**דוח פרויקט: מערכת ניהול חברת אוטובוסים**

**רותם אברהם- 322438045  
תמר קינן- 209335538**

**1. הקדמה:**

מערכת ניהול חברת האוטובוסים היא מערכת אשר מכילה 8 מבנים: בן אדם, נהג, אוטובוס, קו, תחנה, נסיעה, זמן, וניהול החברה.   
קיימים גם מבנים נוספים כללים (חברה לקובץ, קובץ עזר, כללי, רשימה ומאקרו) שמשמשים כפונקציות עזר כלליות חוזרות לכל מיני מבנים.

מטרת המערכת ניהול חברות אוטובוסים ע"י ניהול מערך נהגים ואוטובוסים, רשימות של כל תחנות וקווי החברה ולייצר בעזרתם מערך נסיעות שחלקן ציבוריות וחלקן פרטיות.

המערכת מאפשרת את הפעולות הבאות:

* הוספת נהג
* הוספת אוטובוס
* הוספת קו נסיעה
* הוספת תחנה
* הוספת נסיעה
* מיון או חיפוש נסיעה מסוימת
* הוספת\הורדת תחנה מנסיעה מסוימת
* מציאת הנסיעה הטובה ביותר

במערכת זו עשינו שימוש בפולימורפיזם במבנה בשם Transport המכיל בתוכו 2 בנים (2 סוגי נסיעה): Private ו- Public.

כל נתוני המערכת נטענים לפי בחירת המשתמש מקובץ טקסט או קובץ בינארי ובמידה ואין קובץ ישמור נתונים ידני, בסוף השימוש במערכת הנתונים ישמרו לשני הקבצים.

**2. תיאור מבני המערכת:**

* **בן אדם-**

1. שם פרטי- מחרוזת בגודל לא ידוע
2. מספר תעודת זהות- ייחודי לכל בן אדם
3. סוג- Enum מסוג: גבר\אישה

מבנה זה מייצג אנשים במערכת. לכל בן אדם יש להזין תעודת זהות תקנית וייחודית ובדקנו זאת על ידי פונקציות מתאימות ובנוסף לשמור את הסוג שלו והשם שלו.

* **נהג-**

1. בן אדם- מצביע למבנה מטיפוס אדם
2. משכורת
3. מספר עובד- מספר ייחודי לכל נהג

מבנה זה מנהל מידע על הנהג: לכל נהג בנוסף לתכונות של בן אדם, יש משכורת אשר מחושבת לפי כמות הנסיעות אליהן הוא משויך אליה ומספר עובד שלו בחברת האוטובוסים.

* **אוטובוס-**

1. שם חברת הרכב- מחרוזת בגודל לא ידוע
2. סוג האוטובוס - Enum מסוג: מיניבוס\רגיל\אקורדיון\תיירות

במבנה זה לכל אוטובוס יש את שם חברת הרכב למשל קיה, טויוטה וכו, ובנוסף את סוג האוטובוס.

* **קו-**

1. אוטובוס- מצביע למבנה מטיפוס אוטובוס
2. מספר קו- ייחודי אשר מוגרל באמצעות Random
3. זמינות- Enum מסוג: בשימוש\לא בשימוש.

במבנה זה לכל קו משוייך אוטובוס ושמור לו מספר קו יחודי (בדקנו זאת על ידי פונקציה מתאימה) והאם הוא זמין או לא, כלומר האם הוא כבר משוייך לנסיעה (לא זמין) או שעדיין לא שוייך לנסיעה (זמין).

* **תחנה-**

1. שם תחנה- מחרוזת בגודל קבוע (7 תווים)
2. מספר תחנה ייחודי לכל תחנה

במבנה זה לכל תחנה יש שם ייחודי אשר שמור על ידי 2 רחובות למשל דיזינגוף/גורדון ומספר תחנה ייחודי (נבדק ע"י פונקציות מתאימות).

* **נסיעה-**

1. זמן יציאה- מבנה מטיפוס זמן.
2. זמן הגעה- מבנה מטיפוס זמן.
3. תעריף נסיעה
4. רשימת תחנות- רשימה המכילה בתוכה מצביע למבנה מטיפוס תחנה
5. סוג- Enum מסוג: ציבורית\פרטית
6. Union של שני מבנים: פרטית וציבורית

מבנה הנסיעה אחראי על תכנון נסיעה. נסיעה יכולה להיות פרטית או ציבורית ותשמור שעת יציאה מבוקשת, שעת הגעה לפי חישוב מספר התחנות שמכילה, תעריף נסיעה שמחושב על ידי מספר התחנות וסוג הנסיעה ואת רשימת התחנות המשוייכות לנסיעה זו. רשימה התחנות תהיה ממויינת לפי האות הראשונה של התחנה הראשונה.   
נהג ישמר לנסיעה אם מדובר בנסיעה פרטית וקו ישמר אם מדובר בנסיעה ציבורית.   
נסיעה מחוייבת לכלול לפחות 2 תחנות (מוצא ויעד) וניתן להוסיף תחנות לנסיעה (במידה וירצו להוסיף תחנה שכבר קיימת בנסיעה לא יהיה ניתן להוסיף אותה שוב) וגם ניתן להוריד תחנות מנסיעה (כמובן אך ורק אם התחנה קיימת בנסיעה).  
ניתן להשוות בין נסיעות לפי התעריף שלהן, זמן היציאה וזמן ההגעה.

* **נסיעה פרטית-**

1. משתנה מטיפוס נהג

* **נסיעה ציבורית-**

1. משתנה מסוג קו

* **זמן-**

1. שעה מסוג int
2. דקות מסוג int

מבנה זה נועד בשביל מבנה תכנון הנסיעה שבו יש לשמור שעת יציאה. השעה תיבדק על ידי פונקציות מתאימות שאכן הוזנה נכון. ניתן להשוות בין הזמני הגעה ויציאה של נסיעה, ובנוסף עבור כל הוספת תחנה לנסיעה זמן ההגעה ליעד יעלה לפי חישוב מספר דקות לתחנה (3 דקות עבור כל תחנה).

* **ניהול החברה-**

1. שם החברה- מחרוזת בגודל לא ידוע
2. מערך הנהגים- מערך של מבנים מטיפוס נהג
3. מספר הנהגים- גודל מערך הנהגים
4. מערך האוטובוסים- מערך של מבנים מטיפוס אוטובוס
5. מספר האוטובוסים- גודל מערך האוטובוסים
6. מערך הנסיעות- מערך של מצביעים למבנה מטיפוס נסיעה
7. מספר הנסיעות- גודל מערך הנסיעות
8. רשימת הקווים- רשימה המכילה בתוכה מצביע למבנה מטיפוס קו
9. רשימת התחנות- רשימה המכילה בתוכה מצביע למבנה מטיפוס תחנה
10. משתנה מיון- Enum הקובע על פי מה יתבצע מיון מערך הנסיעות

מבנה ניהול החברה שומר את כל נתוני החברה ואחראי על כל הפונקציות הראשיות של המערכת. במבנה זה קיימים כל המערכים והרשימות של המבנים, וניתן להוסיף בו לכל מערך: הוספת נהג, קו, אוטובוס, נסיעה, ותחנה.   
במבנה זה קיימות כמה פונקציות ראשיות:  
- בכל הוספת קו נוסיף אותו לרשימת הקו בצורה ממויינת לפי מספר הקו (בסדר עולה), וכן בכל הוספת תחנה נוסיף אותה לרשימת התחנות בצורה ממויינת לפי האות הראשונה של הרחוב הראשון.  
- בחירת נסיעה ולהוסיף או להוריד לה תחנה.   
- מיון רשימת הנסיעות לפי זמן יציאה,הגעה ותעריף ולאחר מכן למצוא נסיעה לפי אחת מתכונות אלו.  
- הדפסת כל הקווים עם סוג אוטובוס מסיום.  
- מציאת הנסיעה הכי טובה- פונקציה אשר מקבלת 2 תחנות מוצא ויעד ומוצאת את הנסיעה עם המספר המינימלי ביותר של תחנות בין המוצא והיעד.

**3. שרטוט והסבר על אופן הדחיסה של המבנה זמן הדחוס בקובץ הבינארי**

הדחיסה בוצעה עבור המחלקה Time כאשר היא מכילה משתנה שעה עבורו יש 23 ערכים אפשריים ולכן נשתמש ב-5 סיביות ועבור המשתנה דקות עבורם יש 59 ערכים אפשריים ולכן נשתמש ב-6 סיביות.

בסך הכל 11 סיביות ולכן נשתמש ב-2 בתים.

נשמור את כל 5 הסיביות של השעה בסיביות הגבוהות בבית הראשון ואת 3 הסיביות של הדקות בסיביות הנמוכות של הבית הראשון , ואת 3 הסיביות הנמוכות של הדקות נשמור בסיביות הכי גבוהות של הבית השני.

בזיכרון יראה כך:

hour (5 bits) minutes (6 bits)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Byte 1 Byte 2

**4. הסבר על כל פעולות אותן המשתמש יכול לבצע במערכת:**

**1. הוספת אוטובוס-**

המשתמש יצטרך להזין שם חברת הרכב ולבחור סוג רכב מאחת האופציות: מיניבוס, רגיל, אקורדיון, תיירות ויתווסף האוטובוס החדש למערך האוטובוסים.

**2. הוספת נהג-**

המשתמש יצטרך להזין פרטי בן אדם: מספר תעודת זהות- שנבדק שהוא ת.ז ישראלי תקין וכן ייחודי, שם פרטי וסוג בן אדם: גבר או אישה. ובאתחול הנהג המערכת תשמור לו מספר עובד באופן אוטומטי ובכל הוספת נהג יגדל מספר העובד ב-1. בנוסף באתחול המשכורת של הנהג תהיה מאופסת עד שישייכו אותו לנסיעה (פרטית- עבור כל נסיעה התשלום מוגדר כקבוע של 100 שקלים).  
לבסוף יתווסף הנהג החדש למערך האוטבוסים.

**3. הוספת קו-**

המשתמש יצטרך לבחור תחילה אוטובוס שיהיה משוייך לקו לפי מספר המופיע לצד האוטובוס בהדפסה. באתחול הקו המערכת תגריל לו מספר קו ותוודא שבכל הוספת קו למערכת המספר שלו ייחודי, ותאתחל את הזמינות שלו תחילה לזמין עד שישויך לנסיעה וישתנה ללא זמין. לבסוף יתווסף הקו לרשימת הקווים של החברה בצורה ממויינת לפי מספר הקו בסדר עולה.

**4. הוספת תחנה-**

המשתמש יצטרך להזין את שם התחנה תחילה רחוב אחד ואז רחוב שני. (לא ניתן להזין מספרים בשם התחנה) ולא ניתן שהשם יורכב מאותו רחוב פעמיים.  
1. במידה והמשתמש יזין רחוב המורכב מילה אחת : המערכת תיקח את 3 האותיות הראשונות ותמיר אותן לאותיות גדולות. למשל: dizingof -> DIZ  
2. במידה והמשתמש יזין רחוב המורכב מ- 2 מילים: המערכת תיקח את האות הראשונה של המילה הראשונה ואת האות הראשונה והאחרונה של המילה השנייה. למשל ben Yehuda-> BYA.  
3. במידה והמשתמש יזין רחוב המורכב מ3 מילים ומעלה: המערכת תיקח את האות הראשונה מכל מילה. שהמערכת תתתווסף התחנה לרשימת כל התחנות של החברה בצורה ממויינת לפי האות הראשונה של המילה הראשונה בשם שלה. למשל Sderot ben Gurion-> SDB

לאחר הזנת 2 הרחובות לשם התחנה המערכת תחבר אותן ותוסיף "/" בינהם. למשל: DIZ/BYA  
לאחר מכן המשתמש יצטרך להזין מספר תחנה בין 1000- 9000 המערכת תוודא שמספר התחנה ייחודי.  
לבסוף תתווסף התחנה לרשימת התחנות של החברה בצורה ממוינת לפי סדר ABC של המילה הראשונה בשם התחנה.

**5. הוספת נסיעה-**

המשתמש יבחר האם רוצה להוסיף נסיעה פרטית או ציבורית, במידה ותהיה פרטית, יצטרך לבחור לו נהג ובמידה ותהיה ציבורית יצטרך לבחור קו שאינו בשימוש. עבור כל הוספת נסיעה יבחר את שעת היציאה המבוקשת לפי המבנה הבא: HH:MM

על מנת לאתחל נסיעה יצטרך להוסיף 2 תחנות (מוצא ויעד) ואם ירצה להוסיף תחנות יעשה זאת באופציה נוספת בתפריט. המערכת תחשב את משך זמן הנסיעה לפי הוספת מספר דקות לכל תחנה בנסיעה (3 דקות עבור כל תחנה). וכך תחשב את זמן ההגעה של הנסיעה.   
לבסוף תתווסף הנסיעה למערך הנסיעות.

**6. הוספת תחנה לנסיעה ומחיקה תחנה מנסיעה-**

תחילה המשתמש יבחר לאיזו נסיעה ירצה להוסיף או למחוק תחנה.   
- מחיקת תחנה: לא ניתן למחוק תחנה אם הנסיעה כוללת 2 תחנות או פחות ובנוסף המערכת תוודא שהתחנה אכן נמצאת בנסיעה.   
- הוספת תחנה: אם ירצה להוסיף תחנה לנסיעה, המערכת תוודא שתחנה זו לא קיימת כבר בנסיעה.   
אם התווספה או נמחקה תחנה, המערכת תחשב את זמן ההגעה והתעריף של הנסיעה מחדש. התחנה תתוסף לרשימת התחנות של הנסיעה שבחר לפי סדר ABC. של המילה הראשונה של התחנה.

**7. מציאת הנסיעה הכי טובה-**

המשתמש יבחר 2 תחנות, מוצא ויעד. המערכת תחפש האם קיימת נסיעה בהן 2 תחנות אלו נמצאות, אם קיימת נסיעה כזו המערכת תמצא את הנסיעה בה מספר מסלול הנסיעה בין 2 תחנות אלו הוא המינימלי ביותר.

**8. הדפסת קווים לפי סוג האוטובוס המשוייך אליהם-**

המשתמש יבחר לפי איזה סוג אוטובוס (מיניבוס, רגיל, אקורדיון, תיירות),והמערכת תדפיס (במידה ויש) את כל הקווים בחברה עם סוג אוטובוס זה.

**9. מיון נסיעה-**

קיימות 3 אפשרויות למיין את מערך הנסיעות, המשתמש יבחר לפי איזו תכונה של נסיעה הוא ירצה למיין:

1. זמן יציאה
2. זמן הגעה
3. תעריף נסיעה

פונקציה זו עושה שימוש בפונקציית מיון qsortשממיינת את מערך הנסיעות.

**10. מציאת נסיעה-**

פעולה זו תוכל להתבצע רק ואך ורק אם בוצע מיון למערך הנסיעות לפני כן.  
לאחר שבחר את התכונה יוכל לחפש נסיעה לפי התכונה שבחר למיין בעזרת שימוש בפונקציה bsearch.

**11. הדפסת כל התחנות במערכת-**

המערכת תדפיס את כל התחנות הנמצאות ברשימת התחנות בצורה ממויינת לפי האות הראשונה של המילה הראשונה בשם התחנה

**12. הדפסת כל המערכת-**

המערכת תדפיס את את שם החברה  
מספר הקווים ואת רשימת הקווים בצורה ממויינת לפי מספר הקו  
מספר התחנות ואת רשימת התחנות בצורה ממויינת לפי האות הראשונה של המילה הראשונה בשם התחנה  
מספר האוטובוסים ומערך האוטובוסים  
מספר הנהגים ומערך הנהגים  
מספר הנסיעות ומערך הנסיעות

**5. שרטוט המערכת:**

A screenshot of a computer screen

Description automatically generated

\*השרטוט עבר שינויים מההגשה הקודמת

**6. חלוקת האחריות:**

את העבודה כולה ביצענו ביחד ולכן לא חילקנו ביננו אחריות .