

מגשים:

עידו אנגלנדר, ת"ז: 207004870

אלירז מדר דביר, ת"ז: 313524159

הקדמה:

יצרנו מערכת המיועדת לניהול מעקב אחר רשת חניונים. במערכת קיים מנהל חניונים אשר שומר כרשימה מקושרת את כל מערך החניונים. כל חניון שומר בתוכו מערך של לקוחות ומערך של חניות. כל חניה יכולה להיתפס על ידי כל לקוח בעזרת מצביע לרכב שתפס את החניה. לכל לקוח קיים מערך של רכבים, מערך של תשלומים, ומנוי לחניון (המספק הנחה אם המנוי פעיל).

בהרצת התוכנית, בתפריט הראשי, ניתן לנהל את מערך החניונים. בעת בחירה של חניון (על ידי ID), נפתח תפריט נוסף שבו ניתן להוסיף ולהסיר לקוחות מהחניון, תפיסה או פינוי של חניה, הוספת מקום חניה לחניון ועוד פעולות המפורטות בהמשך.

המבנים הקיימים במערכת:

ParkingLotManager - המבנה הראשי במערכת המכיל: שם, רשימה מקושרת של כל החניונים הקיימים, lastUsedParkingLotId (לצורך יצירת מזהה ייחודי של חניון).

ParkingLot – חניון המכיל: מזהה ייחודי, מחיר שעת, LastCustomerId (לצורך יצירת מזהה ייחודי של לקוח), מערך של לקוחות ומערך של מקומות חנייה.

ParkingCustomer – לקוח המכיל: מזהה ייחודי, שם, מנוי, מערך של מצביעים לרכבים, מערך של תשלומים.

ParkingPayment – תשלום המכיל: מידע על האם החשבון שולם, רכב (לא כמצביע), תאריך כניסה ותאריך יציאה, תעריף שעותי (הכולל בתוכו הנחה אם יש), כמות ששולמה.

ParkingSubscription – מנוי המספק הנחה. מכיל: האם פעיל או לא, תאריך התחלה, תאריך סיום, אחוז הנחה.

Vehicle – רכב המכיל: לוחית רישוי במבנה וגודל קבוע, סוג הרכב מטיפוס VehicleType שהוא enum שיצרנו.

ParkingSpace – מקום חנייה המכיל: מספר חניה, מערך של סוגי הרכבים שיכולים להחנות במקום, מצביע לרכב שחונה במקום החניה (NULL אם אין רכב שחונה).

מצורפת דיאגרמה מעודכנת בעמוד 4.

פעולות ראשיות שניתן לבצע בתפריטים הקיימים במערכת:

בתחילת התוכנית ובסיומה טעינה/שמירה של כל המידע מקובץ בינארי או טקסט לפי `BIN_FILE_NAME` או `TEXT_FILE_NAME`.

תפריט ראשי:

1. הצגה של כל החניונים הקיימים
2. סיכום סך כל התשלומים
3. הוספת חניון
4. הסרת חניון
5. בחירת חניון להצגת תפריט ניהול

תפריט ניהול חניון:

1. הצגת פרטי חניון.
2. שרטוט דיאגרמה של החניון, לפי סוג הרכב שחונה במקום החניה :

```
Parking Lot ID: 1
Hourly Rate: $35.00
Number of Parking Spaces: 8
Number of Occupied Parking Spaces: 4
[ ] [M] [ ] [E] [ ]
[ ] [C] [T]
```

- [C] - רכב
 - [M] - אופנוע
 - [T] - משאית
 - [E] - רכב חשמלי
 - [X] - סימן ברירת מחדל
 - [] - מקום ריק
3. הצגת רשימה של כל מקומות החנייה
 4. הוספת מוקם חנייה
 5. הסרת מקום חנייה
 6. בחירת מקום חנייה והצגת תפריט ניהול
 7. הצגת כל הלקוחות
 8. הוספת לקוח
 9. הסרת לקוח
 10. בחירת לקוח והצגת תפריט ניהול
 11. מיון לקוחות
 12. חיפוש לקוחות
 13. הוספת רכב ללקוח (האופציה קיימת רק בתפריט הזה, הסבר בהמשך)
 14. הסרת רכב מלקוח (האופציה קיימת רק בתפריט הזה, הסבר בהמשך)
 15. הכנסה של רכב לחניון (החניית הרכב במקום החנייה הספציפי)
 16. הוצאה של רכב מהחניון (כולל תשלום במידה ויש)

תפריט ניהול מקום חנייה:

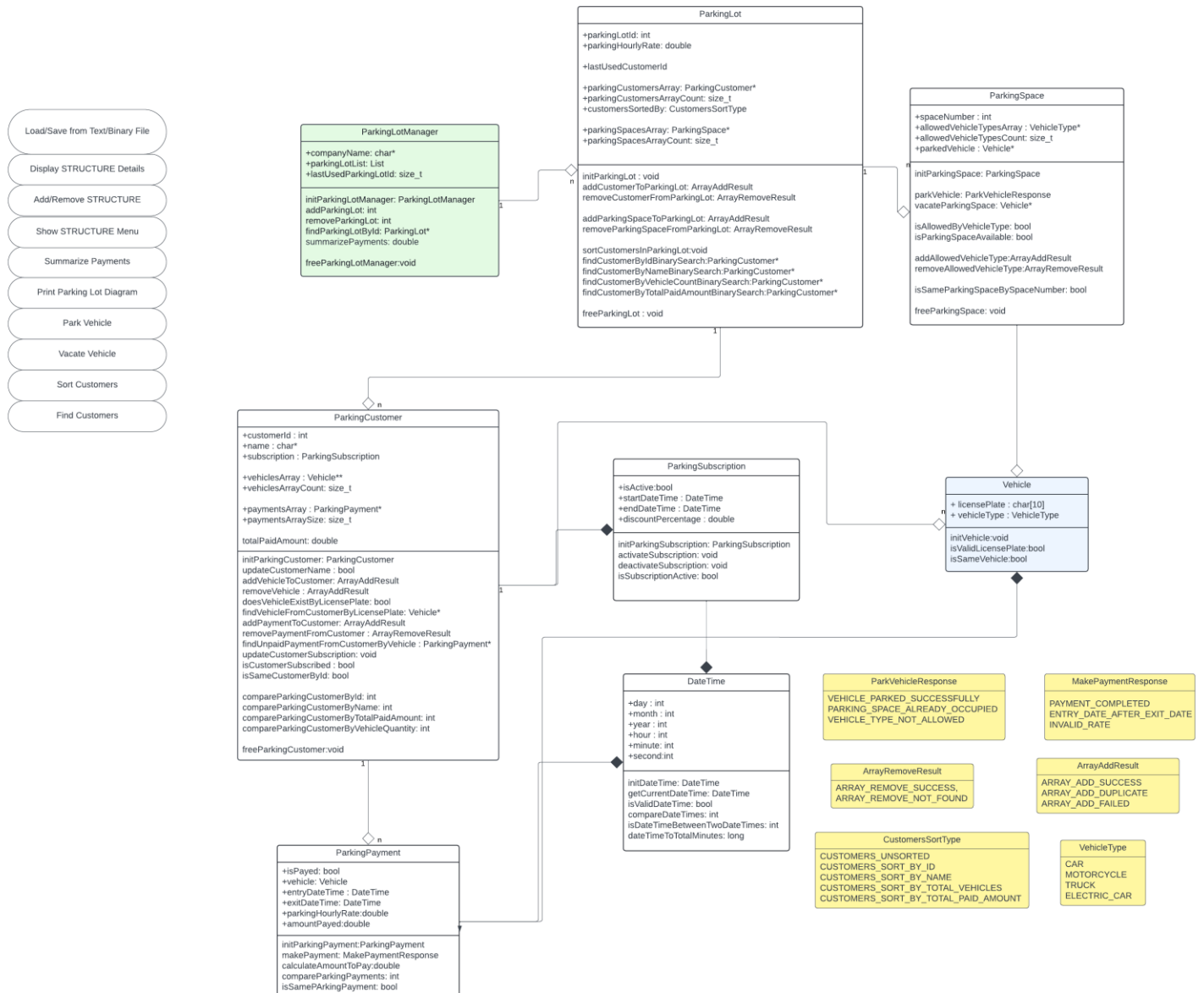
1. הצגת מידע על מקום החנייה
2. הוספת סוג רכב מותר לחנייה
3. הסרת סוג רכב מותר לחנייה

תפריט ניהול לקוח:

1. הצגת פרטי לקוח
2. עדכון פרטי לקוח (שם הלקוח)
3. הצגת פרטי מינוי של לקוח
4. עדכון פרטי המנוי (כולל תוקף המנוי והאם פעיל)
5. הפעלת המנוי (המנוי עדיין צריך להיות בתוקף מבחינת תאריך וזמן כדי לעבוד)
6. השבתת המנוי
7. הצגת כל הרכבים של הלקוח
8. הצגת כלל התשלומים של הלקוח (כולל תשלומים שעוד לא שולמו)
9. הסרה של תשלום של הלקוח

פירוט נוסף:

1. המערכת מניחה כי המשתמש הכניס מספר כאשר הוא התבקש להכניס מספר. (הנחיה בקורס)
2. המערכת משתמשת ב- `size_t` המתנהג כמו `unsigned long` עבור גדלים של מערכים.
3. בחניון אין רכבים מזדמנים, על לקוח להירשם לפני שהוא נכנס עם רכב לחניון.
4. לקוח יכול להכניס יותר מרכב אחד לחניון, גם אם יש לו דרישת תשלום פתוחה.
5. לא ניתן לקשר רכב ליותר מלקוח אחד.
6. לקוח לא יכול להוציא רכב מהחניון לפני שסגר את דרישת התשלום עבור אותו רכב.
7. ישנה האופציה להשתמש בזמן המקומי של המחשב כאשר המשתמש מתבקש להכניס תאריך וזמן.
8. Subscription - מינוי לחניון שמספק הנחה באחוזים. לקוח לא חייב להיות בעל מנוי פעיל כדי להחנות בחנייה, והוא פשוט ישלם מחיר מלא. על המנוי להיות בתוקף מבחינת תאריך וזמן ולהיות במצב פעיל כאשר הרכב נכנס כדי שההנחה תכנס לתוקף. ההנחה נשארת תקפה עבור הרכב עד אשר הרכב יוצא מהחניון, גם אם בוטל המנוי. ניתן לשנות את אחוז ההנחה. במקרה של שינוי, ההנחה תהיה בתוקף רק עבור כניסות עתידיות.
9. ניתן להוסיף רכב ללקוח רק דרך התפריט של החניון. הסיבה היא כי יש לבדוק שאותו רכב לא קיים כבר אצל לקוח אחר.
10. ניתן להסיר רכב מלקוח רק דרך התפריט של חניון. הסיבה היא כי אותו רכב יכול להיות מקושר למקום חנייה, ובמקרה כזה לא ניתן להסיר את הרכב. תשלומים של הרכב ישמרו בלקוח (מכיוון שהרכב לא נשמר כמצביע)
11. ישנה התעלמות מהערות קומפילציה C6031, כפי שהונחנו בקורס.
12. לוחית רישוי מורכבת מאותיות גדולות או ספרות בלבד.
13. הערך של `LICENSE_PLATE_LENGTH` הוא 10. בגלל הכיווץ הבינארי ערך גדול יותר לא יתכווץ טוב (פירוט בהמשך).
14. כל החניונים מתנהלים תחת מבנה של `ParkingLotManager` (סעיף 1.1). החניונים כולם נשמרים כרשימה מקושרת בתוך `ParkingLotManager` (סעיף 1.2)
15. הקריאה/כתיבה לקבצים עלולה להפוך את הסדר בו נשמרו מערכים. לכן לאחר קריאת מערך המשתמשים, מתבצע מיון מחדש לפי הסדר המקורי.



הדיאגרמה מצורפת גם כקובץ [Template Diagram.pdf](#) לקריאה נוחה.

אופן דחיסת מבנה רכב:

האלגוריתם פועל על ידי דחיסת מחרוזת לוחית הרישוי וסוג הרכב למספר שלם מסוג unsigned long long (64-ביט), ובאופן דומה מחלץ את המספר בחזרה למחרוזת לוחית רישוי וסוג רכב.

1. כל תו בלוחית רישוי מכוון ל-6 ביטים:

a. אותיות גדולות (A-Z) נשמרות כערכים 0-25.

b. ספרות (0-9) נשמרות כערכים 26-35.

בסה"כ 36 אפשרויות אשר תופסות 6 ביטים.

2. סוג הרכב נשמר ב-2 הביטים האחרונים של המספר הדחוס.

64 לפי הגודל של unsigned long long	סה"כ ביטים זמינים
2	ביטים עבור סוג הרכב (VehicleType)
$(64-2)=62$	ביטים עבור לוחית רישוי
6 $(36 \text{ ערכים אפשריים: } 26 \text{ אותיות גדולות} + 10 \text{ מספרים})$	ביטים עבור כל תו
10 $(62/6 \text{ ללא שארית})$	אורך מקסימלי ללוחית רישוי
2 $(64 - 10 * 6 - 2)$	ביטים שאינם בשימוש

למשל עבור לוחית רישוי "AB523CDG79" וסוג רכב VT כלשהו, החלוקה תראה כך:

Byte		Byte		Byte		Byte		Byte		Byte		Byte	
A	B	5	2	3	C	D	G	7	9	-	VT		

(Byte = 8 Bits)

חלוקת אחריות בפרויקט

פרטים באחריות עידו:

- מבנה ארכיטקטורת המערכת כולל תפריט ראשי
- שרטוט המערכת
- המבנים: ParkingSpace, Vehicle, ParkingLot, ParkingLotManager

פרטים באחריות אלירז:

- מבנה הדפסה/קליטה מהמשתמש
- פונקציות שמירה/טעינה מקבצים כולל דחיסת מבנה רכב
- המבנים: ParkingCustomer, ParkingSubscription, ParkingPayment, DateTime

כל אחד היה אחראי על כל הפעולות הנגררות משם המבנה, כולל פעולות בקבצים של View ושל Repository.