**בעיית החנייה -** **parkingProblem**

**DoubleLinkedList– רשימה מקושרת דו כיוונית מעגלית**

**הסבר פתרון –**

נקרא לפונקציית עזר שתבדוק לנו מקרי קיצון:

* נבדוק האם הרשימה null או שהאיבר הראשון ברשימה הוא null , במידה וכן נחזיר כמות רכבים של 0
* נבדוק האם יש רק איבר אחד, אם כן נחזיר כמות רכבים 1
* אם שניהם לא קורים נחזיר 1-
* לאחר מכן נבדוק מה חזר לנו לתוצאה במידה שחזר לנו 1- אנחנו נרצה לרוץ ולבדוק כמה רכבים יש לנו.
* נגדיר 2 משתנים (m, n) שניהם יתחיל מראש הרשימה, ונאפס את המשתנה(result) שחזר מהפונקציה שבדקה לנו את הרשימה ל- 0.
* נרוץ בלולאת while כל עוד m שווה ל-n ובכל ריצה נעלה את result ב-1 ונרוץ בעוד לולאה עד לresult פעמים ונקדם את m result צעדים קדימה, לאחר מכן נרוץ בעוד לולאה שרצה result פעמים ובכל ריצה נחזיר את m result פעמים אחורה.
* נסיים את הריצה כאשר m שונה מ-n (או שm יהיה שווה לnull או אם נכנסנו למעגל אז הוא יהיה שווה לערך אחר מההתחלה).
* נבדוק למה הפסקנו לרוץ: אם m שווה לnull אז זה אומר שאנחנו לא ברשימה מעגלית ואז פשוט נחזיר את המשתנה result, במידה והוא שונה מnull זה אומר שאנחנו בתוך מעגל ואז נצטרך לבדוק עוד כמה דברים.
* נרוץ שוב כל עוד הם שונים ונקדם את n עד שהוא יגיע לנקודת מפגש עם m, ולאחר שהגענו למצב שהם שווים כלומר, נמצאים באותו קודקוד, נקדם את n לקודקוד הבא ונוסיף ל-result 1.
* לאחר מכן נרוץ בעוד לולאה כל עוד הם שונים ונקדם את n לאיבר הבא ונוסיף לספירה 1, בסוף הריצה נגיע למצב שm שווה ל-n ואז סיימנו לרוץ ונחזיר את result

**הוכחת סיבוכיות-**

בלולאה הראשונה אנחנו רצים מקסימום כגודל הרשימה ובתוך כל ריצה אנחנו רצים מקסימום כגודל הרשימה (להלוך ולחזור), אם הגענו לסוף הרשימה כלומר, הקודקוד שאנחנו מקדמים ומחזירים אחורה (m) יהיה שווה לnull אז אנחנו לא ברשימה מעגלית ואז נחזיר את הספירה לכן הסיבוכיות של זה היא  *כאשר n הוא גודל הרשימה.*

אם הגענו למצב שזה מעגלי אז נרוץ בדיוק כמו שאמרנו למעלה, ולאחר מכן נרוץ עוד x פעמים (עד לנקודת המפגש שלהם), לאחר מכן נקדם את m לאיבר הבא ונוסיף ל-result 1 לאחר מכן נרוץ עוד y פעמים עד לנקודת המפגש הבאה שלהם, ולכן הסיבוכיות היא:

כאשר