**בינה מלאכותית: תרגיל בית מס' 1**

דוד וייצמן, 204734461

עידן שפר, 308342807

**פרק ראשון:**

1. מספר פרמוטציות אפשריות לבעיית המשלוחים עבור :

ללא דלק:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | k |
| 3628800 | 362880 | 40320 | 5040 | 720 | 120 | 24 | 6 | 2 | 1 | מספר פרמוטציות |

עם דלק:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | k |
| 7087500000000 | 141750000000 | 3150000000 | 7875000 | 225000 | 7500 | 300 | 150 | 10 | 1 | מספר פרמוטציות |

1. מקדם הסיעוף המקסימלי הוא , במקרה של גרף מלא וכאשר יש מספיק דלק כדי להגיע מהיעד הנוכחי לכל יעד אחר (כולל תחנות דלק).

מקדם הסיעוף המינימלי הוא מקדם סיעוף 1 במקרה של "שרוך".

1. כן, ייתכנו מעגלים! דוגמה למעגל כזה: מקרה שבו אנו נמצאים בתחנת דלק L1, נוסעים לתחנת הדלק L2 ואז חוזרים לתחנת הדלק L1. במקרה כזה כמות הדלק שלנו תהיה מלאה בשני המקרים, מספר ההזמנות המחכות והגמורות בהן נשאר זהה, והצומת תישאר L1.

*המעגל הינו:*

וזאת כמובן בתנאי שישנו מסלול בין המקיים .

1. גודל מרחב המצבים הוא אינסופי מאחר וכל מצב מכיל מספר ממשי המתאר את כמות הדלק הנוכחית. לא בהכרח כל המצבים ישיגים. לדוגמה אם קיים יעד שמרחקו המינימלי מנקודת ההתחלה ומכל תחנת דלק מקיים אזי כל המצבים מהצורה כך ש- אינם ישיגים.
2. ייתכנו בורות ישיגים שאינם מצבי מטרה. לדוגמה:

אם גודל מיכל הדלק הוא 60 (נניח ) והמפה הנתונה היא:

A

B

50

100

נוצר בור בגרף המצבים של הבעיה כאשר אנחנו בצומת A ולא נותר מספיק דלק להגיע לשום יעד אחר. באופן פורמלי המצב:

יהווה בור ישיג בגרף המצבים של הבעיה שאינו מצב סופי.

1. ראשית נגדיר 2 פונקציות עזר המייצגות את היעדים ואת תחנות הדלק שניתן לעבור אליהן:

כעת נוכל להגדיר את פונקציית העוקב:

1. החסם התחתון הוא כאשר הוא מספר היעדים.

הסבר: לכל הפחות עלינו לבקר בכל היעדים. אם יש לנו מספיק דלק לבקר בכל היעדים ברצף נוכל לעבור על מצבים בגרף המצבים ולהגיע למצב סופי, ובמקרה זה עומק המצב

יהיה .

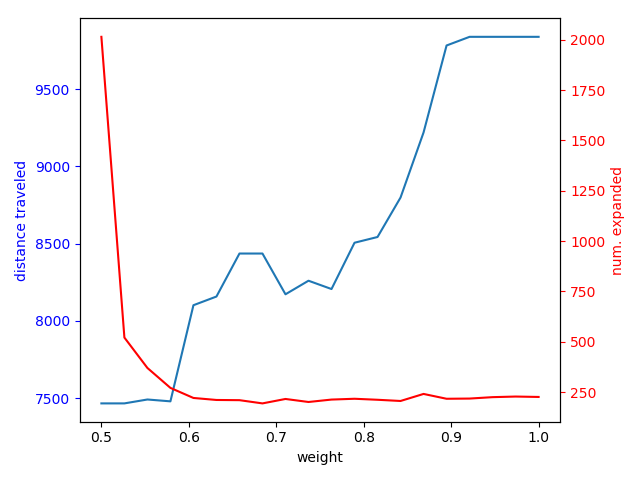
1. פלט הריצה:

Map(src: 54 dst: 549) UniformCost time: 1.23 #dev: 17354 total\_cost: 7465.52960 |path|: 137 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]

11. פלט ריצת A\* עם היוריסטיקת מרחק אווירי:

Map(src: 54 dst: 549) A\* (h=AirDist, w=0.500) time: 0.19 #dev: 2015 total\_cost: 7465.52960 |path|: 137 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]

1. השפעת המשקל w על ריצת אלגוריתם A\* בבעיית המרחקים:



הסבר התוצאות: ככל שאנו נותנים יותר משקל ליוריסטיקה כך יעילות החיפוש משתפרת על חשבון איכות התוצאה. עם הגדלת משקל היוריסטיקה האלגוריתם יעדיף להרחיב צמתים קרובות לפתרון תוך התחשבות פחותה יותר במחיר הפתרון. כצפוי דבר זה יביא להתכנסות מהירה יותר למצב סופי (הרחבת פחות מצבים).

1. היוריסטיקה קבילה. בהינתן מצב כלשהו , הרי שעל מנת להגיע למצב מטרה עלינו לכל הפחות לבקר בכל היעדים ב- T. לכל אחד מהיעדים, הביטוי מהווה חסם תחתון למחיר המסלול האופטימלי בין ל- . לפיכך במקרה בו נקבל

כאשר מחזירה את מחיר המסלול האופטימלי למצב מטרה בגרף המצבים ו- מחזירה את המרחק האופטימלי בין שני יעדים ברשת הכבישים.

כמובן במקרה בו נקבל , בכל מקרה כנדרש.

1. פלט ריצת A\* על בעיית המשלוחים המופשטת עם יוריסטיקת MaxAirDist:

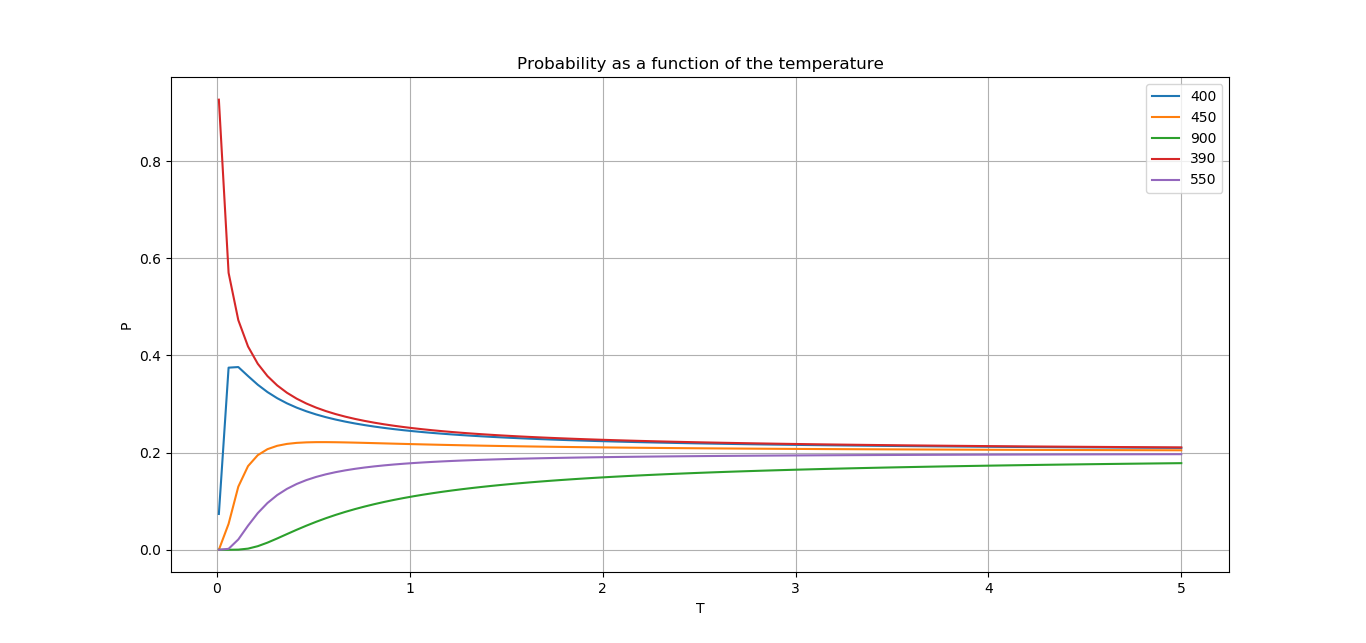
RelaxedDeliveries(big\_delivery) A\* (h=MaxAirDist, w=0.500) time: 8.47 #dev: 3907 total\_cost: 40844.21165 |path|: 11 path: [33919, 18409, 77726, 26690, 31221, 63050, 84034, 60664, 70557, 94941, 31008] gas-stations: [31221, 70557]

1. פלט ריצת A\* על בעיית המשלוחים המופשטת עם יוריסטיקת MSTAirDist:

RelaxedDeliveries(big\_delivery) A\* (h=MSTAirDist, w=0.500) time: 2.42 #dev: 86 total\_cost: 40844.21165 |path|: 11 path: [33919, 18409, 77726, 26690, 31221, 63050, 84034, 60664, 70557, 94941, 31008] gas-stations: [31221, 70557]

ניתן לראות כי מספר הצמתים שהורחבו ירד משמעותית בהשוואה ל- MaxAirDist.

1. השפעת המשקל w על ריצת אלגוריתם A\* בבעיית המשלוחים המופשטת, MSTAirDist:
2. שינוי הסקאלה מהווה הכפלה של בקבוע של המונה והמכנה ולכן לא משנה את ההתפלגות:
3. פונקציית ההסתברות לאיברי הוקטור X, כפונקציה של הטמפרטורה T :



1. כאשר הסתברות האיבר המינימלי שואפת ל-1 ואילו ההסתברות עבור שאר האיברים שואפת לאפס.

הסבר אנליטי :

נסתכל על המונה: לפי ההגדרה של אלפא, הוא שווה לאיבר המינימלי. אם כך, עבור האיבר המינימלי נקבל במונה:

עבור כל איבר אחר (שאינו המינימלי) נקבל בתוך החזקה במונה מספר גדול מ-1. נסמנו b. מתקיים:

כעת נסתכל על המכנה : הוא בוודאות מכיל את האיבר המינימלי ולכן נקבל:

כלומר המכנה לא משפיע על ההתכנסות, ובסה"כ נקבל:

לכן ככל שהטמפרטורה תרד האלגוריתם "יאבד" את אופיו הסטוכאסטי ויקבל אופי חמדני בו תיבחר הצומת בעלת היוריסטיקה המינימאלית.

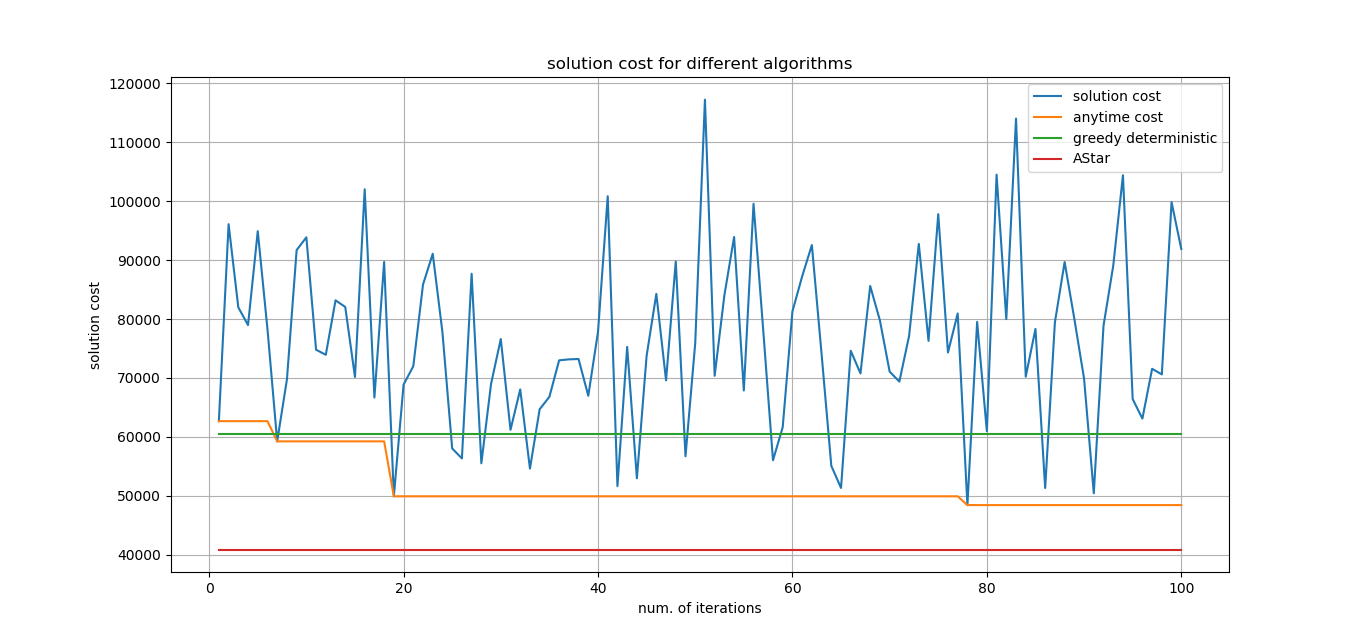
1. כאשר ההסתברות שואפת להסתברות יוניפורמית.

הסבר:

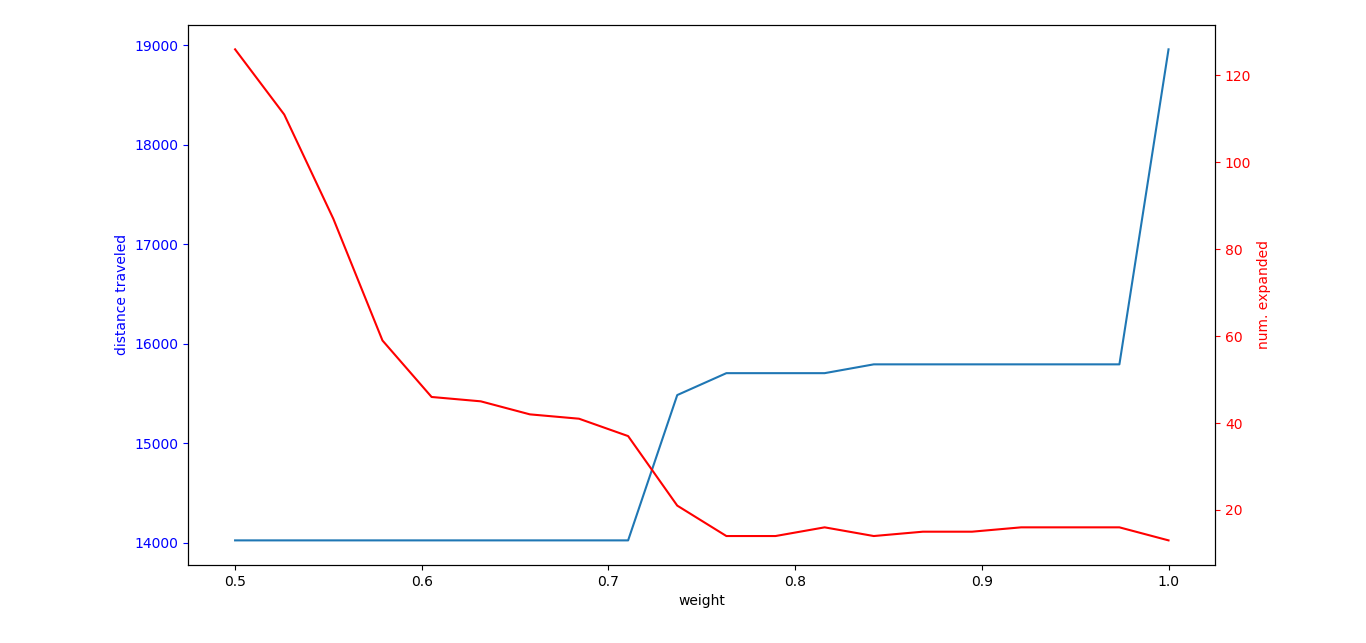
לכן לכל איבר:

ככל שנתחיל מטמפרטורה גבוהה יותר, האלגוריתם יתעדף פחות את הצומת בעלת היוריסטיקה המינימאלית ויתנהג בצורה רנדומלית יותר.

1. תוצאות הרצת האלגוריתם החמדני הסטוכאסטי בהשוואה לאלגוריתם A\* ולאלגוריתם החמדני דטרמיניסטי:



1. השפעת המשקל w על ריצת אלגוריתם A\* בבעיית המשלוחים, MSTAirDist:



1. נסמן את היוריסטיקת ה- RelaxedDeliveries ב- . נוכיח כי ההיוריסטיקה קבילה: נסמן את המרחק האופטימלי ממצב למצב מטרה בבעיית ה- StrictDeliviries ב- . נניח בשלילה כי קיים גרף מצבים בבעיית ה- StrictDeliviries ומצב כך ש- . נסמן ב-P מסלול אופטימלי בגרף המצבים של StrictDeliviries אשר מחירו (בהכרח קיים כזה). ידוע כי המרחק האווירי בין כל שני יעדים בגרף הכבישים מהווה חסם תחתון למרחק המסלול האופטימלי ביניהם, ולכן מחיר מעבר בין מצבים בבעיית ה- RelaxedDeliveries הוא חסם תחתון למחירים בבעיית ה- StrictDeliviries. לפיכך המסלול P מהווה פתרון תקף גם לבעיית ה- RelaxedDeliveries ומתקיים ש- בסתירה לאופטימליות הפתרון לבעיית ה- RelaxedDeliveries. לפיכך בהכרח מתקיים כנדרש.
2. פלט ריצת אלגוריתם A\* על בעיית המשלוחים, RelaxedDeliveriesHeuristic:

StrictDeliveries(small\_delivery) A\* (h=RelaxedProb, w=0.500) time: 11.28 #dev: 83 total\_cost: 14021.87620 |path|: 8 path: [43516, 67260, 17719, 43454, 43217, 32863, 7873, 42607] gas-stations: [17719, 32863]

ניתן לראות כי בהשוואה לריצה בסעיף 26, מספר המצבים שהורחבו ירד מ-120 בקירוב ל- 83. מכאן ניתן להסיק שהיוריסטיקת RelaxedDeliveriesHeuristic מיודעת יותר. החל ממשקל w=0.6 בקירוב מספר המצבים שהורחבו עם MSTAirDistHeuristic ירד גם הוא ל-80 בקירוב, ואיכות הפתרון לא ירדה משמעותית (הגרף כמעט אופקי עד w=0.7). עם זאת, זמן הפתרון עלה באופן דרסטי מאחר וכעט אנו פותרים בעיית A\* פנימית לכל מצב. לפיכך, למרות שהיוריסטיקה RelaxedDeliveriesHeuristic מיודעת יותר, שימוש ב- MSTAirDistHeuristic עם משקל w=0.6 מהווה פשרה טובה יותר במקרה זה, אם איננו נדרשים לספק את הפתרון האופטימלי.

**פרק שני**

1. נוכיח ש. נפריד למקרים:

במידה וh מוגדרת אז נקבל לפי הגדרת שמתקיים שנתון שהיא קבילה.

במידה ולא, אז נקבל ונתון ושוב נפריד לשני מקרים:

במידה וs היא צומת מטרה אז .

במידה וs אינה צומת מטרה אז לכל הפחות יהיה צריך לעבור על קשת אחת בשביל צומת מטרה, ולכן כאשר זה המחיר האופטימלי.

1. מכיוון שהתשובה של סעיף ג' בהכרח תופסת גם עבור סעיף זה אז נשתמש באותו אלגוריתם בדיוק.

ובכל זאת נכתוב אלגוריתם:

לצומת e שאנחנו רוצים לחשב לה יוריסטיקה:

1. אם isgoal(e)=True אז נחזיר 0. אחרת:
2. בודקים אם h מוגדרת על צומת זה באמצעות הפרדיקט.
3. אם כן- אז נגדיר את e להיות h(e).
4. אחרת:
5. ניצור מערך בגודל b שהוא מקדם הסיעוף:

לכל צומת s בsucc של e:

אם h מוגדרת על s אז נכניס למקום הבא במערך את .

אם לא אז נכניס כאשר זה מחיר הקשת בין s לe.

1. נחזיר את המינימום של המערך.

הנימוק יהיה בסעיף ג' שכן מדובר באותו אלגוריתם.

1. נשתמש באותו אלגוריתם בדיוק.

הוכחת קבילות(נקרא ליוריסטיקה שלנו h’):

במידה וh מוגדרת אז נקבל לפי הגדרת שמתקיים שנתון שהיא קבילה.

במידה וs היא צומת מטרה אז ולכן קבילה מההגדרה.

במידה ואין לצומת שלנו e שום שכן שהוא צומת מטרה אז בהכרח חייבים לעבור על אחת מהקשתות שיוצאות מהצומת e כדי להגיע לצומת מטרה. לכן לכל s שהוא מתקיים:

*הסבר לאי השוויון האחרון: בהכרח לכל s ואנחנו לוקחים מינימום שלהם.*

*השתמשנו בכך שh קבילה.*

*הוכחת מיודעות:*

*אם היוריסטיקה מוגדרת על צומת אז מתקיים שוויון . אחרת*

*כי משקל כל צלע הוא לפחות דלתא.*

1. *כן! קיים אלגוריתם כזה! והוא:*

*נריץ את A\* עם היוריסטיקה מסעיף ב'. מכיוון שהוכחנו שהיא מיודעת יותר מאשר אז הרצת A\* עם יוריסטיקה זו חסומה מלמעלה על ידי הרצת A\* עם . זה לפי משפט שלמדנו בהרצאה.*

*כלומר זמן הריצה של האלגוריתם שלנו קטן או שווה מאשר של A\* עם . לכן רק נותר למצוא דוגמה שבה הפתרון שלנו טוב יותר וזה יעיל.*

והנה דוגמה כזאת:

A

B

E

D

C

F

G

2

2

2

2

1

2

נתון שסדר היורשים הוא משמאל לימין וקודקוד המטרה הוא G.

האלגוריתם A\* עם יעבור על הקודקודים בסדר הבא:

בעוד שהאלגוריתם שהצענו יעבור בסדר הבא: