**מבוא לבינה מלאכותית (236501)**

**תרגיל בית 1**

חלק א':

|  |  |  |  |
| --- | --- | --- | --- |
| Calculate time |  |  | K |
| 0.0338 | 6.56 | 3628800 | 10 |
| 5.8 [sec] | 9.8 | 6227020800 | 13 |
| 20.3 [mins] | 12.11 | 1.3x | 15 |
| 5.41 [hours] | 13.32 | 2.09x | 16 |
| 3.83 [days] | 14.55 | 3.556x | 17 |
| 71.84 [years] | 18.38 | 2.432x | 20 |
| 1508.8 [years] | 19.7 | 5.109x | 21 |
| 18.32 [million years] | 23.79 | 6.204x | 24 |

חלק ב':

1. ערכי הקיצון האפשריים של דרגת יציאה במרחב החיפוש הם:

0 – כאשר נגיע למצב שהוא "בור", דבר שיקרה כאשר הגענו אל מצב סופי, וממנו אנחנו sעד אשר נגיע למעבדה האחרונה שלא ביקרנו בה וממנה לא נוכל להמשיך יותר לאף מקום (כי היינו בכל הדירות והיינו בכל המעבדות).

m+k –מהמצב ההתחלתי, נוכל להמשיך לכל אחת מהמעבדות (m) ע"מ לקחת עוד בדיקות **או** להמשיך לכל אחת מהדירות (k) ע"מ לעשות בדיקות.

1. במרחב המצבים שלנו **לא** יתכנו מעגלים וזאת כיוון שעפ"י המגבלות לא ניתן לחזור אל אותה דירה פעמיים ואילו במצב שבו נחזור בשנית אל מעבדה (רק עבור פריקת בדיקות מהאמבולנס) ערך Transferred יהיה **שונה** ולכן זה לא אותו מצב אלא רק מצב אחר שמייצגת את אותה מעבדה.
2. מהגדרת מרחב המצבים ניתן לשים לב כי מספר המטושים ההתחלתי שייך לקבוצת המספרים הטבעיים ולכן יש אינסוף מצבים התחלתיים כאלה שבכל אחד מספר מטושים התחלתי אחר, לכן יש אינסוף מצבים במרחב רק עם מצבים התחלתיים.

בגלל אבחנה זאת, לא כל המצבים במרחב הינם ישיגים כיוון שלמשל, אם במצב ההתחלתי יש באמבולנס כמות מטושים שיחד עם כל המטושים שניתן לאסוף בכל המעבדות הינה 500 מטושים, לא נוכל להשיג מצב אחר שבו יש 1000 מטושים, מצב שראינו שקיים כי כמות המטושים היא אינסופית.

1. **ייתכנו** בורות שהם אינם מצב מטרה כיוון שנוכל למשל להגיע אל מצב שמייצג מעבדה עם כמות מטושים קטנה כלשהי כך שמשם לא נוכל להמשיך אל אף מעבדה אחרת כיוון שכבר ביקרנו בכל שאר המעבדות וגם לא נוכל להמשיך אל דירה שלא היינו בה כי אין לנו מספקים מטושים לבדוק את כל דיירי הדירה.
2. טווח האורכים האפשריים של מסלולים במרחב ממצב התחלתי אל מצב סופי הינו:

[(k+1) – (k+m)] , המינימום יהיה מהמצב ההתחלתי דרך כל הדירות (k) ברצף ומשם אל מעבדה עם עוד קשת. המקסימום יהיה למשל מעבר בכל הדירות (k) ומשם מעבר לכל המעבדות (m) או מעבר דירה-מעבדה-דירה לסירוגין עד שנגיע למצב מטרה.

1. Succ(s) = {*|*

*)AmbulanceTestCapacity - )}*

*{|*

*(s.Taken)}*

*חלק ד':*

*8 - g.*

*הפלט שקיבלנו הוא*

StreetsMap(src: 54 dst: 549)  UniformCost  time: 1.44   #dev: 17354   |space|: 17514    total\_g\_cost:  7465.52560   |path|: 136   path: [   54  ==>     55  ==>     56  ==>     57  ==>     58  ==>     59  ==>     60  ==>  28893  ==>  14580  ==>  14590  ==>  14591  ==>  14592  ==>  14593  ==>  81892  ==>  25814  ==>     81  ==>  26236  ==>  26234  ==>   1188  ==>  33068  ==>  33069  ==>  33070  ==>  15474  ==>  33071  ==>   5020  ==>  21699  ==>  33072  ==>  33073  ==>  33074  ==>  16203  ==>   9847  ==>   9848  ==>   9849  ==>   9850  ==>   9851  ==>    335  ==>   9852  ==>  82906  ==>  82907  ==>  82908  ==>  82909  ==>  95454  ==>  96539  ==>  72369  ==>  94627  ==>  38553  ==>  72367  ==>  29007  ==>  94632  ==>  96540  ==>   9269  ==>  82890  ==>  29049  ==>  29026  ==>  82682  ==>  71897  ==>  83380  ==>  96541  ==>  82904  ==>  96542  ==>  96543  ==>  96544  ==>  96545  ==>  96546  ==>  96547  ==>  82911  ==>  82928  ==>  24841  ==>  24842  ==>  24843  ==>   5215  ==>  24844  ==>   9274  ==>  24845  ==>  24846  ==>  24847  ==>  24848  ==>  24849  ==>  24850  ==>  24851  ==>  24852  ==>  24853  ==>  24854  ==>  24855  ==>  24856  ==>  24857  ==>  24858  ==>  24859  ==>  24860  ==>  24861  ==>  24862  ==>  24863  ==>  24864  ==>  24865  ==>  24866  ==>  82208  ==>  82209  ==>  82210  ==>  21518  ==>  21431  ==>  21432  ==>  21433  ==>  21434  ==>  21435  ==>  21436  ==>  21437  ==>  21438  ==>  21439  ==>  21440  ==>  21441  ==>  21442  ==>  21443  ==>  21444  ==>  21445  ==>  21446  ==>  21447  ==>  21448  ==>  21449  ==>  21450  ==>  21451  ==>    621  ==>  21452  ==>  21453  ==>  21454  ==>  21495  ==>  21496  ==>    539  ==>    540  ==>    541  ==>    542  ==>    543  ==>    544  ==>    545  ==>    546  ==>    547  ==>    548  ==>    549]

*חלק ה':*

*הפלט שקיבלנו*

StreetsMap(src: 54 dst: 549)          A\* (h=AirDist, w=0.500)       time:   0.16   #dev: 2015    |space|: 2229     total\_g\_cost:  7465.52560   |path|: 136   path: [   54  ==>     55  ==>     56  ==>     57  ==>     58  ==>     59  ==>     60  ==>  28893  ==>  14580  ==>  14590  ==>  14591  ==>  14592  ==>  14593  ==>  81892  ==>  25814  ==>     81  ==>  26236  ==>  26234  ==>   1188  ==>  33068  ==>  33069  ==>  33070  ==>  15474  ==>  33071  ==>   5020  ==>  21699  ==>  33072  ==>  33073  ==>  33074  ==>  16203  ==>   9847  ==>   9848  ==>   9849  ==>   9850  ==>   9851  ==>    335  ==>   9852  ==>  82906  ==>  82907  ==>  82908  ==>  82909  ==>  95454  ==>  96539  ==>  72369  ==>  94627  ==>  38553  ==>  72367  ==>  29007  ==>  94632  ==>  96540  ==>   9269  ==>  82890  ==>  29049  ==>  29026  ==>  82682  ==>  71897  ==>  83380  ==>  96541  ==>  82904  ==>  96542  ==>  96543  ==>  96544  ==>  96545  ==>  96546  ==>  96547  ==>  82911  ==>  82928  ==>  24841  ==>  24842  ==>  24843  ==>   5215  ==>  24844  ==>   9274  ==>  24845  ==>  24846  ==>  24847  ==>  24848  ==>  24849  ==>  24850  ==>  24851  ==>  24852  ==>  24853  ==>  24854  ==>  24855  ==>  24856  ==>  24857  ==>  24858  ==>  24859  ==>  24860  ==>  24861  ==>  24862  ==>  24863  ==>  24864  ==>  24865  ==>  24866  ==>  82208  ==>  82209  ==>  82210  ==>  21518  ==>  21431  ==>  21432  ==>  21433  ==>  21434  ==>  21435  ==>  21436  ==>  21437  ==>  21438  ==>  21439  ==>  21440  ==>  21441  ==>  21442  ==>  21443  ==>  21444  ==>  21445  ==>  21446  ==>  21447  ==>  21448  ==>  21449  ==>  21450  ==>  21451  ==>    621  ==>  21452  ==>  21453  ==>  21454  ==>  21495  ==>  21496  ==>    539  ==>    540  ==>    541  ==>    542  ==>    543  ==>    544  ==>    545  ==>    546  ==>    547  ==>    548  ==>    549]

*מספר פיתוחי המצבים היחסי שחסכנו לעומת הריצה העיוורת הוא*

*12.*

A close up of a map

Description automatically generated

*ניתן לראות באופן כללי את כלל האצבע בגרף, כך שככל ש-W עולה מחיר הפיתרון עולה, אך מספר פיתוחי הצמתים יורד. אך ניתן לראות שהכלל הזה אינו מופיע בגרף באופן גורף, למשל ניתן לראות שמחיר הפיתרון ב- קטן יותר מאשר המחיר ב-.*

*ערך W שנבחר הוא 0.57, (נקודת מפגש הגרפים) כיוון שבערך זה יש את האיזון האופטימלי בין שני המדדים. מצד אחד עלות נמוכה יחסית וגם מספר פיתוחי צמתים נמוך.*

*חלק ו':*

*15.*

(i)

@dataclass(frozen=True)

(ii) שורה זו למעשה לא מספיקה כיוון שיתכן שהמשתנים הפנימיים של האובייקט יהיו mutable ולכן נצטרף לגרום גם להם להיות immutable, כפי שנעשה בתוך המחלקה של MDAState עבור המשתנים הפנימיים עם ההצהרה FrozenSet:

current\_site: Union[Junction, Laboratory, ApartmentWithSymptomsReport]  
tests\_on\_ambulance: FrozenSet[ApartmentWithSymptomsReport]  
tests\_transferred\_to\_lab: FrozenSet[ApartmentWithSymptomsReport]  
nr\_matoshim\_on\_ambulance: int  
visited\_labs: FrozenSet[Laboratory]

(iii)   
במקרה ולמשל השדה tests\_on\_ambulance של המחלקה MDAState היה set או כל מבנה אחר שהוא mutable, היינו יכולים לעשות משהו כזה:

# apartments successors  
for apartment in self.get\_reported\_apartments\_waiting\_to\_visit(state\_to\_expand):  
 # if apartment can be visited

if apartment.nr\_roommates<=state\_to\_expand.nr\_matoshim\_on\_ambulance …:  
  
 tests\_on\_ambulance = state\_to\_expand.tests\_on\_ambulance  
 tests\_on\_ambulance.add(apartment)  
  
 ……  
 succ\_state = MDAState(apartment, tests\_on\_ambulance,  
 state\_to\_expand.tests\_transferred\_to\_lab,  
 new\_matoshim\_num, state\_to\_expand.visited\_labs)  
 yield OperatorResult(succ\_state, …)

כלומר כדי ליצור מצב חדש עם קבוצת הדירות המעודכנת (הכוללת את הדירה שנרצה לבקר) היינו יכולים לבצע השמת משתנים, ואז לבצע פעולת add על המשתנה החדש. אבל המשתנה החדש הוא למעשה מצביע לאותו מקום בזיכרון בו שמור ה-set של המצב ה**מקורי** ולא העתק שלו, ולכן למעשה נעדכן את המצב המקורי ונוסיף גם לו את הדירה שנרצה לבקר בה.

18. נוכיח כי היוריסטיקה הנ"ל קבילה.

נניח כי האמבולנס נמצא בצומת N וכי יתר הצמתים שעליו לעבור מהן מיוצגות ע"י הקבוצה A.

יהיו שתי צמתים x,y המייצגות את המרחק המקסימאלי בין שתי צמתים כלשהן בקבוצה A. לכן, h(N)=AirDistance(x,y) ולכן אורך המסלול של האמבולנס לעבור בהכרח גדול או שווה למרחק h(N) וזה כי המרחק האווירי (מרחק אוקלידס) בין x ל-y הוא המרחק המינימאלי בין שתי הצמתים הללו וכל מרחק אחר שיילקח בהכרח יהיה לכל הפחות שווה לו, לכן,

h(N) , וכמובן שגם מתקיים 0 כיוון וסוכמים משהו אי שלילי.

21. נראה כי היוריסטיקה הנ"ל אינה קבילה באמצעות דוגמא נגדית.

נסתכל על פיתוח הצמתים הבא:



אם נעבור בין הצמתים עפ"י SumAirDistance אז כאשר צומת ההתחלה היא הצומת השחורה, נלך ימינה אל הצומת האדומה, אז שמאלה אל הצומת הצהובה ואז שוב ימינה אל הצומת הירוקה, לכן בסה"כ נעשה מרחק של 2+5+11=18, אך ישנו מרחק קצר יותר שמתחיל מהצומת השחורה, הולך שמאלה את הצומת הצהובה, ומשם ימינה אל הצומת האדומה ואז אל הצומת הירוקה, שזה בסה"כ מרחק של 3+5+6=14, לכן לא מתקיים התנאי h(N) ולכן היוריסטיקה הנ"ל אינה קבילה.

24. נוכיח כי היוריסטיקה הנ"ל קבילה.

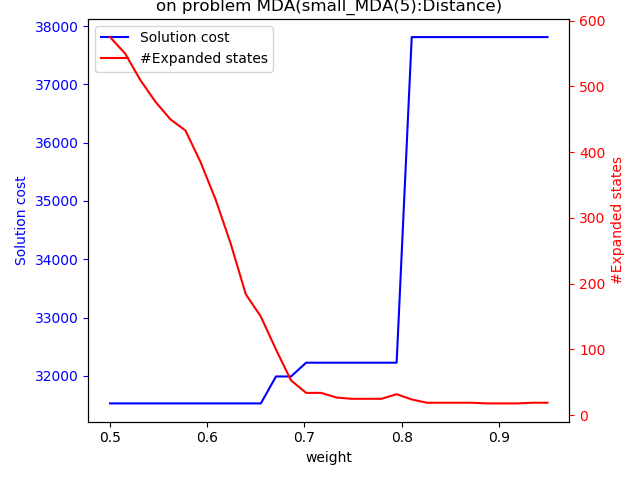
נסתכל על גרף הצמתים שנותרו כאשר ישנה קשת בין כל צומת וצומת והמשקל של כל קשת שווה למרחק דרכים בין הצמתים, נסמנו A. כעת נסתכל הגרף של כל הצמתים שעוד נותרו, כאשר המשקלים על הקשתות הם המרחקים האוויריים בין כל צומת, נסמנו B. נמצא עפ"מ של A המייצג מסלול הקיים גם ב-B. כיוון שהמרחק האווירי הוא המרחק המינימאלי בין כל שתי קשתות, משקל אותו מסלול בB יהיה קטן שווה למסלול הנ"ל בA, כמובן שמסלול זה מהווה עץ פורש בB, ולכן משקלו גדול שווה למשקל של עפ"מ בB. כמובן שמשקלי הקשתות חיוביים ולכן משקל עפ"מ הינו חיובי ולכן היוריסטיקה קבילה.

25. עבור MDAMSTAirDistanceHutistic מתקבל הגרף הנ"ל:

A screenshot of a cell phone

Description automatically generated

ועבור MDASumAirHuristic מתקבל הגרף הנ"ל:



ניתן לראות שהגרפים מקיימים את הכלל מסעיף 12 שככל שהמשקל W קטן יותר כך הפתרון איכותי יותר ואילו מספר המצבים שפותחו גדול יותר. הייתי בוחר W=0.67 שעבורו בגרף השני איכות הפתרון ומספר המצבים המפותחים זהה וגם נקבל פתרון באיכות גבוה עם מספר מצבים מפותחים יחסית קטן.

חלק ז׳:

26. הפרכה. נראה כי ההיוריסטיקה הנ"ל אינה קבילה עבור . נציג את הדוגמא הנגדית הבאה עבור מרחב עם דירה אחת עם דייר אחד, ומעבדה אחת. (הצומת השחור הוא המיקום ההתחתלי, הכחול הוא הדירה והירוק הוא המעבדה) נניח כי אנחנו במיקום ההתחלתי.

1 m

1000 m

ההיוריסטיקה במקרה הזה מחזירה 1000, המרחק הגדול ביותר. אבל המסלול הקיים הוא בעל זול יותר - . ולכן ההיוריסטיקה אינה קבילה.

27. הפרכה. נראה כי ההיוריסטיקה הנ"ל אינה קבילה עבור . נציג את הדוגמא הנגדית הבאה עבור מרחב עם שתי דירות כל אחת עם דייר אחד, ומעבדה אחת. (הצומת השחור הוא המיקום ההתחלתי, הכחולים הם הדירות והירוק הוא המעבדה) נניח כי אנחנו במיקום ההתחלתי.

1

1 m

1 m

1.5 m

1 m

1

1 m

ההיוריסטיקה במקרה הזה מתחילה באחת הדירות, ואז עוברת לדירה היחידה שנותרה, ולכן הערך שהיא מחזירה הוא . אבל קיים מסלול בעל זול יותר. המסלול המתחיל באחת הדירות, מעביר את הבדיקה שלה למעבדה, נוסע לדירה השניה ומעביר את הבדיקה שלה למעבדה. נקבל ערך . ולכן ההיוריסטיקה אינה קבילה.

28. הפרכה. נראה כי ההיוריסטיקה הנ"ל אינה קבילה עבור . נציג את הדוגמא הנגדית הבאה עבור מרחב עם שתי דירות כל אחת עם דייר אחד, ומעבדה אחת. (הצומת השחור הוא המיקום ההתחתלי, הכחולים הם הדירות והירוק הוא המעבדה) נניח כי אנחנו במיקום ההתחלתי.

1 m

1 m

1

1.5 m

2 m

1 m

1

ההיוריסטיקה במקרה הזה ,תבנה את העץ הפורש המינימלי הבא:

1.5 m

1 m

1

1

ולכן הערך שהיא מחזירה הוא . אבל קיים מסלול בעל זול יותר. המסלול המתחיל באחת הדירות, מעביר את הבדיקה שלה למעבדה, נוסע לדירה השניה ומעביר את הבדיקה שלה למעבדה. נקבל ערך . ולכן ההיוריסטיקה אינה קבילה.

30.

1. נוכיח כי ההיוריסטיקה הנ"ל היא קבילה.   
   נסתכל על בדיקה כלשהי – המרחק המינימלי שהיא יכולה לעבור מרגע אסיפתה מהדירה ועד הגעתה למעבדה הוא כאשר מיד מרגע אסיפתה מועברת ישירות למעבדה הכי קרובה. אחרת המרחק הנ"ל יהיה בהכרח גדול יותר. אם נסמן את המרחק המינמלי שהיא עוברת ב- אז מתקיים . כיוון שזהו הערך המינמלי שמסלול יכול לקבל אז נקבל כי (המרחק תמיד חיובי) ולכן ההיוריסטיקה קבילה.
2. אם המרחק בין כל דירה ומעבדה הכי קרובה אליה הוא המרחק האווירי (כלומר יקים כביש ישר ביניהם) אז מתקיים כי וזה כיוון שההיוריסטיקה הזו במקרה הזה מחשבת את המרחק **האמיתי** למטרה.

31.

time: 51.69 #dev: 29148 |space|: 41799 total\_g\_cost: 104387.48471 total\_cost: MDACost(dist= 124686.816m, tests-travel= 104387.485m) |path|: 17

בתוצאות שקיבלנו בסעיף זה ניתן לראות כי ערך ה-test-travel הוא 104387.485 לעומת מה שקיבלנו בסעיפים קודמים 131811.935.

את הסעיפים הבאים עשינו לפי גרסה א'

32.

הוכחה

נניח כי קיים פתרון במרחב המקורי. נסמן אותו ב- כאשר וגם . כיוון שזהו פתרון חוקי מתקיים כי וגם . ולכן מתקיים כי לכל וגם . מכאן נקבל כי הסדרה הינה סדרת פיתוחים חוקית ב- ולכן האלגוריתם מחזיר פתרון (יכול להיות עם COST של אבל בהכרח יהיה פתרון)

33.

הפרכה

נראה שעבור המרחב הבא ( שתי דירות בכל דירה דייר אחד ומעבדה אחת) ועבור הפתרון המוחזר ע"י האלגוריתם לא אופטימלי. (נניח כי יש מספיק מטושים התחלתיים – 2 למשל-)

A

1 m

L

O

1 m

1 m

B

במרחב הנ"ל אפשר לעבור בשתי הדירות ואז למעבדה, או לעבור לדירה כלשהי ואז למעבדה ואז לדירה השניה ולבסוף במעבדה. המסלולים בגישה הראשונה הינם באורך 5, ובגישה השניה 7, ולכן המסלולים של שתי הגישות נמצאים ב-DistEpsOptimal של הבעיה.   
מתקיים שה- בגישה הראשונה הוא 6, ובגישה השניה הוא 4 ולכן עבור אחד המסלולים או . נראה שהאלגוריתם לא מחזיר אחד מהמסלולים האלה. נצייר את גרף הפיתוחים (כל צומת מייצג מסלול, הקשתות מייצגות מעבר ממסלול למסלול העוקב, המספר על הקשת מייצג את עלות האופרטור, המספר מתחת לצומת מייצג את העלות המצטברת, והמספרים בכחול מעל לצמתים מיצגים את סדר פיתוח הצמתים לפי אלג' UC) . האלגוריתם מחזיר אחד מהמסלולים המסומנים באדום כיוון שלהם יש את העלות המינימילית. אבל ה- של מסלולי אלה הוא 6.

*A picture containing text, map

Description automatically generated*

הערה: מתעלמים כאן מהמסלולים מהסוג למרות שהעלות של מסלול כזה היא 0, וזה כיוון שבהכרח באיזהו שלב מגיעים ליותר מ- ואז העלות הופכת ל-.

34.

קיבלנו את התוצאה הבאה

**MDA(moderate\_MDA(8):TestsTravelDistance) A\* (h=MDA-TimeObjectiveSumOfMinAirDistFromLab, w=0.500) time: 26.31 #dev: 28747 |space|: 40472 total\_g\_cost: 104387.48471 total\_cost: MDACost(dist= 90441.473m, tests-travel= 104387.485m) |path|: 16**  path: [(loc: initial-location tests on ambulance: [] tests transferred to lab: [] #matoshim: 3 visited labs: []) ==(go to lab Lowman-Kohn)==> (loc: lab Lowman-Kohn tests on ambulance: [] tests transferred to lab: [] #matoshim: 9 visited labs: ['Lowman-Kohn']) ==(visit Hana Hockman)==> (loc: test @ Hana Hockman tests on ambulance: ['Hana Hockman (2)'] tests transferred to lab: [] #matoshim: 7 visited labs: ['Lowman-Kohn']) ==(go to lab Woolum-Mulholland)==> (loc: lab Woolum-Mulholland tests on ambulance: [] tests transferred to lab: ['Hana Hockman (2)'] #matoshim: 11 visited labs: ['Woolum-Mulholland', 'Lowman-Kohn']) ==(visit Krysta Valentine)==> (loc: test @ Krysta Valentine tests on ambulance: ['Krysta Valentine (3)'] tests transferred to lab: ['Hana Hockman (2)'] #matoshim: 8 visited labs: ['Woolum-Mulholland', 'Lowman-Kohn']) ==(go to lab Woolum-Mulholland)==> (loc: lab Woolum-Mulholland tests on ambulance: [] tests transferred to lab: ['Krysta Valentine (3)', 'Hana Hockman (2)'] #matoshim: 8 visited labs: ['Woolum-Mulholland', 'Lowman-Kohn']) ==(visit Cleora Alaniz)==> (loc: test @ Cleora Alaniz tests on ambulance: ['Cleora Alaniz (4)'] tests transferred to lab: ['Krysta Valentine (3)', 'Hana Hockman (2)'] #matoshim: 4 visited labs: ['Woolum-Mulholland', 'Lowman-Kohn']) ==(go to lab Lowman-Kohn)==> (loc: lab Lowman-Kohn tests on ambulance: [] tests transferred to lab: ['Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Hana Hockman (2)'] #matoshim: 4 visited labs: ['Woolum-Mulholland', 'Lowman-Kohn']) ==(visit Kurt Dockstader)==> (loc: test @ Kurt Dockstader tests on ambulance: ['Kurt Dockstader (4)'] tests transferred to lab: ['Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Hana Hockman (2)'] #matoshim: 0 visited labs: ['Woolum-Mulholland', 'Lowman-Kohn']) ==(go to lab Neri-Basta)==> (loc: lab Neri-Basta tests on ambulance: [] tests transferred to lab: ['Kurt Dockstader (4)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Hana Hockman (2)'] #matoshim: 4 visited labs: ['Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(visit Raven Woolum)==> (loc: test @ Raven Woolum tests on ambulance: ['Raven Woolum (2)'] tests transferred to lab: ['Kurt Dockstader (4)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Hana Hockman (2)'] #matoshim: 2 visited labs: ['Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(go to lab Bouldin-Boyland)==> (loc: lab Bouldin-Boyland tests on ambulance: [] tests transferred to lab: ['Hana Hockman (2)', 'Kurt Dockstader (4)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Raven Woolum (2)'] #matoshim: 6 visited labs: ['Bouldin-Boyland', 'Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(visit Gussie Foran)==> (loc: test @ Gussie Foran tests on ambulance: ['Gussie Foran (2)'] tests transferred to lab: ['Hana Hockman (2)', 'Kurt Dockstader (4)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Raven Woolum (2)'] #matoshim: 4 visited labs: ['Bouldin-Boyland', 'Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(visit Veronique Katz)==> (loc: test @ Veronique Katz tests on ambulance: ['Veronique Katz (1)', 'Gussie Foran (2)'] tests transferred to lab: ['Hana Hockman (2)', 'Kurt Dockstader (4)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Raven Woolum (2)'] #matoshim: 3 visited labs: ['Bouldin-Boyland', 'Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(go to lab Neri-Basta)==> (loc: lab Neri-Basta tests on ambulance: [] tests transferred to lab: ['Veronique Katz (1)', 'Hana Hockman (2)', 'Kurt Dockstader (4)', 'Gussie Foran (2)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Raven Woolum (2)'] #matoshim: 3 visited labs: ['Bouldin-Boyland', 'Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(visit Pierre Lowman)==> (loc: test @ Pierre Lowman tests on ambulance: ['Pierre Lowman (3)'] tests transferred to lab: ['Veronique Katz (1)', 'Hana Hockman (2)', 'Kurt Dockstader (4)', 'Gussie Foran (2)', 'Cleora Alaniz (4)', 'Krysta Valentine (3)', 'Raven Woolum (2)'] #matoshim: 0 visited labs: ['Bouldin-Boyland', 'Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn']) ==(go to lab Lowman-Kohn)==> (loc: lab Lowman-Kohn tests on ambulance: [] tests transferred to lab: ['Hana Hockman (2)', 'Kurt Dockstader (4)', 'Cleora Alaniz (4)', 'Veronique Katz (1)', 'Gussie Foran (2)', 'Krysta Valentine (3)', 'Pierre Lowman (3)', 'Raven Woolum (2)'] #matoshim: 0 visited labs: ['Bouldin-Boyland', 'Woolum-Mulholland', 'Neri-Basta', 'Lowman-Kohn'])]

נשווה בין התוצאות השונות מהסעיפים הקודמים עבוא אותה הבעיה (moderate)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tests-Travel Cost | Dist Cost | Space | Dev | Heuristic Function | Optimization objective | #Ex |
| 131811.935 | 58254.187 | 41773 | 33261 | MaxAirDist | Distance | 17 |
| 131811.935 | 58254.187 | 38550 | 26966 | SumAirDist | Distance | 20 |
| 131811.935 | 58254.187 | 40024 | 29766 | MSTAirDist | Distance | 23 |
| 104387.485 | 124686.816 | 41799 | 29148 | NearestLab | Tests-Travel | 31 |
| **104387.485** | **90441.473** | **40472** | **28747** | **MST+NearestLab** | **Both** | **34** |

ניתן לראות כי הערך עבור ה- tests-travel cost הינו המינימלי מבין הפלטים, ובנוסף זה לא בא על חשבון ה-Dist Cost כפי שקיבלנו בסעיף 31. כלומר קיבלנו ערך המדד tests-travel cost מינימלי וגם ערך Dist Cost שאינו חורג מ-160% מהפיתרון האופטימלי עבור מדד זה.

35.

הפרכה

*36.  
הוכחה*

37.

חלק ח׳:

39.

MDA(small\_MDA(5):Distance) A\* (h=MDA-MST-AirDist, w=0.500) time: 0.99

#dev: 575 |space|: 947 total\_g\_cost: 31528.65909 total\_cost:

MDACost(dist= 31528.659m, tests-travel= 52112.429m) |path|: 8

MDA(small\_MDA(5):Distance) A\*eps (h=MDA-MST-AirDist, w=0.500) time: 2.22 #dev: 564 |space|: 933 total\_g\_cost: 31528.65909 total\_cost:

MDACost(dist= 31528.659m, tests-travel= 52112.429m) |path|: 8

אכן חסכנו בפיתוח של 11 מצבים.

ציפינו מראש לחסוך במספר הפיתוחים בתצורה שבו הרצנו את A\*epsilon שבה הפונקציה היוריסטית הקבילה שימשה את דירוג הצמתים בopen- והיוריסטיקה הלא קבילה שימשה לבחירת הצומת המתאים מה-focal. מכיוון שבתצורה בה הרצנו את A\*epsilon, בחרנו כל פעם את הצמתים המתאימים מה open בדיוק כמו ש-A\* עושה אבל גם שיפרנו את A\* בכך שמתוך הצמתים המתאימים בopen- בכל שלב בחרנו את הצומת הכי טוב בעזרת הפונקציה היוריסטית המיודעת שלנו (sum) .

חלק ט׳:

41.

MDA(moderate\_MDA(8):Distance) Anytime-A\* (h=MDA-MST-AirDist, w=0.800)

time: 4.22 #dev: 1027 |space|: 740 total\_g\_cost: 64055.65000

total\_cost: MDACost(dist= 64055.650m, tests-travel= 131870.337m) |path|: 13

עבור משקל גדול יותר על הקשתות ביחס להרצות קודמות של A\* על אותה הבעיה, כאן בהרצת Anytime-A\* זמן החישוב המתקבל, המקום ומספר המצבים המפותחים יוצאים נמוכים יותר בגלל האופטימיזציה של האלגוריתם הנ״ל בפיתוח הצמתים.