תרגיל בית 1:

שימוש באלגוריתמי חיפוש היוריסטיים לתכנון מסלולי חלוקה אופטימליים

מגישים

טל רוזנצוויג 307965806

שני אופיר 204512396

חלק א':

שאלה 1:

תשובה:

- קיימות !k אפשרויות לסידור k הדירות, כלומר סדר המעבר בהן הוא כמו סידורן בשורה.
- במעבר בין דירות אלו, אנו יכולים לבחור האם לבקר במעבדה או לא לבקר במעבדה ולכן עבור m מעבדות שניתן לעבור בהן בין כל שתי דירות נוסיף את האפשרות שנבחר לא לעבור כלל במעבדה. בנוסף, קיימים לנו בין הדירות k-1 מקומות ומקום אחד לפני הדירה הראשונה, ולכן נקבל כי קיימים k מקומות. מכאן, בהתבסס על ההנחה כי ניתן לבקר בכל מעבדה יותר מפעם אחת, נקבל $(m+1)^k$ אפשרויות.
- כעת, על מנת להבטיח שהמסלול יסתיים במעבדה לאחר הביקור בדירה האחרונה, נבחר מתוך m המעבדות במי נבקר ולכן m.

נשתמש בעקרון הכפל ונקבל:

$$k! \cdot (m+1)^k \cdot m$$

שאלה 2:

תשובה:

הסבר עד 3 שורות כנדרש:

תחילה, $k!$ הוא מספר כל האפשרויות ללכת לדירות בכל סדר. כעת, עבור מעבר בין שתי דירות נוכל כל פעם לבקר תחילה בכל דירה ואחרי זה בעוד דירות מיד אחרי בתנאי שלא ביקרנו בהן. כמו כן, בכל פעם נבחר את המעבדות בהן אנו רוצים לבקר ונבדוק האם לאחר ביקור בדירה נבקר שוב במעבדה כלשהי או נבחר שלא לבקר.

הסבר מפורט:

- נבחר את המעבדות בהן נבקר לפני הביקור בדירה הראשונה, נשתמש בעקרון הכפל ונכפול במספר הסידורים האפשריים של מעבדות אלו, סה"כ $\binom{m}{i_1} \cdot i_1!$ אפשרויות.
- נבחר דירה לבקר בה, קיימות k דירות ולכן k אפשרויות.
- במצב זה אנו נמצאים לאחר שביצענו ביקור בדירה כלשהי, כעת נבחר מהמעבדות שביקרנו בהן, מעבדה אחת שבה נוכל לבקר שוב או שנבחר לא לבקר כלל במעבדה ולכך יש $(l_1 + 1)$ אפשרויות.
- תוך שימוש בעקרון הכפל, נמשיך לבצע פעולות אלו עבור כל מרווח בין שתי דירות עוקבות, כלומר $(k-1)$ ונבחר מתוך המעבדות בהן לא ביקרנו את המעבדות שנבקר בהן ונכפול במספר הסידור שלהן. אנו בוחרים בכל פעם מתוך המעבדות שלא ביקרנו בהן מאחר ובשלב זה אנו לא נמצאים במצב של "לאחר" ביקור בדירה. נשים לב כי בביטוי האחרון, כלומר בסיגמא האחרונה, ישנו כפל בערך אחד והוא מבטא את הדירה האחרונה שבה אנו עוברים. כמו כן, מאחר ואנו צריכים לעבור במעבדה לאחר הביקור בדירה האחרונה, נבחר מתוך m המעבדות הקיימות במי לבקר.

לצורך נוחות נסמן $t = \sum_{j=0}^{k-1} i_j$ ו-i בכל סיגמא מבטא את מספר המעבדות הנבחרות מתוך m הקיימות בקלט.

$$\sum_{i_1=0}^{m} \binom{m}{i_1} \cdot i_1! \cdot k \cdot (1+i_1) \cdot \sum_{i_2=0}^{m-i_1} \binom{m-i_1}{i_2} \cdot i_2! \cdot (k-1)(1+i_1+i_2) \cdot \ldots$$

$$\cdot \sum_{i_k=0}^{m-t} \left(\binom{m-\sum_{j=0}^{k-1} i_j}{i_k} \right) \cdot l_k! \cdot 1 \cdot m$$

$$= m \cdot k!$$

$$\cdot \left[\sum_{i_1=0}^{m} \frac{m!}{(m-i_1)!} \cdot (i_1+1) \cdot \sum_{i_2=0}^{m-i_1} \frac{(m-i_1)!}{(m-i_1-i_2)!} \cdot i_2! \cdot (1+i_1+i_2) \cdot \ldots \right.$$

$$\left. \cdot \sum_{i_1=0}^{m-t} \frac{(m-t)!}{(m-t-i_k)!} \right]$$

שאלה 3:

k	m	$\#possiblePaths$	Estimated calculation time
7	2	~ 2.204496e+07	~18.47 [secs]
7	3	~ 2.477261e+08	~3.84 [mins]
8	3	~ 7.927235e+09	~2.25 [hours]
8	4	~ 6.300000e+10	~19.55 [hours]
9	3	~ 2.853804e+11	~3.69 [days]
10	3	~ 1.141522e+13	~5.33 [months]
11	3	~ 5.022696e+14	~21.05 [years]
12	3	~ 2.410894e+16	~1.08 [thousand years]
12	4	~ 4.677750e+17	~22.4 [thousand years]
13	4	~ 3.040538e+19	~1.54 [million years]

- הערה: כאשר נכתב ביטוי מהצורה "e+x" הכוונה היא ל- 10^x(כתיב מדעי).

חלק ג':

שאלה 4: יבש (1 נק'): מהם ערכי הקיצון (המקסימלי והמינימלי) האפשריים של דרגת היציאה במרחב החיפוש? ספקו ביטוי מתמטי כפונק' של הפרמטרים k, m של השאלה בלבד. נמקו בקצרה (שורה אחת לכל מקרה).

תשובה:

מקסימלי – במצב ההתחלתי, ישנם k+m מעבדות שעוד לא ביקרנו בהן ולכן זה המספר המקסימלי.

מינימלי - כאשר עברנו על כל הדירות והמעבדות, נשארו אפס מקומות לעבור בהם ולכן זה מינימלי.

שאלה 5: יבש (1 נק'): האם ייתכנו מעגלים במרחב החיפוש שלנו? אם כן תנו דוגמה למעגל כזה, אחרת נמקו. (עד 5 שורות).

תשובה: לא ייתכנו מעגלים במרחב החיפוש מכיוון שלא ייתכנו מעגלים במרחב המצבים. כעת, נראה כי לא ייתכנו מעגלים במרחב המצבים: נניח בשלילה כי קיים מעגל במרחב המצבים, אזי אם במעגל קיימות דירות אז ביקרנו באותה דירה פעמיים בסתירה לאופרטור ביקור בדירה. לכן במעגל **אין דירות** והוא מכיל מעבדות בלבד, אך זה מוביל לכך שביקרנו במעבדה ולאחר מכן ביקרנו בה שוב למרות שאין בה מטושים אז הסיבה היחידה לביקור בה היא שיש לנו בדיקות במקרר - אך אין לנו בדיקות במקרר מכיוון **שלא ביקרנו בדירות בין המעבדות במעגל** - בסתירה לאופרטור ביקור מעבדה.

שאלה 6: יבש (1 נק'): כמה מצבים יש במרחב זה (כפי שהוגדר)? האם כולם ישיגים (ציינו כן/לא)? נמקו (עד 3 שורות).

תשובה: במרחב החיפוש ישנם אינסוף מצבים ולא כולם ישיגים, מכיוון שישנם אינסוף מצבים במרחב המצבים ולא כולם ישיגים. כעת, נסביר מדוע ישנם אינסוף מצבים במרחב המצבים: מספר המטושים $Matoshim \in \mathbb{N}$ ולכן יכול להיות כל מספר טבעי, מכאן שקיימים אינסוף מצבים. נסביר מדוע לא כל המצבים ישיגים: מספר המטושים ההתחלתי + סכום המטושים במעבדות הוא **מספר סופי**, ולכן קיימים אינסוף מצבים עם מספר מטושים גדול יותר שעבורם ניתן להגדיר **מצב חוקי** אך המצב אינו ישיג כי זה יהיה מספר מטושים שהאמבולנס לא יכול לאסוף.

שאלה 7: יבש (1 נק'): האם ייתכנו בורות ישיגים מהמצב ההתחלתי שאינם מצבי מטרה במרחב החיפוש? אם כן – איך זה ייתכן? אם לא – למה? (נימוק לכל היותר שורה אחת)

תשובה: ייתכנו בורות ישיגים מהמצב ההתחלתי שאינם מצבי מטרה במידה ומתקיים שאין מספיק מטושים:

$$\sum_{i=1}^{m} l_i.matoshim + InitialNrMathoshimAmb < \sum_{i=1}^{k} d_i roomates$$

שאלה 8: יבש (1 נק'): מהו טווח האורכים האפשריים של מסלולים במרחב ממצב התחלתי אל מצב סופי? (אורך מסלול = מס' הקשתות) (לכל היותר 7 שורות סה"כ).

תשובה:

מינימלי – המקרה שבו יש מספיק מטושים באמבולנס ומספיק מקום לאחסן את כולם, ולכן מעבר על כל הדירות והגעה למעבדה – מסלול באורך $k+1$.

מקסימלי – במקרה בו אנו נעבור תחילה בכל m המעבדות בכדי לאסוף מטושים, לאחר מכן נעבור ב-k הדירות ונאסוף את הבדיקות כך שבין מעבר בין דירה לדירה נעבור במעבדה על מנת לפרוק את הבדיקות – מסלול באורך $2k+m$.

שאלה 9: יבש (1 נק'): הגדירו פורמלית ובצורה ישירה את פונקציית העוקב $Succ_{MDA}: S \to \mathcal{P}(S)$ המתאימה לבעיה זו (ללא שימוש בקבוצת האופרטורים O).

שימו לב, אנו מצפים לביטוי מהצורה: $Succ_{MDA}(s) = \{(?,?,\ldots)|?\} \cup \{(?,?,\ldots)|?\}$

תשובה:

$$\begin{aligned} Succ_{MDA}(s) = \{(&d_i.loc, s.Taken \cup \{d_i\}, s.Transferred, s.Matoshim \\ &- d_i.rommates, s.VisitedLabs) \mid di \\ &\notin s.Taken \cup s.Transferred \ \wedge \ di.roomates \\ &\leq s.Matoshim \ \wedge \ di.roomates \\ &\leq AmbulanceTestsCapacity \quad - \sum\nolimits_{d \in s.Taken} d.roommates \end{aligned}$$

$$\cup$$

$$\begin{aligned} \{(l_i.loc, s.Transferred \cup s.Taken, s.Matoshim + l_i Matoshim, s.visitedLabs \cup \{l_i\} | l_i \\ \notin s.VisitedLabs \vee s.Taken \neq \emptyset \vee l_i.Matoshim > 0\} \end{aligned}$$

חלק ה:

שאלה 14:

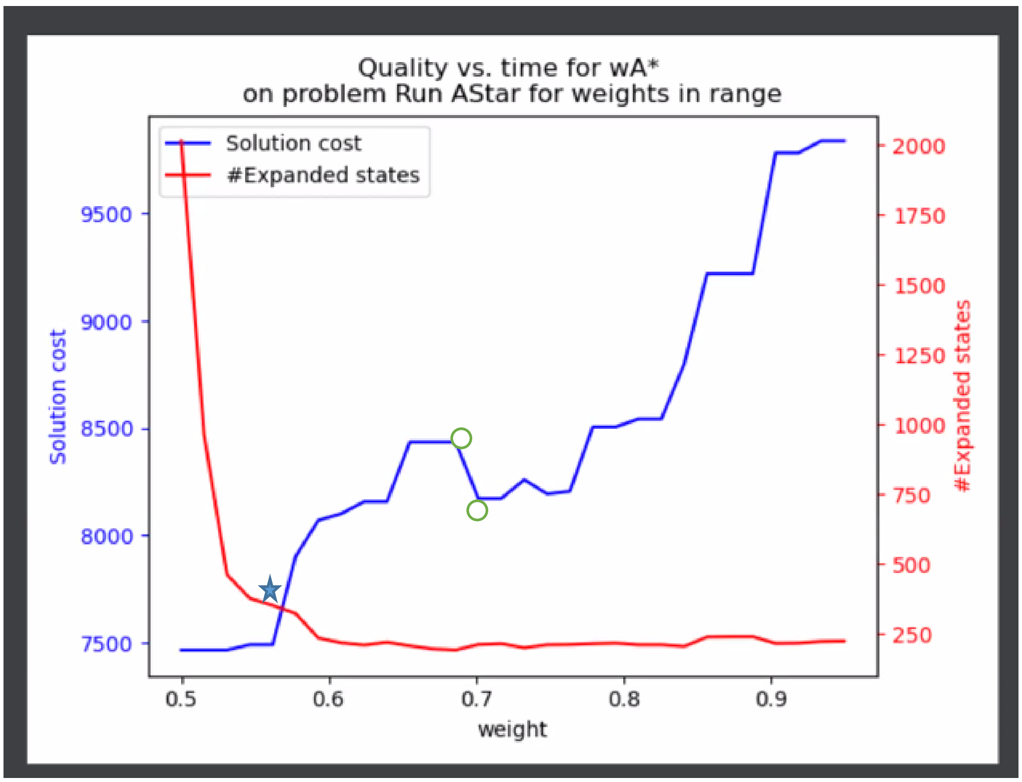
תשובה: בסעיף 12 היו לנו 17,354 פיתוחים וכעת לאחר שימוש ביוריסטיקה בסעיף 13 יש לנו 2015 פיתוחים כלומר חסכנו $\frac{17,354-2015}{17,354} = 0.88$, כלומר ההפרש היחסי הוא 0.88.

```
StreetsMap(src: 54 dst: 549)        A* (h=0, w=0.500)           time:   2.38   #dev: 17354    |space|: 17514    total_g_cost:  7465.52560
StreetsMap(src: 54 dst: 549)        A* (h=AirDist, w=0.500)     time:   0.28   #dev: 2015     |space|: 2229     total_g_cost:  7465.52560
```

שאלה 16:

תשובה: בציור מטה ניתן לראות שככל שנותנים משקל גבוה יותר ליוריסטיקה, מספר המצבים שנפתח יהיה נמוך יותר וכך גם זמן הריצה. אולם, איכות הפתרון שהיא האורך בין מצב ההתחלה למצב סופי, יהיה גרוע יותר.

כמו כן, ניתן לראות כי המגמה הכללית של הגרף מזכירה את כלל האצבע הנלמד בכיתה, אך נשים לב כי קיימות נקודות מסוימות המתנהגות אחרת. לדוגמא, עבור הקטע בין המשקלים 0.7 ו-0.67, מסומן ב-○ בגרף, ניתן לראות שמחיר הפתרון יורד אפילו כאשר המשקל עולה. היינו בוחרים בערך המשקל $w = 0.57$, כלומר נקודת החיתוך בגרף המסומנת ב-★, או בכל משקל הנמצא בתחום $[0.55,0.6]$, מאחר ובתחום הזה הפתרון מספיק טוב הן מבחינת המחיר שלו והן מבחינת מספר הפיתוחים אשר נמוך מאד יחסית לערכי המשקלים האחרים.



חלק ו:

שאלה 19:

החיסרון בגישה זו הוא שגודל מרחב המצבים יגדל משמעותית מאחר ולכל צומת ברשת הכבישים, נדרש לפתח את כל המצבים האפשריים, כלומר נקבל כי מספר הקשתות היוצאות מכל צומת הוא גדול מאד וכך נקבל בעיית זיכרון. בנוסף, כאשר לא מפרקים את הבעיה לגורמים, צריכת הזיכרון גדלה באופן ניכר וגם מתבצעות בדיקות מיותרות על חלק מהצמתים שכן אלו צמתים שידועים מראש שלא ניתן להשתמש בהם.

שאלה 20:

תשובה:

(i) מעל הגדרת המחלקה MDAState מופיעה שורת הקוד הבאה(השורה המבוקשת היא העליונה):

```
@dataclass(frozen=True)
class MDAState(GraphProblemState):
```

כלומר, המחלקה מוגדרת כ-frozen וכל ניסיון לבצע השמה לשדות של המחלקה יזרוק חריגה.

(ii) שורה זו אינה מספיקה מאחר ובפייתון עובדים עם מצביעים לאובייקטים עצמם וחלק מהשדות במחלקה הם מצביעים לאובייקטים שאנו לא רוצים לשנות. כך למשל, בפונקציה expand_state_with_costs אנו לוקחים מצב קיים ויוצרים ממנו מצבים אחרים, והרי כי חשוב לנו לא לשנות את אחד הערכים של ה- sets ב-state הנוכחי כאשר אנו רוצים לפתח את השכנים שלו. לכן, נוסיף את הערכים שבתוך המחלקה ל- frozenSet. נניח ולא היינו מגדירים אותם כ-frozenSet, היינו עלולים לשנות ערכים וכך ליצור טעויות שעלולות לפגוע בתקינות הקוד או בציפייה שלנו לערכים מסוימים.

(iii) כן, אלגוריתם *A עשוי לפגוש מצב בפעם השנייה וזה יבוא לידי ביטוי בשורת הפסאודו קוד:

$CLOSED \leftarrow CLOSED \setminus \{\text{old\_node}\}$

$OPEN \leftarrow OPEN \cup \{\text{old\_node}\}$; Move old node from CLOSED to OPEN

כאשר אנו מוציאים את הצומת מתוך ה-CLOSE ומחזירים אותו לתור ה-OPEN מאחר ומצאנו מסלול טוב יותר עבורו.

(iv) דוגמא:

חשוב לנו להקפיא את MDAState כך שכאשר אנחנו מפתחים את המצב העוקב, לא נזין "בטעות" ערכים שגויים.

ביצוע ה-frozenset על המחלקה מגן על השדות הפנימיים, אך חלק מהשדות הם מצביעים לאובייקטים אחרים כאשר גם אותם אנו לא רוצים לשנות, כדוגמת state_to_expand.test_on_ambulance כפי שניתן לראות:

```
tests_on_ambulance: FrozenSet[ApartmentWithSymptomsReport]
```

כעת, אם בפונקציה expand_state_with_costs נבצע את השורות הבאות:

```python
new_test_trasfered = state_to_expand.tests_transferred_to_lab
for test in state_to_expand.tests_on_ambulance:
    new_test_trasfered.add(test)
```

בהשמה של state_to_expand.tests_transferred_to_lab לתוך new_test_trasfered מתבצעת השמה של copy by reference ומשתמש לא משופשף בשפה יכול ליפול בשגיאה ובעצם לשנות את אובייקט המקור לדוגמא ע"י הוספת איברים כמו בדוגמא למעלה. אם אובייקט המקור מוגן באמצעות frozenset, אנחנו מגנים על האובייקט מטעויות אלו.

שאלה 23:

הוכח/הפרך: ההיוריסטיקה MDAMaxAirDistHeuristic הינה קבילה (עבור פונק' המחיר $cost_{MDA}^{dist}$). ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

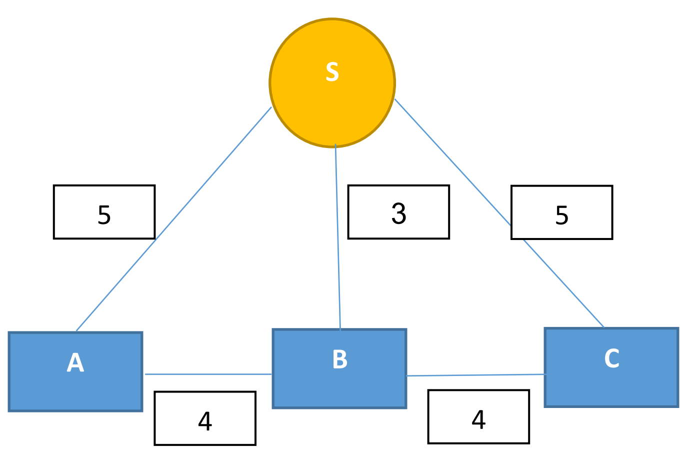
תשובה -הוכחה:

תהיי פונקציית מחיר h*(s) כך שלכל מצב היא מגדירה את המחיר הזול ביותר מהדירה למטרה בהגדרה. בנוסף, $h(s) \geq 0$ מאחר והמרחק האווירי בין 2 נקודות נתונות הוא אי שלילי לפי MDAMaxAirDistHeuristic.
נניח בשלילה כי קיים מצב s כך ש- h(s)>h*(s) וכי האמבולנס נמצא כעת במצב s. אזי, קיימות 2 דירות a, b כך שהמרחק האווירי ביניהן הוא h(s) והאמבולנס עדיין לא ביקר בהן. מהגדרת המסלול האופטימלי, מסלול זה חייב לעבור בכל הדירות וכך גם ב-a ו-b. אם כן, נניח בה"כ כי האמבולנס עובר תחילה בדירה a וכי על מנת לעבור לדירה b, האמבולנס יצטרך לעבור מרחק אווירי שהוא לפחות h(s). אולם, נשים לב כי h*(s)<h(s) וכי לא ייתכן שמרחק מסלול כלשהו יהיה קטן מהמרחק האווירי. סתירה.

שאלה 26:

הוכח/הפרך: ההיוריסטיקה MDASumAirDistHeuristic הינה קבילה (עבור פונק' המחיר $cost_{MDA}^{dist}$). ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

תשובה: הפרכה - נשתמש בדוגמא נגדית:



הסבר לגרף המתואר:

- S – המעבדה היחידה והמיקום ההתחלתי
- A,B,C – הדירות שהאמבולנס צריך לעבור בהן

נניח כי בכל דירה יש דייר אחד וכי האמבולנס אסף מספיק מטושים במעבדה כאשר יש מספיק מקום לכולם באמבולנס. לפי היוריסטיקה MDASumAirDistHeuristic האמבולנס ייבחר במסלול האוויר הבא:

$$\mathrm{P} = S \to B \to A \to C$$

לכן, נקבל כי משקלו הוא: $h(s) = 3 + 4 + 8 = 15$

נשים לב כי קיים מסלול קצר יותר והוא: $P_{new} = S \to A \to B \to C$

עבור מסלול זה נקבל כי המשקל הוא: $W(P_{new}) = 5 + 4 + 4 = 13$

מכאן נקבל: $h^*(s) \leq 13 < 15 = h(s)$, כלומר קיבלנו כי היורסטיקה אינה קבילה.

הערה: במקרה שבו היוריסטיקה מחושבת עד לצומת סיום נקבל תוצאה זהה, וזאת מאחר שבשני המסלולים P ו-P' יש לנסוע מצומת C למעבדה ולכן נוסיף מרחק של 5 לכל אחד מהמסלולים והיחס עדיין יישאר כפי שהצגנו.

שאלה 29:

הוכח/הפרך: ההיוריסטיקה MDAMSTAirDistHeuristic הינה קבילה (עבור פונק' המחיר $cost_{MDA}^{dist}$). ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

תשובה:

הוכחה:

סימונים:

h*(s) - המחיר של המסלול המינימלי מ-s לצומת הסיום.
h(s) – הערך היוריסטי של הצומת s עבור היוריסטיקה הנ"ל.
נשים לב כי, מתקיים כי $h(s) \geq 0$ מאחר הערכה של מחיר היא אי שלילית לפי MDAMSTAirDistHeuristic.

נניח בשלילה כי קיים צומת v כך ש- $h(v) > h^*(v)$. יהי P המסלול מ- v לצומת הסיום שמשקלו $h^*(v)$ כך ש- $\mathrm{P} = v \to v_0 \to v_1 \to \cdots \to v_n$ כאשר v_n הוא צומת מטרה. לפי הגדרת מסלול, P בהכרח עובר בכל הדירות שבהן אנו צריכים לבקר. יהי עפ"מ T על המסלול P. מאחר ומסלול זה עובר דרך כל הדירות, T הוא עפ"מ גם עבור הבעיה המקורית ולכן מתקיים $W(T) = h(v)$. אולם, $h(v)$ הוא משקל העפ"מ בגרף של כל הדירות שעוד לא עברנו בהן. בנוסף, מאחר שכל הקשתות במסלול P אי שליליות, מתקיים $W(T) \leq W(P)$ ולכן מתקיים: $h(v) \leq W(P) = h^*(v)$, וזו סתירה לכך ש-$h(v) > h^*(v)$ ולכן בהכרח מתקיים $h^*(s) \leq h(v)$ לכל מצב s.

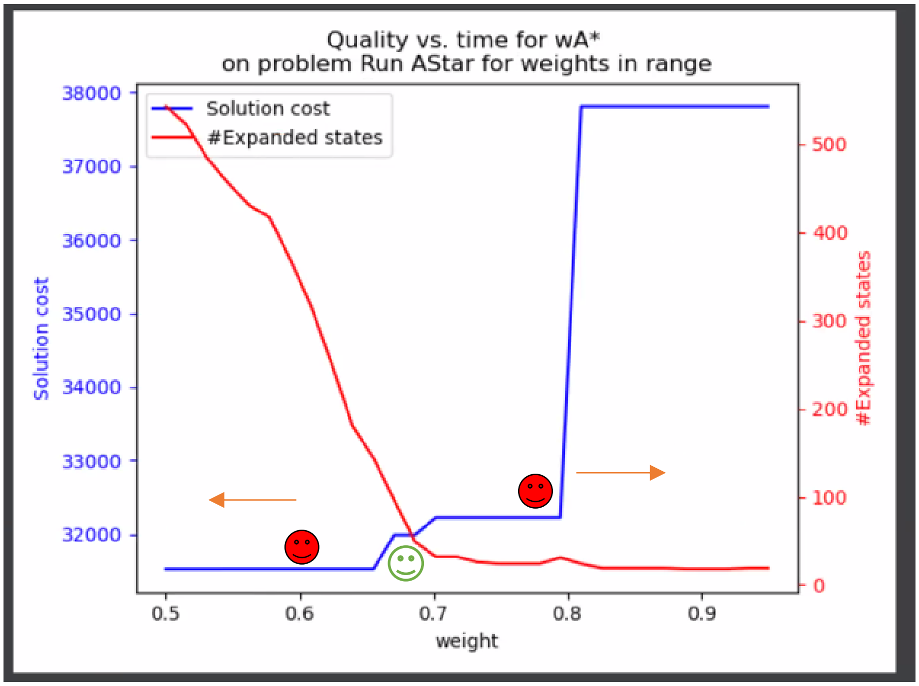
שאלה 30:

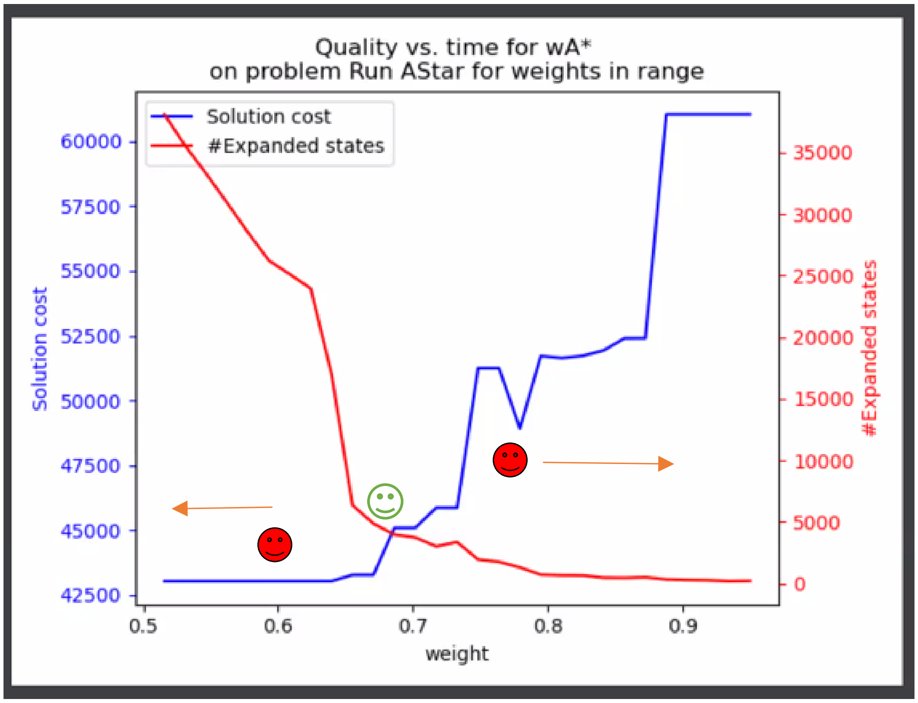
תשובה:

כפי שלמדנו, ככל שהמשקל שאנו נותנים ליוריסטיקה גבוה יותר, כך מספר הפיתוחים גדל ואיכות הפתרון יורדת. בשני היוריסטיקות נקבל כי עבור ערכים הנמצאים קצת פחות מערך הנקודה $W = 0.7$ נקבל את הערך הכי טוב מבחינת מספר פיתוחים ועלות פתרון ביחד, לכן נבחר להשתמש בערך זה. נסמן בגרף ב- את הערך הכי טוב.

הערכים המסומנים החל מ- ובכיוון החצים, הם פחות כדאיים מכיוון:

- בתחום הימני ככל שנלך יותר ימינה, מחיר הפתרון עולה אך כמות הפיתוחים לא יורדת באופן משמעותי.
- בתחום השמאלי ככל שנלך יותר שמאל, המחיר יורד באופן לא משמעותי אך מספר הפיתוחים מאוד גבוה.





חלק ז':

שאלה 31:

	MDAMaxAirDistHeuristic	MDASumAirDistHeuristic	MDAMSTAirDistHeuristic
$cost_{MDA}^{test\ travel}$	לא	לא	לא
$cost_{MDA}^{monetary}$	לא	לא	לא

שאלה 32:

הרצה ראשונה – פלט

MDA(small_MDA(5):Distance) UniformCost time: 11.47 #dev: 1024 |space|: 1714 total_g_cost: 31528.65909 total_cost: MDACost(dist= 31528.659m , money= 49.717NIS, tests-travel= 52112.429m) |path|: 8

הרצה שנייה – פלט

Solve the MDA problem (monetary objectives).

MDA(small_MDA(5):Monetary) UniformCost time: 13.09 #dev: 2236 |space|: 2532 total_g_cost: 42.04962 total_cost: MDACost(dist= 31923.809m , money= 42.050NIS, tests-travel= 53317.118m) |path|: 8

הסבר:
בהרצה הראשונה עם אופטימיזציה לפי מרחק, קיבלנו MDACost עם הערכים:
- $distance = 31528.659m$
- $money = 49.717\text{NIS}$

בהרצה השנייה עם אופטימיזציה לפי עלות, קיבלנו MDACost עם הערכים:
- $distance = 31923.809\text{m}$
- $money = 42.050\text{NIS}$

כלומר, קיבלנו כי האופטימיזציה עובדת כפי שציפינו כך שהעלות קטנה על חשבון המרחק.

שאלה 34:

הוכח/הפרך: ההיוריסטיקה *MDATestsTravelTimeToNearestLabHeuristic* הינה קבילה עבור פונק' המחיר $cost_{MDA}^{test\ travel}$. ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

תשובה - הוכחה:
לכל צומת s נגדיר h*(s) כמחיר של מסלול אופטימלי המתחיל בנקודה s ומגיע למצב סיום(מעבדה כלשהי). נסמן ב- h(s) את הערך היוריסטי עבור הצומת s. נשים לב כי לכל s מתקיים: $h(s) \geq 0$ מאחר והמרחקים והבדיקות מכילים מספרים אי שליליים. נניח בשלילה שהיוריסטיקה איננה קבילה, לכן קיים צומת s עבורו $h(s) > h^*(s)$. יהי P המסלול האופ' עבור s ש- h*(s) מחשבת. נסמן את סך מספר הבדיקות שבוצעו ב-n ולכל בן אדם i שנבדק נסמן את הדרך שבוצעה הבדיקה שלו ב-way(i). אזי, לכל מסלול P מתקיים:

$$cost_{travel}^{test} = \sum_{i=1}^{n} way(i)$$

מתקיים $h(s) > h^*(s)$. מאחר ו-h(s) מחשבת מסלול ומספר הבדיקות זהה בשני המסלולים, אזי קיים אדם j כך ש- way(i) עבור היוריסטיקה > way(i) של המסלול האופטימלי P. אולם, היוריסטיקה מחשבת לכל דירה את המרחק הכי קצר ממנה למעבדה מסוימת (אליה היא נוסעת), ולכן לא ייתכן שבמסלול P הבדיקה עברה מסלול קצר יותר. סתירה.

שאלה 35:

הראו בדו"ח איך רואים בתוצאות שהפתרון המתקבל אכן ממזער את המדד הרלוונטי בהתאם לפונק' העלות שהופעלה (אין צורך לצרף את כל הפלט עם המסלול, רק את העלויות).

תשובה:
הרצה ראשונה - פלט:
MDA(moderate_MDA(8):Distance) A* (h=MDA-MST-AirDist, w=0.500) time: 137.48 #dev: 46054 |space|: 66167 total_g_cost: 43034.79407 total_cost: MDACost(dist= 43034.794m, money= 95.847NIS, tests-travel= 176505.013m) |path|: 13 path: [(loc: initial-location tests on ambulance: [] tests transferred t.

הרצה שנייה - פלט:

MDA(moderate_MDA(8):TestsTravelDistance) A* (h=MDA-TimeObjectiveSumOfMinAirDistFromLab, w=0.500) time: 66.74 #dev: 51388 |space|: 88474 total_g_cost: 131265.15303 total_cost: MDACost(dist= 93226.428m, money= 127.199NIS, tests-travel= 131265.153m) |path|: 18 path: [(loc: initial-location tests on ambulance[] :

הסבר:

בהרצה הראשונה עם אופטימיזציה לפי מרחק, קיבלנו MDACost עם הערכים:

- $distance = 43034.794m$
- $tests - travel = 176505.013m$

בהרצה השנייה עם אופטימיזציה לפי testTravel, קיבלנו MDACost עם הערכים:

- $distance = 93226.428m$
- $\mathrm{tests} - \mathrm{travel} = 131265.153\mathrm{m}$

כלומר, קיבלנו כי האופטימיזציה עובדת כפי שציפינו כך שה-test-travel קטנה על חשבון המרחק.

שאלה 36:

יבש (3 נק'): הוכח/הפרך: אם קיים פתרון במרחב המקורי $\mathcal{S}_{MDA}$, אלג' $\mathcal{A}_1$ בהכרח מחזיר פתרון. ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

תשובה:

הוכחה: יהי פתרון במרחב המקורי $\mathcal{S}_{MDA}$ שנסמנו ב- P כך ש- $P = s_0 \rightarrow s_1 \rightarrow \cdots \rightarrow s_t$ ונגדיר כי s_0 הוא מצב התחלתי ו-s_t הוא מצב סיום.

נניח בשלילה כי האלגוריתם A_1 לא מוצא פתרון.

יהי s_i הצומת האחרון מהמסלול *P* אשר האלגוריתם פיתח. נשים לב כי בהכרח קיים אחד כזה מאחר ולכל פתרון ישנו מצב התחלתי יחיד.

כעת, כאשר האלגוריתם A_1 מגיע לצומת s_i מתקיים כי הצומת s_{i+1} נכנס לתור ה-*OPEN*, אולם האלגוריתם לא המשיך אליו מאחר והוא נתקע בשלב אחר של המסלול. על כן, בכל שלב לאחר מכן, מתקיים כי s_{i+1} נמצא בתור ה-*OPEN* ובפרט בשלב בו האלגוריתם נתקע ולכן יתקיים כי בשלב מסוים הצומת s_{i+1} יגיע לסוף התור ובהכרח הוא ייבחר להיות חלק מהמסלול האופטימלי. קיבלנו סתירה לכך ש-s_i הוא הצומת האחרון שהאלגוריתם פיתח ולכן הנחת השלילה הייתה שגויה ונקבל כי הטענה נכונה.

שאלה 37:

יבש (3 נק'): הוכח/הפרך: אם אלג' $\mathcal{A}_1$ מחזיר פתרון אז הפתרון המוחזר בהכרח אופטימלי ע"פ **הקריטריון המשולב** שהוגדר מעלה. ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

תשובה:

<u>הוכחה:</u> נניח כי האלגוריתם $\mathcal{A}_1$ מחזיר פתרון.
תחילה, נסמן את המסלול המיוצג ע"י המצב הסופי המתקבל מהאלגוריתם $\mathcal{A}_1$ ב-K כך שמתקיים $K \in K_{MDA}^{I \to G}$,
וכך גם מהגדרת פונקציית העלות, מתקיים כי הפתרון K המוחזר ע"י $\mathcal{A}_1$ מקיים כי
$cost_{MDA}^{test\ travel}(K) \leq (1+\epsilon) \cdot C_{dist}^*$, ולכן $K \in DistEpsOptimalPaths$. בנוסף, אלגוריתם USC הינו קביל, כפי שנלמד בהרצאה, ולכן מובטח כי אלגוריתם זה יחזיר תמיד את הפתרון המינימלי. על כן, מאחר והערך אותו אנו ממזערים הוא $cost_{MDA}^{test\ travel}$, נקבל כי $cost_{MDA}^{test\ travel}(K) = \bar{C}^*$ ולכן $K \in OptimalPath$, כלומר הפתרון המוחזר על ידי $\mathcal{A}_1$ בהכרח אופטימלי ע"פ הקריטריון המשולב שהוגדר מעלה.

שאלה 38:

בשלב זה נממש ונריץ וריאציה של $\mathcal{A}_2$ (השינוי הוא שבמימוש נשתמש ב- A^* עם היוריסטיקה קבילה במקום ב- UCS).
השלימו בקובץ main.py את הקוד תחת ההערה הרלוונטית לסעיף זה. צרפו את התוצאות שקיבלתם לדו"ח (אין צורך במסלולים – מספיק עלויות הפתרון). השוו בטבלה לתוצאות הריצה מסעיפים קודמים (על אותה הבעיה עם שתי פונק' עלות השונות) ובדקו מספרית האם הפתרון המתקבל בסעיף זה אכן מקיים איזון בין שני המדדים. חשבו וצרפו לדו"ח את הערך $\frac{DistCost(ReturnedSolution)}{C_{dist}^*} - 1$. האם אכן נשמר ערך ה- ε הנקוב?

תשובה:

<u>הרצה של מימוש A_2 – פלט:</u>

MDA(moderate_MDA(8):TestsTravelDistance)   A (h=MDA-TimeObjectiveSumOfMinAirDistFromLab, w=0.500)   time: 40.08   #dev: 51032 |space|: 83783   total_g_cost: 132209.98140   total_cost: MDACost(dist= 65686.522m, money=   99.486NIS, tests-travel= 132209.981m) |path|: 15*

<u>פלט של הרצות קודמות:</u>

MDA(moderate_MDA(8):Distance)   A* (h=MDA-MST-AirDist, w=0.500)   time: 126.22   #dev: 46054 |space|: 66167   total_g_cost: 43034.79407   total_cost: MDACost(dist= 43034.794m, money=   95.847NIS, tests-travel= 176505.013m) |path|: 13

ניתן לראות כי הפתרון המתקבל בסעיף זה מקיים איזון בין שני המדדים.
בהרצה לפי מימוש של A_2 קיבלנו את הערכים:
- $distance = 65686.522m$
- $tests - travel = 132209.981m$

בהרצה הקודמת קיבלנו את הערכים:

- $distance = 43034.794\text{m}$
- $\text{tests} - \text{travel} = 176505.013\text{m}$

בנוסף, כאשר $\epsilon = 0.6$ מתקיים:

$$\frac{DistCost(ReturnedSolution)}{C^*_{dist}} - 1 = \frac{65686.522}{43034.794} - 1 \cong 0.53$$

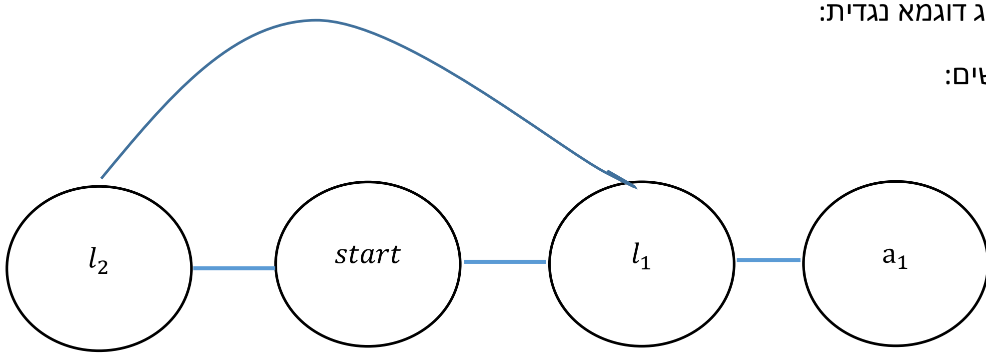
<u>שאלה 39:</u>

הוכח/הפרך: אם קיים פתרון במרחב, אלג' $\mathcal{A}_2$ בהכרח מחזיר פתרון.
טיפ: כדי לקבל קצת יותר אינטואיציה, אתם יכולים להריץ את הדוגמא מסעיף קודם עם ערכי ε שונים. ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך (הטיפ כאן ניתן רק ככלי עזר לפיתוח האינטואיציה. יש לספק הוכחה/הפרכה פורמלית ומלאה לפי ההוראות וללא התייחסות לתוצאות ריצה כזו או אחרת).

<u>תשובה:</u>

הפרכה, נציג דוגמא נגדית:

מפת הכבישים:



מרחב בעיית ה-MDA:
האמבולנס מתחיל מ-start, כאשר הוא ריק מבדיקות ומטושים ונניח כי fridgeCapacity=10.
בדירה a_1 קיימות 2 בדיקות לביצוע, ובכל מעבדה יש מטוש אחד בלבד.
נקבע כי $\epsilon = 0.1$ ומרחק כל דרך הוא 1.

1. נריץ אלגוריתם A^* עם יוריסטיקה קבילה כלשהי עם אופטימיזציה לפי פונקציית עלות $cost^{dist}_{MDA}$.
2. מכיוון שהיוריסטיקה קבילה, נקבל פתרון אופטימלי וערך המסלול האופטימלי מבחינת מרחק הוא $c^*_{dist} = 4$.
3. נריץ UCS על המרחב עם אופטימיזציה לפי $cost^{test-travel}_{MDA}$ ונסכום את $cost^{dist}_{MDA}$ בשדה נפרד כאשר עבור כל צומת חדש נבדוק האם הסכום שלו חרג מ- $(1+\epsilon) * c^*_{dist} = (1+0.1) * 4 = 4.4$ ובמידה והוא חרג **לא נכניס אותו ל-open**.

- מצב s' מייצג מצב דומה ל-s במיקום אך שונה מבחינת שאר הפרמטרים.

בטבלת המעקב ניתן לראות כי הגענו למצב מטרה אך תור ה-open ריק, ולמרות שקיים פתרון במרחב, האלגוריתם A_2 **לא מחזיר פיתרון**.

	Current state	open	close	path	comments
1.	Location: start Transferred = 0 Matoshim = 0 Taken = 0 Dist = 0 Test-travel = 0	$\{l_1, l_2\}$	$\{\}$	$start$	האלגוריתם יבחר ללכת לאחת המעבדות באופן שרירותי כי המחיר זהה, לצורך הדוגמא נבחר במעבדה 1.
2.	Location: l_1 Transferred = 0 Matoshim = 1 Taken = 0 Dist = 1 Test-travel = 0	$\{l_2, l_2'\}$	$\{start\}$	$start \to l_1$	לא מכניסים את דירה 1 ל- open מאחר ואין מספיק מטושים על האמבולנס
3.	Location: $l_2{}'$ Transferred = 0 Matoshim = 2 Taken = 0 Dist = 2 Test-travel = 0	$\{a_1, l_2\}$	$\{start, l_2\}$	$start \to l_1$ $\to l_2{}'$	
4.	Location: a_1 Transferred = 0 Matoshim = 0 Taken = 2 Dist = 4 Test-travel = 0	$\{l_2\}$	$\{start, l_1, l_2'\}$	$start \to l_1$ $\to l_2' \to a_1$	זה השלב שבמהלך הבא של האלגוריתם הצומת יחרוג מה-$(1+\epsilon) \cdot c^*_{dist}$ ולא נכניס אותו לOpen. האלגוריתם רוצה לחזור ל- l_1 אך הוא חורג מהמגבלה ומתעלמים מהפתרון הזה.
5.	Location: l_2 Transferred = 0 Matoshim = 1 Taken = 0 Dist = 1	$\{l_1'\}$	$\{start, l_1, l_2', a_1\}$	$start \to l_2$	

	Test-travel = 0				
6.	Location: $l_1{}'$ Transferred = 0 Matoshim = 2 Taken = 0 Dist = 2 Test-travel = 0	$\emptyset$	$\{start, l_1, l_2, l_2', a_1\}$	$start \to l_2$ $\to l_1{}'$	a_1 לא נכנס ל-open מכיוון שהמצב הזה נמצא ב-close ולא משפרים את המסלול אליו ולכן לא נוציא אותו ונעדכן את ערכיו.

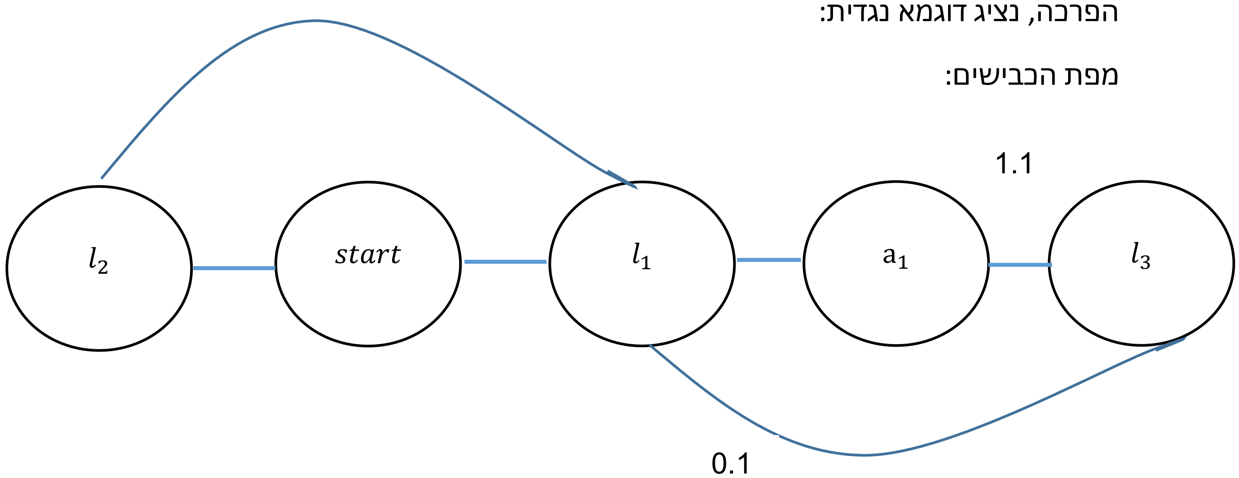
<u>שאלה 40:</u>

הוכח/הפרך: אם אלג' $\mathcal{A}_2$ מחזיר פתרון אז הפתרון המוחזר בהכרח אופטימלי ע"פ **הקריטריון המשולב** שהוגדר מעלה.
ראה בעמוד השני במסמך את ההערות המתייחסות לשאלות הוכח/הפרך.

<u>תשובה:</u>

הפרכה, נציג דוגמא נגדית:

מפת הכבישים:



מרחב בעיית ה-MDA:
האמבולנס מתחיל מ-start, כאשר הוא ריק מבדיקות ומטושים ונניח כי fridgeCapacity=10.
בדירה a_1 קיימות 2 בדיקות לביצוע, ובכל מעבדה יש מטוש אחד בלבד.
נקבע כי $\epsilon = 0.1$ ומרחק כל דרך הוא 1.

1. נריץ אלגוריתם A^* עם יוריסטיקה קבילה כלשהי עם אופטימיזציה לפי פונקציית עלות $cost_{MDA}^{dist}$.
2. מכיוון שהיוריסטיקה קבילה, נקבל פתרון אופטימלי וערך המסלול האופטימלי מבחינת מרחק הוא $c_{dist}^* = 4$.

3. נריץ UCS על המרחב עם אופטימיזציה לפי $cost_{MDA}^{test-travel}$ ונסכום את $cost_{MDA}^{dist}$ בשדה נפרד כאשר עבור כל צומת חדש נבדוק האם הסכום שלו חרג מ- $(1+\epsilon)*c_{dist}^{*} = (1+0.1)*4 = 4.4$ ובמידה והוא חרג **לא נכניס אותו ל-open**.

- מצב s' מייצג מצב דומה ל-s במיקום אך שונה מבחינת שאר הפרמטרים.

ראשית נראה כי קיים פתרון אופטימלי במרחב עפ"י הקריטריון המשולב, נביט במסלול

$$p: start \rightarrow l_2 \rightarrow l_1 \rightarrow a_1 \rightarrow l_1$$

$$dist(p) = 4 \wedge test-travel(p) = 2$$

כעת נראה כי האלגוריתם A_2 מחזיר פתרון **לא אופטימלי** על פי הקריטריון המשולב:

	Current state	open	close	path	comments
1.	Location: start Transferred = 0 Matoshim = 0 Taken = 0 Dist = 0 Test-travel = 0	$\{l_1, l_2\}$	$\{\}$	$start$	האלגוריתם יבחר ללכת לאחת המעבדות באופן שרירותי כי המחיר זהה, לצורך הדוגמא נבחר במעבדה 1.
2.	Location: l_1 Transferred = 0 Matoshim = 1 Taken = 0 Dist = 1 Test-travel = 0	$\{l_2, l_2', l_3\}$	$\{start\}$	$start \rightarrow l_1$	שוב לא נפתח את הדירה כי אין מספיק מטושים.
3.	Location: l_2' Transferred = 0 Matoshim = 2 Taken = 0 Dist = 2 Test-travel = 0	$\{l_2, l_3\}$	$\{start, l_1\}$	$start \rightarrow l_1 \rightarrow l_2'$	האלגוריתם יכול לבחור בין המעבדות 2 ו-3 באותו מחיר ולכן נבחר במעבדה 2.
4.	Location: a_1 Transferred = 0 Matoshim = 0 Taken = 2 Dist = 4 Test-travel = 0	$\{l_2, l_3\}$	$\{start, l_1, l_2'\}$	$start \rightarrow l_1 \rightarrow l_2' \rightarrow a_1$	נשים לב כי במצב זה l_3' אינו נכנס ל-*open* וגם לא l_1'' ולכן לא ניתן לעבור אליהם ולסיים.

5.	Location: l_2 Transferred = 0 Matoshim = 1 Taken = 0 Dist = 1 Test-travel = 0	$\{l_1' l_3\}$	$\{start, l_1, l_2', a_1\}$	$start \to l_2$	
6.	Location: $l_1{}'$ Transferred = 0 Matoshim = 2 Taken = 0 Dist = 2 Test-travel = 0	$\{a_1', l_3\}$	$\{start, l_1, l_2, l_2', a_1\}$	$start \to l_2$ $\to l_1{}'$	
7.	Location: l_3 Transferred = 0 Matoshim = 3 Taken = 0 Dist = 2.1 Test-travel = 0	$\{a_1'\}$	$\{start, l_1, l_2, l_2', l_1', a_1\}$	$start \to l_2$ $\to l_1' \to l_3$	כעת יכול האלגוריתם לבחור בין מעבדה 1 ל3 באותה עלות נבחר ב 3.
8.	Location: a_1 Transferred = 0 Matoshim = 1 Taken = 2 Dist = 3.2 Test-travel = 0	$\{l_3{}'\}$	$\{start, l_1, l_1', l_2', a_1, l_3\}$	$start \to l_2$ $\to l_1' \to l_3$ $\to a_1$	מעבדה 1 לא בopena מביוון שחרגה קודם מהמרחק המותר.
9.	Location: $l_3{}'$ Transferred = 2 Matoshim = 1 Taken = 0 Dist = 4.3 Test-travel = 2.2	$\{\}$	$\{start, l_1, l_1', l_2', a_1, l_3\}$	$start \to l_2$ $\to l_1' \to l_3$ $\to a_1 \to l_3{}'$	הגענו לפתרון.

נשים לב כי הפתרון שהתקבל $p: start \to l_2 \to l_1 \to l_3 \to a_1 \to l_3$ מקיים:

$$dist(p) = 4.3 \leq (1+\epsilon) * c_{dist}^{*} = 4.4$$

ומתקיים כי: $cost_{MDA}^{test-travel}(p) = 2.2$ ולכן **אינו** הפתרון האופטימלי כפי שראינו בתחילת התרגיל שקיים פתרון אופטימלי המקיים $cost_{MDA}^{test-travel}(p^*) = 2$.

שאלה 41:

ציין והסבר בקצרה יתרון צפוי של $\mathcal{A}_2$ ע"פ $\mathcal{A}_1$ במובנים של זמני ריצה. התייחס בתשובתך ליחסי הגדלים בין שני המרחבים (עליהם שני האלג' רצים). תשובה עד 3 שורות.

תשובה:

היתרון הצפוי לריצת $\mathcal{A}_2$ על $\mathcal{A}_1$ במובן זמני ריצה הוא השימוש בהיוריסטיקה. אלגוריתם $\mathcal{A}_1$ עושה שימוש ב-USC, אשר מסוגל להגיע לזמנים גבוהים ואילו אלגוריתם $\mathcal{A}_2$ עושה שימוש בהיוריסטיקה קבילה, מה שיוביל לפיתוח של פחות צמתים ובכך מתבטא היתרון שלו על פני $\mathcal{A}_1$.

חלק ח':

שאלה 44:

צרפו לדו"ח את התוצאות שקיבלתם בסעיף הקודם (אל תצרפו את המסלולים עצמם). האם חסכנו בפיתוחים? אם כן, בכמה? הסבירו למה בכלל ציפינו מראש ש- $A^*\varepsilon$ יוכל לחסוך במס' הפיתוחים בתצורה שבה הרצנו אותו. לא מספיק לטעון ש- $A^*\varepsilon$ גמיש יותר בבחירה של הצומת הבא לפיתוח. נסו להסביר למה בעצם אנחנו מצפים שהגמישות הזאת של $A^*\varepsilon$ אכן תעזור לנו במקרה הזה לבחור מ- open צומת לפיתוח שיקדם אותנו מהר יותר למטרה. מה בעצם הוספנו לאלג' החיפוש? תשובה עד 2 שורות.

תשובה:

הרצה ראשונה – פלט:

MDA(small_MDA(5):Distance) A* (h=MDA-MST-AirDist, w=0.500) time: 0.74 #dev: 543 |space|: 877 total_g_cost: 31528.65909 total_cost: MDACost(dist= 31528.659m, money= 49.717NIS, tests-travel= 52112.429m) |path|: 8 path: [(loc: initial-location tests on ambulance: [] tests transferred to lab: []

הרצה שנייה – פלט:

MDA(small_MDA(5):Distance) A*eps (h=MDA-MST-AirDist, w=0.500) time: 2.35 #dev: 492 |space|: 821 total_g_cost: 31528.65909 total_cost: MDACost(dist= 31528.659m, money= 49.717NIS, tests-travel= 52112.429m) |path|: 8 path: [(loc: initial-location tests on ambulance: [] tests transferred to lab: []

הסבר:

לפי תוצאות שתי ההרצות, ניתן לראות כי חסכנו 51 פיתוחים. אלגוריתם A*eps משתמש ב-focal ולכן הוא מיודע ועובר עוד שכבת סינון בזמן פיתוח הצומת. כעת, ב-A* ניקח את הצומת בעל ה-f המינימלי אשר נמצא ב-OPEN וב-A*eps מבין קבוצת הצמתים המינימליים עד כדי $1+\epsilon$ נבחר את הצומת בעל הערך היוריסטי הנמוך ביותר. כלומר, נקבל כי כאשר ההבדלים ב-f הם יחסית זניחים, ניתן יותר ערך ליוריסטיקה וכאשר ההבדלים ב-f הם לא זניחים, ניתן יותר ערך ל-f. בנוסף, נציין כי היוריסטיקה ב-*focal* מיודעת יותר מאשר היוריסטיקה העיקרית של אלגוריתם החיפוש אשר משפיעה על הערך של f.

<u>חלק י':</u>

<u>סעיף א':</u>

כזכור, בכיתה הצגנו את אלגוריתם A^* שהינו שלם וקביל. לאחר מכן, הצגנו את אלג' *IDA שמטרתו הייתה לשפר מדד ביצועי כלשהו של אלג' A^*. ציין במילה **אחת** מהו אותו מדד ביצועי עבורו אלג' *IDA עדיף תמיד על פני אלג' *A. הסבר (עד 2 שורות).

<u>תשובה:</u>

המדד הביצועי: זיכרון

<u>הסבר:</u> באלגוריתם *A צריכת הזיכרון פרופורציונית למספר הצמתים שנוצרו, כלומר הצמתים שנמצאים ב-OPEN וב-CLOSE. באלגוריתם *IDA צריכת הזיכרון היא לינארית באורך המסלול מאחר והיא עובדת לפי העמקה הדרגתית, כלומר מבצעת שימוש חוזר במתודולוגיית אלגוריתם חיפוש לעומק, לעומק הולך וגדל בכל איטרציה וכך היא משיגה מדד ביצועי של זיכרון טוב יותר וכפי שהראינו בתרגול מתקיים כי סיבוכיות המקום שלהם היא: $IDA^* = O(bd) < O(b^d) = A^*$.

<u>הערה:</u> b מקדם סיעוף, d העומק המקסימלי

<u>סעיף (ב) – 5 נק' יבש</u>

(i) (1 נק' יבש) באיזה מדד ביצועי אלג' *IDA עלול להיות משמעותית פחות טוב מאשר אלג' A^* במקרים רבים? תשובה עד 2 מילים.
<u>תשובה:</u> פיתוח צמתים

(ii) (2 נק' יבש) למה מדד זה נפגע ב- *IDA (לעומת A^*)? תשובה עד שורה אחת.
<u>תשובה:</u> אלגוריתם *IDA מפתח מחדש צמתים שהוא כבר פיתח מבלי לדעת שהוא כבר היה בהם, וזאת מאחר והוא אינו שומר אילו מהמצבים הוא כבר פיתח לעומת *A.

(iii) (2 נק' יבש) האם מדד זה נפגע באותו האופן כמו שהוא נפגע ב- ID-DFS לעומת BFS? אם כן, למה? אם לא, מה ההבדל? תשובה עד 3 שורות.
<u>תשובה:</u>

לא, מדד זה לא נפגע באותו אופן: תחילה, נבחין כי *IDA ו-ID-DFS פועלים באופן דומה ומפתחים צמתים מבלי לדעת האם הם פותחו בעבר ולכן יכולים לפתח את אותו הצומת יותר מפעם אחת. לכן, ההבדל העיקרי באופן הפגיעה במדדים הוא באופן הפעולה ב-*A לעומת BFS. מאופן פעולתו, אלגוריתם BFS לא מפתח שוב צומת שכבר פיתח, אולם ייתכן כי *A יגיע לצומת <u>שכבר פיתח</u> אך הפעם בדרך זולה יותר מהמקורית ולכן יצטרך לטפל במקרה זה בהתאם. מכאן, נסיק כי המדד פגע פחות מאשר פגע ב-ID-DFS לעומת BFS.

(iv) סעיף (ג) – 6 נק' יבש

(v) אלג' $\mathcal{A}_1$ דומה ל- IDA* (הרגיל), עם השינויים הבאים:

(א) משנים את ערך ה- f-limit ההתחלתי להיות $Q_k\big(h(I)\big)$ =: f-limit.

(ב) משנים את כלל העדכון של f-limit באופן שהוסבר בתרגיל.

(i) (3 נק') כמה איטרציות לכל היותר יבצע $\mathcal{A}_1$ על $\mathcal{S}$? הסבר (לכל היותר 3 שורות).

תשובה:

באלגוריתם $\mathcal{A}_1$ ניתן לראות כי העומק החדש שהוגדר בין 2 צעדים הוא $1/k$, כלומר לפי הגדרת $nextFLimit$ ההפרש בין $nextFLimit - prevFlimit$ הוא לכל הפחות $1/k$. בנוסף, נתון כי C_S^* הוא הפתרון האופטימלי במרחב ומכך נסיק כי הפתרון של האלגוריתם $\mathcal{A}_1$ יהיה בטווח $1/k$ של $Q_k(C_S^*)$.כעת, נסתכל על ההפרש בין f_limit ההתחלתי לבין f_limit המינימלי הסופי האפשרי על מנת לקבל את מספר האיטרציות המקסימלי ונחל בגודל הצעד שביצענו, כלומר $1/k$ (ההפרש המינימלי שבין 2 חסמים בכל איטרציה ואיטרציה. מכאן, נקבל כי מספר האיטרציות הוא:

$$\frac{[(Q_k(C_S^*) + \frac{1}{k} - Q_k(h(I)))]}{1/k} = kQ_k(C_s^*) + 1 - kQ_k(h(I))$$

(ii) (3 נק') ספק חסם עליון הדוק עבור $\varepsilon(\mathcal{A}_1, \mathcal{S})$. הסבר (לכל היותר 3 שורות).

תשובה:

נסתכל על האיטרציה האחרונה לפני מציאת הפתרון ונבחין כי בעת החסם של $nextFLimit$ הוא המקסימום בין $Q_k(C_s^*)$ לבין $Q_k(prevFlimit) + \frac{1}{k}$. בנוסף, הפונקציה Q_k מוגדרת באופן הבא: $Q_{k(x)} = \frac{l}{k}$ עבור $l \in [\frac{l}{k}, \frac{l+1}{k})$, כלומר ההגדלה האחרונה של f_limit בלפחות $1/k$ היא זו שתאפשר לנו למצוא את הפתרון עבור האלגוריתם $\mathcal{A}_1$ ולכן ההפרש בין העלות C_S^* לבין העלות של הפתרון שמצאנו הוא לכל היותר $1/k$ ונקבל כי $\varepsilon(A_1, s) \leq \frac{1}{k}$ כלומר החסם העליון הוא $1/k$.

<u>הערה:</u> במספר שאלות הייתה דרישה של מספר שורות ולעיתים לצערנו חרגנו מכך על מנת לרשום תשובה מספקת שמוסברת כמו שצריך. נשמח אם תוכלו להתחשב בכך, ניסינו לצמצם כמה שאפשר והשקענו שעות רבות בעבודה לצורך הבנת החומר לעומקו.