO(1)
5. O(log(M)) בסיכוי הגרוע ביותר כל השיעורים במבנה נמצאים כבר בעץ השיעורים הנצפים ולכן נצטרך להכניס איבר בעץ AVL בגודל M. כאשר M= מספר השיעורים הכולל שנמצא במבנה.
6. O(1)- ניגשים למקום המתאים במערך השני של "המערך הכפול" ומוחקים את האיבר הבודד מהרשימה
   1. O(log(n))- במקרה הכי גרוע לכל אחד מהקורסים יש לפחות שיעור שלא נצפה ולכן עץ השיעורים הלא הצפים יכיל n איברים. וכן כפי שראינו בהרצאה הוצאה של איבר מעץ AVL היא בסיבוכיות הנ"ל. כאשר n=מספר הקורסים הכולל במבנה
7. O(log(M))במקרה הגרוע ביותר כל השיעורים נמצאים בעץ הכיתות הנצפות לכן הכנסה של איבר לעץ AVL שכזה היא זאת.
8. פעולה אחת

**StatusType TimeViewed(void \*DS, int courseID, int classID, int \*timeViewed)**

האלגוריתם:

-נבדוק את תקינות הפלט- אם הוא לא תקין נחזיר את הערך המתאים

-נמצא את הקורס בעץ הקורסים- אם הוא לא נמצא נחזיר את הערך המתאים

-נפנה לאיבר של השיעור במרך המתאים לקורס, עבור השיעור הזה נפנה בעזרת המצביע ששמור איתו לאיבר המתאים לו בעץ השיעורים וממנו נמצא את זמן הצפייה הכולל ונשים אותו לתוך המשתנה הנדרש

-נחזיר הצלחה(לכל allocation שאנחנו עושים כמובן נוודא שהיא מצליחה ואם לא נחזיר ערך מתאים)

סיבוכיות זמן האלגוריתם:

-O(1)- מספר סופי קבוע של פעולות יחידות

-O(logn) כאשר n מספר הקורסים במערכת- חיפוש בעץ AVL בגודל n

-O(1)- מספר סופי קבוע של פעולות יחידות

-O(1)- מספר סופי קבוע של פעולות יחידות

לכן סה"כ הסיבוכיות זמן תהיה O(logn) כנדרש.

**StatusType GetMostViewedClasses(void \*DS, int numOfClasses, int \*courses, int \*classes)**

**void Quit(void \*\*DS)**

האלגוריתם:

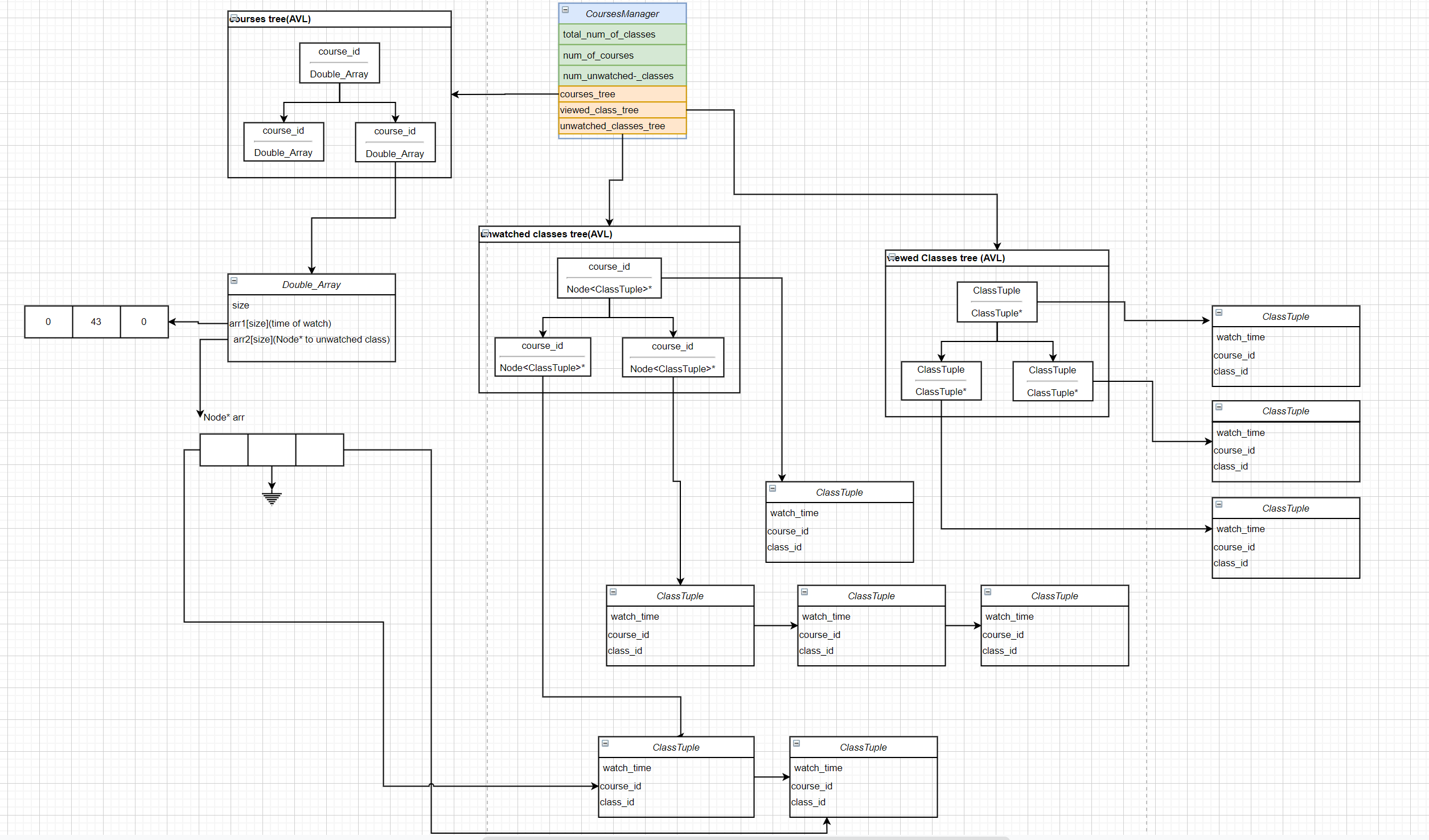
* נמחק את עץ הקורסים-נעבור על עץ הקורסים,לכל איבר בו נמחק את כל אחד מהקורסים שבו ואז את המערך ואז נמחק את העץ
* נמחק את עץ השיעורים- נעבור על כל איבר ונמחר את הרשימה שבו ואז נמחק את העץ
* נמחק את שאר המצביעים

סיבוכיות זמן האלגוריתם:

-O(n+m)- בסך הכל אנחנו עוברים פעם אחד לכל שיעור ומבצעים עליו מספר סופי של פעולות ואז פעם אחת לכל קורס ומבצעים עליו מספר סופי של פעולות סה"כ לכן במקרה אחי גרוע מספר הפעולות יהיה בסדר גודל של O(n+m)

O(m)-- בסך הכל אנחנו עוברים פעם אחד לכל שיעור במקרה הכי גרוע, ומבצעים עליו מספר סופי של פעולות ולכן במקרה אחי גרוע מספר הפעולות יהיה בסדר גודל של O(m)

-O(1) – מספר סופי קבוע של פעולות

תרשים סכמתי של המבנה במצב ממוצע: