תיאור הבעיה:

ישנם רכבים בתוך מבוך שאינו ידוע מראש.  
כל רכב מתחיל בנקודת התחלה שונה, ומקבל נקודת סוף שונה, אליה הוא רוצה להגיע.  
כל רכב סורק את המבוך, ומנסה לעדכן בזמן אמת את תכנון המסלול שלו בהתחשבות במכשולים חדשים.  
לרכב יש מידע מדויק אודות איפה הוא נמצא ביחס לנקודת ההתחלה שלו.  
כאשר 2 רכבים קרובים מספיק זה לזה, הם יכולים לשתף מידע שהם צברו אודות המבוך.  
המטרה היא לכתוב אלגוריתם שמביא כל מכונית לנקודת הסוף, בזמן מינימלי.

מדריך למשתמש:

ניתן להכניס מבוך כרשימת קווים פוליגונליים בתוך קובץ json,  
באותו הקובץ ניתן לבחור את מספר המכוניות, ונקודות התחלה וסיום של כל אחת מהמכוניות (קיימת קבוצת מבוכים לדוגמה בתיקייה mazes)  
התוכנה בונה סימולציה של תנועת המכוניות במבוך והנסיון שלהן להגיע לנקודת הסיום.  
ניתן לבקש מהתוכנה לבחור באופן אקראי את אחד מהארגומנטים, או לבחור מצב דיפולטי נתון.

מדריך למתכנת:

הפרויקט בנוי ממספר קבצים:  
הקובץ מכיל את כל הקבועים של האלגוריתם, מחולקים לפי קטגוריות.  
הקובץ מכיל פונקציות עזר, בעיקר לחישובים גיאומטרים ולדיסקרטיזציה של הסביבה.  
הקובץ המרכזי PRM שפותר את הבעיה לרכב אחד במבוך ידוע מראש.  
הספרייה mazes מכילה מספר מבוכים דיפולטים, עם נקודות התחלה וסיום מוגדרות מראש.  
הקובץ בונה מפה שהיא קירוב של המבוך – בעזרת הסריקות שהמכונית סורקת.  
הקובץ מכיל את כל הנתונים לגבי הסביבה, המכוניות בה והמבוך.  
הקובץ מממש מחלקה של מכונית, ואחראי על כל הנתונים של המכונית, עדכון וקבלת ההחלטות שלה.  
הקובץ מכיל פונקציות שעושות שימוש ב על מנת לבנות את המבוך בסביבה.  
הקובץ מממש מחלקה של גרף ממושקל, ותומך בפעולות של הורדת והוספת קשתות וקודקודים.  
הקובץ מממש את אלגוריתם ה-

הקובץ מכיל את כל הקבועים החשובים בפרויקט, כל הפרמטרים של כל הפונקציות.  
הוא מחולק למקטעים על פי מקורות הקבועים בקבצים, כל הקבועים מהקובץ למשל בבלוק אחד של הגדרות וקוד.

הקובץ (צריך לשנות את השם של )   
אחראי על בניית המבוך בסביבת ה.   
הפונקציה המרכזית מייצרת ,בהינתן רשימה של מסילות פוליגונליות, את המבוך המתאים בסביבה של .

הקובץ :  
תפקיד הקובץ הוא לממש את הסביבה ולרכז את פעילות התכנית, קובץ ה קורא ל שמתחיל ומריץ את הסימולציה.  
בקובץ 2 פונקציות מרכזיות,  
פונקצית ה שמייצרת את הסביבה: פותחת את סביבת ה, מאתחלת את גרף ה, וקוראת לבניית כל האובייקטים מסוג ,   
פונקציית ה שאחראית על לקיחת צעד של כל מכונית, עדכון הגרף וסיום התוכנית במקרה הצורך: אם כל הרכבים הגיעו ליעדן או שאחת המכוניות התנגשה במכונית אחרת או בקיר.

הקובץ :

תפקיד הקובץ לממש את המחלקה ,  
המחלקה מכילה את כל הנתונים על המכונית: המיקום שלה, הצומת על גרף הPRM עליה היא נמצאת, המטרה שלה.  
היא אחראית על בניית המכונית בסביבת ה ועל התנועה שלו בסביבה- בדיקת התנגשויות, בחירת מסלול, סריקת מבוך בעזרת , וקבלת החלטות (כמו זכות קדימה לרכב אחר).  
בדומה למחלקה , מממשת 3 פונקציות מרכזיות:  
 שמאתחלת משתנים, ומוסיפה את הרכב לסביבת ה   
 שמבצעת תנועה אחת (תנועה של יחיד) של המכונית, בהתאם לאלגוריתם ה ההולונומי.  
 שבודקת את המצב החדש של המכונית: האם התנגשה בגוף אחר או האם הגיעה לנקודת המטרה.

הקובץ :

תפקיד הקובץ לטפל בבניית המפה – ייצוג המבוך ב"מוח" של המכונית, בעזרתו מחושב גרף הPRM ובהתאם לגרף - המסלול המינימלי למטרה.  
אובייקט מסוג מכיל 2 תכונות:

1. מפה: ייצוג של המבוך בעזרת רשימה של מסילות פוליגונליות.  
   המסילה הפוליגונלית המיוצגת על ידי רשימת הנקודות

שקולה לרשימת הקטעים:

בפועל, למכשולים יש עובי שמוגדר בתור בקובץ .  
לכן בפועל כל ישר מתורגם למלבן עם עובי ואורך .

1. גודל: מימדי הסביבה, ערך הx המקסימלי (אנו מניחים שכל סביבה היא ריבועית וממורכזת ב)

בעת ריצת התוכנה, כאשר מכונית תפגוש במכשול, עליה לעדכן את המסלול שלה בהתאם.  
על מנת לעשות זאת, היא תעדכן את המפה שלה, באמצעותה מחושב מחדש הגרף, ובאמצעותו מחושב מחדש המסלול.  
סריקות של המכשול החדש יכולות להיות מאוד רועשות, מרכבות מספר קטעים שונים, ומגלמות בתוכן טעויות מדידה וקירובים.  
על מנת למצות מהמידע הזה את המידע הרלוונטי, אנו משתמשים באלגוריתם (RDP).  
האלגוריתם מקבל רשימת נקודות והופך אותן למסילה פוליגונלית כך שהמלבנים המתאימים לה מכילים את כל נקודות הסריקה.  
  
הפונקציה העיקרית שבעזרתה מוסיפים נקודות למפה היא .  
היא מסננת נקודות לא רלוונטיות, שכבר מוכלות במלבן ששייך למפה  
מפרקת את רשימת הנקודות הנותרת לנקודות שנבעו ממכשולים נפרדים, בעזרת קבוע מרחק בין 2 נקודות סמוכות – אם 2 סריקות סמוכות (בזמן) רחוקות זו מזו , הקוד יפרק את הישרים ויניח שנבעו מ2 מכשולים שונים.  
כעת לכל קבוצת נקודות, מחושבת המסילה הפוליגונלית בעזרת .