מבוא לבינה מלאכותית

תרגיל בית 1

אורעד בר-אל
311288203

מבוא לבינה מלאכותית - 236501

תרגיל בית 1

מרחבי חיפוש

מטרות התרגיל

- נתמודד עם בעיות פרקטיות ותיאורטיות של חיפוש במרחבי מצבים.
- נתרגל את הנלמד בהרצאות ובתרגולים.
- נתנסה בתכנות ב-python לפתרון בעיות פרקטיות.

הנחיות כלליות

- תאריך הגשה **2/3** יום שבת, בשעה 23:59.(לשנות מועד ולפתוח פיאצה)
- את המטלה יש להגיש בזוגות בלבד.
- יש להגיש מטלות מוקלדות בלבד בעברית או באנגלית. פתרונות בכתב יד לא ייבדקו.
- ניתן לשלוח שאלות בנוגע לתרגיל בפיאצה בלבד.
- המתרגל האחראית על תרגיל: **שאדי דיאב** .
- בקשות דחיה מוצדקות (מילואים, אשפוז וכו') יש לשלוח למתרגל האחראי (**ספיר טובול**) בלבד.
- במהלך התרגיל ייתכן שנעלה עדכונים, למסמך הנ"ל – תפורסם הודעה בהתאם.
- העדכונים הינם מחייבים, ועליכם להתעדכן עד מועד הגשת התרגיל.
- שימו לב, התרגיל מהווה כ- 15% מהציון הסופי במקצוע ולכן העתקות תטופלנה בחומרה!
- ציון המטלה יורכב מהגורמים הבאים:
  - 65% - המסמך היבש.
  - 35% - הקוד המוגש.
- אנו יודעים שעבור חלקכם זו התנסות ראשונה בכתיבת קוד בפיתון ותרגיל זה מתוכנן בהתאם לכך.
- שימו לב שלא יענו שאלות בסגנון: ״איך מוצאים את עלות הפתרון שהוחזר?״ / ״איך ניגשים למפות הכבישים מתוך המימוש של הפונק׳ ההיא?״ / ״באיזה שדה שמור ה...?״ וכדומה.
- אנחנו רוצים לעודד אתכם לעיין בקוד ולמצוא פרטים אלו בכוחות עצמכם. הכרת סביבת העבודה שסיפקנו לכם והתמצאות בה הן למעשה חלק מהתרגיל.
- בתרגילי הבית בקורס הרצת הניסויים עשויה לקחת זמן רב. לכן מומלץ מאוד להימנע מדחיית העבודה על התרגיל ו/או כתיבת הדו״ח לרגע האחרון. לא תינתנּה דחיות על רקע זה.
- מסמך זה כתוב בלשון זכר מטעמי נוחות בלבד, אך מתייחס לנשים וגברים כאחד.

אנחנו קשובים לפניות שלכם במהלך התרגיל ומעדכנים את המסמך הזה בהתאם. גרסאות עדכניות של המסמך יועלו לאתר. הבהרות ועדכונים שנוספים אחרי הפרסום הראשוני יסומנו כאן בצהוב. בנוסף, לכל עדכון יהיה מספר גרסה כדי שתוכלו לעקוב. ייתכן שתפורסמנה גרסאות רבות – אל תיבהלו מכך. השינויים בכל גרסה יכולים להיות קטנים.

הנחיות לחלק היבש

1. ככלל אצבע, בהינתן שאלה ראשית ספקו את התשובה המיידית ולאחר מכן תרחיבו ותסבירו. למשל, אם שואלים מה סיבוכיות הזמן של אלגוריתם BFS תשובה תהיה ״$O(b^d)$, מכיוון שבקרה הכי גרוע נאחסן את כל עץ החיפוש של הבעיה בCLOSE״.

הנחיות לחלק הרטוב

1. אנו מעודדים אתכם לעבור על הקבצים המצורפים ולהבין כיצד הסביבה בנויה ובאילו פונקציות תוכלו להשתמש במימוש שלכם.
2. הקוד שלכם ייבדק בקפדנות על ידי טסטים. הטסטים יבדקו את הפתרונות המוחזרים על ידי האלגוריתמים שלכם אל מול המימוש שלנו על פני בעיות שונות. אנו מצפים ממכם (אלא אם צוין אחרת)

להחזיר את אותם ערכים בדיוק. אנחנו נבדוק את המסלול המוחזר, מספר הצמתים שפתחו ואת עלות הפתרון המוחזר. הטסטים יהיו מוגבלים בזמן אך תקבלו זמן גדול מאוד לכל טסט.

3. ספקו קוד ברור ונקי הניתן לבדיקה ידנית.

מבוא ורקע

התרגיל מתפרש על פני מסמך זה והמחברת המצורפת. מומלץ לענות על השאלות לפי הסדר במסמך זה.

במטלה זו נעסוק בהפעלת אלגוריתמי חיפוש על מרחבי מצבים לבעיות ניווט. מומלץ לחזור על שקפי ההרצאות והתרגולים הרלוונטיים לפני תחילת העבודה על התרגיל.

סיפור מסגרת

לקאקרוטו וגוהאן יש 5 כדורי דרקון וחסר להם שני כדורים, והם ממש צריכים אותם כדי להזמין הדרקון שן-ראן ולבקש ממנו להחזיר את החברים שלהם לחיים, לכן הם הלכו לכוכב לכת נאמיק כדי לחפש כדורי הדרקון, קאקרוטו הציע שיחפשו על הכדור דרך ה ג׳׳.פי.אס שלהם אבל גוהאן מסביר לקאקרוטו שיש לו חברים שלוקחים הסמסטר את קורס ״מבוא לבינה מלאכותית״. גוהאן מבקש ממכם לעזור לו לתכנן את המסלול הטוב ביותר כדי לאסוף כדורי הדרקון ולהגיע לקאקרוטו שמחקה לו.

שאלה 1 – מבוא (8 נק׳):

השאלות בחלק זה מתבססות על הלוח "8x8" שמופיע במחברת אלא אם נכתב אחרת:

```
SFFFFFFF
FFFFFTAL
TFFHFFTF
FFFFFHTF
FAFHFFFF
FHHFFFHF
DFTFHDTL
FLFHFFFG
```

1. **רטוב**: עברו על המחברת עד שאתם מגיעים לחלק של BFS-G ועצרו שם.
2. יבש (1 נק׳): תחילה נרצה להגדיר את מרחב החיפוש כפי שנלמד בתרגול. הגדר את (S, O, I, G) עבור סביבת כדורי הדרקון. כאשר S זה מרחב המצבים, O, זה מרחב האופרטורים, I, זה המצב ההתחלתי וG הוא קבוצת מצבי המטרה. מה גודל מרחב המצבים S? הסבירו.

<u>תשובה:</u>

מרחב החיפוש הוא:

$$S=\underbrace{\{pos:0\le pos\le 64\}}_{position\ on\ board}\times\underbrace{\{d_1:d_1\in\{False,True\}\}\times\{d_2:d_2\in\{False,True\}\}}_{False\ if\ we\ didn't\ collect\ d_i\ and\ True\ if\ we\ did}$$
$$O=\{DOWN,RIGHT,UP,LEFT\}$$
$$I=(0,False,False)$$
$$G=\{(63,True,True)\}$$

גודל מרחב המצבים הוא:

$$|S|=\underbrace{(8*8)}_{position}*\underbrace{2}_{d_1}*\underbrace{2}_{d_2}=256$$

3. יבש (1 נק׳): מה תחזיר לנו הפונקציה Domain על אופרטור 2 (UP)?

<u>תשובה:</u>

$$Domain(UP)=\{(pos,d_1,d_2):state(pos)\neq H\}$$

הסבר: גם אם השחקן בשול העליון, ניתן להפעיל עליו UP שפשוט ישאיר אותו במקום. רק אם הוא הגיע לבור לא ניתן להפעיל עליו <u>אף אופרטור</u> ובפרט את UP.

4. יבש (1 נק׳): מה תחזיר לנו הפונקציה Succ על המצב ההתחלתי 0?

<u>תשובה:</u>

$$Succ\big((0,0),0,0\big)=\left\{\underbrace{(1,False,False)}_{o=RIGHT},\underbrace{(8,False,False)}_{o=DOWN},\underbrace{(0,False,False)}_{o=UP,LEFT}\right\}$$

5. יבש (1 נק׳): האם קיימים מעגלים במרחב החיפוש שלנו?

<u>תשובה:</u>

כן. למשל על ידי הפעלת DOWN על המצב ההתחלתי ואז הפעלת UP על המצב שהתקבל. נוצר מעגל בגודל 2.

6. יבש (1 נק׳): מה הוא מקדם הסיעוף בבעיה?

<u>תשובה:</u>

מקדם הסיעוף הוא **4**. זאת כיוון שלכל צומת יש לכל היותר 4 בנים – 1 או 0 לכל אופרטור.

7. יבש (1 נק׳): במקרה הגרוע ביותר, כמה פעולות ידרשו לסוכן כללי להגיע למצב הסופי?

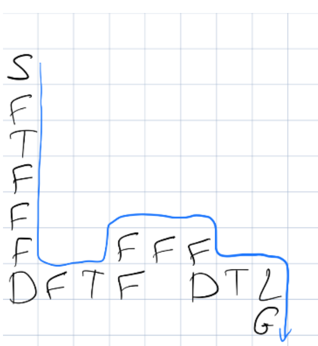
<u>תשובה:</u>

∞. כיוון שציינו בסעיף 5 כי ייתכנו מעגלים, הרי שמספר הפעולות לא חסום ויכול להיות אינסופי.

8. יבש (1 נק׳): במקרה הטוב ביותר, כמה פעולות ידרשו לסוכן כללי להגיע למצב הסופי?

<u>תשובה:</u>

במקרה הטוב ביותר **16**. זה יקרה עבור המקרה האופטימלי המתואר בציור:



9. יבש (1 נק׳): עבור לוח כללי, המסלול הקל ביותר הוא המסלול שמגיע למצב מטרה שהכי קרוב למצב ההתחלתי (במונחים של Manhattan distance)? אם כן, הוכיחו. אם לא, ספקו דוגמא נגדית.

<u>תשובה:</u>

לא נכון. להלן בציור דוגמה נגדית:

ניתן לראות כי $MD(G_1)>MD(G_2)$. ואולם נתבונן במסלול האופטימלי לכל אחד מצמתי המטרה:

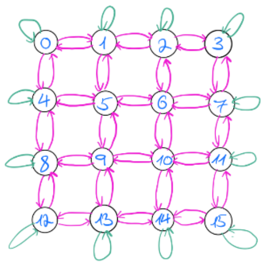
$$S\to F\to F\to D_1\to D_2\to G_1$$
$$S\to F\to F\to D_1\to D_2\to D_1\to F\to F\to S\to F\to G_2$$

ניתן לראות כי המסלול הקל ביותר הוא מ – S אל G_1. (מחירי הקשתות חיוביים) והמסלול אל G_1 הוא תת-מסלול ממש של המסלול אל G_2.

שאלה 2 – Breadth First Search-G (7 נק׳):

השאלות בחלק זה מתבססות על הלוח "8x8" שמופיע במחברת אלא אם נכתב אחרת.

1. **רטוב**: ממשו את אלג׳ BFS-G (על גרף) במחברת ע״פ ההנחיות המופיעות שם.
2. יבש (1 נק׳): מה צריך להיות התנאי על גרף החיפוש (לא בהכרח בבעיית כדורי הדרקון) כך שBFS על גרף ו-BFS על עץ ייצרו ויפתחו צמתים זהים באותו הסדר?
   תשובה:
   על גרף החיפוש להיות חסר מעגלים.
   הסבר: ב – BFS-G, בכל פעם שמגיעים לצומת, בודקים האם ראינו אותו כבר ורק אם לא אנו מכניסים אותו ל – Open ומפתחים אותו. כך, אם קיים מעגל בגרף, בהגיענו לצומת שכבר ביקרנו בו (זה שסוגר את המעגל), לא נפתח אותו. לעומת זאת, במקרה המקביל ב – BFS-T, נכניס אותו ל – Open ונפתח אותו.
3. יבש (2 נק׳): עבור הלוח "4x4" שמופיע במחברת, ציירו את גרף המצבים.
   תשובה:
   להלן הציור:



4. יבש (2 נק׳): נתון לוח בגודל NxN. הציעו דרך להשתמש באלגוריתם BFS-G כך שיחזיר פתרון אופטימלי (עלות מינימלית) והסבירו.
   - רמז: עליכם לספק פונקציה $T: G \to G'$ המקבלת את גרף המצבים G ויוצרת גרף חדש G' ובעזרתו למצוא את המסלול האופטימלי בגרף G.

תשובה:

נשתמש ביתרון של $BFS - G$ שיודע להחזיר מסלול אופטימלי אם הגרף בעל משקלים אחידים (ובפרט חסר משקלים). ניצור גרף שיגרום למחיר ההגעה לכל צומת ע"י קשת לבוא לידי ביטוי באורך המסלול החדש אליו.

נגדיר $T: G \to G'$ המספקת גרף המוגדר <u>ללא מחירים כלל</u> כדלהלן:

<u>צמתים:</u>

סוג ראשון: לכל צומת שמחירו $c(s) = 1$ או $c(s) = \infty$, יש בדיוק צומת אחד מקביל ב - G'.

סוג שני: לכל צומת שמחירו $1 < c(s) < \infty$, ניצור $c(s)$ צמתים מקבילים, עם אינדקסים $\{1, \dots, c(s)\}$.

<u>קשתות:</u>

כל הצמתים שנוצרו מהסוג השני יהיו מחוברים זה לזה בקשתות חד כיווניות $(s_i' \to s_{i+1}')$, כאשר לכל הצמתים שהם אינם $s_1', s_{c(s)}'$, כלומר אינם הראשון או האחרון, לא יהיו עוד קשתות מעבר לכך. את כל הקשתות שהיו לצומת המקביל s ב $G -$, אם הם קשתות נכנסות נחבר ל - s_1' ואם הן יוצאות נחבר ל - $s_{c(s)}'$. בכך נכריח מעבר בצמתים אלו במעבר בכל השרוך שאורכו כמשקל הצומת.

כל הצמתים שנוצרו מהסוג הראשון, אם מחירם $c(s) = 1$, הם יהיו מחוברים באופן דומה גם בגרף החדש (אל הצמתים מהסוג השני יהיו מחוברים בהתאם להסבר הפסקה הקודמת). אם מחירם $c(s) = \infty$ הם לא יהיו מחוברים כלל (כלומר נסיר את הקשתות הנכנסות שאולי היו להם בגרף המקורי). בכך נבטיח שאורך המסלול אליהם יהיה אינסוף.

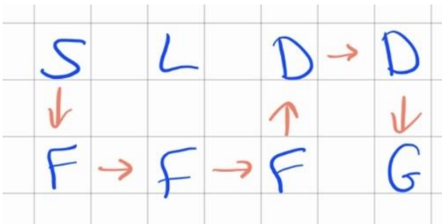
5. יבש (2 נק׳): נתון לוח בגודל NxN, ללא חורים, המכיל $N^2 - 2$ משבצות רגילות (F,T,A,L) מצב התחלתי בפינה השמאלית עליונה ומצב מטרה בפינה הימנית תחתונה. כמה צמתים <u>יפותחו</u> <u>וייווצרו</u> במהלך חיפוש BFS-G? הסבירו?
   תשובה:
   ייווצרו N^2 מצבים. זאת כיוון ש - $BFS - G$ גדל לכל הכיוונים כלפי חוץ וסדר יצירת הצמתים שלו הוא כמרחק הצומת מהמקור. כיוון שמצב המטרה הכי רחוק הוא ייווצר אחרון.
   יפותחו $N^2 - 2$ מצבים. הצמתים היחידים שלא יפותחו הם צומת המטרה (כאשר מגיעים לצומת מטרה מחזירים פתרון ולא ממשיכים לפתח) ושכנו מלמעלה (שכנו משמאל ייכנס ראשון לתור ולכן נפתח ראשון וכשנראה את צומת המטרה כאחד מבניו מחזיר פתרון).

שאלה 3 – Depth First Search-G (6 נק׳):

1. יבש (1 נק׳): עבור בעיית כדורי הדרקון עם לוח NxN, האם האלגוריתם שלם? האם הוא קביל?

תשובה:

האלגוריתם שלם. כיוון שהגרף סופי וכיוון שמדובר ב – DFS על <u>גרף</u> לא תיתכן ריצה במעגלים שכן בודקים צמתים שכבר ראינו.
וכפי שראינו בהרצאה, DFS במקרים כאלה DFS שלם.
האלגוריתם לא קביל. נראה דוגמה נגדית:
בדוגמה שלפנינו, אלגוריתם DFS יבחר את המסלול המתואר בציור.
משקלו: $10+10+10+1+1+1=33$.
זה כיוון שהוא מעמיק עד כמה שניתן (בהתאם לסדר הפיתוח שהוגדר כמובן) ולמעשה אינו מתייחס למחירים.
ואולם, קיים מסלול טוב יותר מזה: $S \to L \to D \to D \to D$.
משקלו: $1+1+1+1=4$. לכן הוא אינו קביל.



2. יבש (1 נק׳): האם אלגוריתם DFS (על עץ), עבור בעיית כדורי הדרקון על לוח NxN, היה מוצא פתרון כלשהו? אם כן, מה המסלול שיתקבל? אם לא, כיצד האלגוריתם היה פועל?

תשובה: לא.

כזכור, האלגוריתם מבצע חיפוש לעומק, כלומר הולך הכי לעומק שיכול עד שנתקע. לכן, האלגוריתם יילך מצומת המקור מטה (הוגדר כי הראשון בסדר הוא התחתון) עד שיגיע למצב ה - $[N-1,0]$. משם הוא ימשיך מטה בקשת עצמית אל אותו המצב שוב ושוב (כיוון שזה על עץ) ובכך ייתקע בלולאה אינסופית.

3. יבש (2 נק׳): נתון לוח בגודל NxN, ללא חורים, המכיל N^2-2 משבצות רגילות (F,T,A,L) מצב התחלתי בפינה השמאלית עליונה ומצב מטרה בפינה הימנית תחתונה (תניח כי שני כדורי הדרקון הם בפינה ימנית תחתונה) . כמה צמתים <u>יפותחו</u> <u>וייווצרו</u> במהלך חיפוש DFS-G? הסבירו?

תשובה:

יפותחו $2N-2$ וייווצרו $4N-5$.
המסלול שהאלגוריתם יבצע הוא הליכה למטה עד למצב ה - $[N-1,0]$. ומשם הליכה ימינה עד למצב ה - $[N-1,N-1]$. זה לפי עקרון הפעולה של $DFS-G$ וכיוון שבודקים קודם את התחתון לאחר מכן את הימני ואז את כל השאר. במסלול זה יש $2N-1$ מצבים. את כולם נפתח חוץ מאת האחרון שהוא מצב מטרה. לכן נפתח $2N-2$ מצבים.
הצמתים שייווצרו הם $2N-2$ המצבים שפותחו ובנוסף $N-1$ מצבים בהליכה למטה ו $N-2$ מצבים בהליכה ימינה. סה"כ:
$(2N-2)+(N-1)+(N-2)=4N-5$

4. יבש (2 נק׳): נתון לוח בגודל NxN, ללא חורים, המכיל N^2-2 משבצות רגילות (F,T,A,L) מצב התחלתי בפינה השמאלית עליונה ומצב מטרה בפינה הימנית תחתונה (תניח כי שני כדורי הדרקון הם בפינה ימנית תחתונה). כמה צמתים <u>יפותחו</u> <u>וייווצרו</u> במהלך חיפוש DFS-G backtracking? הסבירו?

תשובה:

יפותחו $2N-2$ וייווצרו $2N-1$.
המסלול שהאלגוריתם יבצע זהה למסלול של $DFS-G$ ולכן גם כאן נלך עד הסוף למטה ואז עד הסוף ימינה. גם כאן נפתח את כולם חוץ מאת האחרון ולכן נפתח $2N-2$ מצבים.
ואולם, ההבדל בין האלגוריתמים הוא שהנוכחי יוצר צמתים רק מיד לפני הביקור בהם ולכן ייווצרו נפתח $2N-1$ מצבים.

שאלה 4 – ID-DFS (6 נק'):

.1

a. (1 נק') האם האלגוריתם שלם? אם כן, הוכיחו. אם לא, ספקו דוגמה נגדית.

תשובה:

האלגוריתם שלם.

הוכחה: נראה כי אם קיים פתרון אז האלגוריתם מוצא אותו. נניח כי קיים פתרון. נסמן את עומק הפתרון ב - L. מאופן פעולת האלגוריתם, באיטרציה ה – L, יבצע האלגוריתם ריצת DFS חסומה עד גובה L ולכן יבקר בכל הצמתים שרחוקים מרחק L מהמקור ולכן ימצא את הפתרון.

b. (1 נק') נניח כי עלות כל פעולה היא 1, האם האלגוריתם קביל? אם כן, הוכיחו. אם לא, הסבירו.

תשובה:

האלגוריתם קביל.

הוכחה: נראה כי אם קיים פתרון אז האלגוריתם מוצא את הפתרון האופטימלי, כלומר שמרחקו מהמקור הוא הקטן ביותר (עלות כל פעולה היא 1). נניח כי קיים פתרון ונסמן את עומק הפתרון האופטימלי ב – L. כלומר, לא קיים פתרון שעומקו קטן ממש מ – L. מאופן פעולת האלגוריתם, הוא ירוץ באיטרציות על גובה החיפוש המקסימלי ולא ייעצר (כי לא ימצא פתרון) עד שיגיע לאיטרציה ה – L שבה ימצא את הפתרון האופטימלי (או פתרון אחר שעומקו גם כן L והוא גם אופטימלי).

.2 הניחו כי יש לנו ידע מקדים על חסם עליון למרחק למצב מטרה, נסמן D. בת (Beth) הציעה את האלגוריתם חיפוש הבא:

```
function ReverseDFS (problem, D):
        L ← D
        result ← failure
        While Not Interrupted:
                new_result ← DFS-L (problem, L)
                if new_result = failure:
                        break
                L ← L - 1
                result ← new_result
```

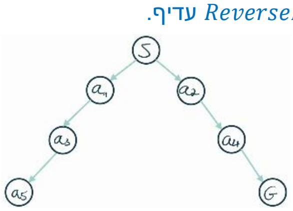
.3 בשאלות הבאות הניחו כי יש מספיק זמן לסיום האיטרציה הראשונה.

a. (1 נק') ספקו דוגמה בה ReverseDFS עדיף על ID-DFS ודוגמה בה ID-DFS עדיף על ReverseDFS. הדוגמאות יכולות להיות כלליות ולא בהכרח מסביבת התרגיל.

תשובה:

נתבונן בגרף המצבים המובא בציור. בגרף זה, המרחק של צומת המטרה (היחיד) מהמקור הוא $d = 3$. עבור גרף זה, נתבונן בשני מקרים כתלות ב – D:

מקרה א': $D = 3$. במקרה זה $ID - DFS$ ייצור וייפתח יותר מצתים כיוון שהוא יריץ DFS על $L = 1$ ואז על $L = 2$ ואז על $L = 3$. לעומת זאת $ReverseDFS$ יריץ רק על $L = 3$ ואז על $L = 2$. במקרה זה $ReverseDFS$ עדיף.



מקרה ב': $D = 6$. ריצת $ID - DFS$ כמובן תישאר זהה ואולם אלגוריתם $ReverseDFS$ יריץ DFS עם החסמים $L = 6, 5, 4, 3, 2$ ובכך ייצור ויפתח יותר צמתים (ריצות של $L = 6, 5, 4, 3$ למעשה שקולות). במקרה זה $ID - DFS$ עדיף.

b. (2 נק') הציעו כיצד ניתן לייעל את האלגוריתם. רמז: האם אתם יכולים לחשוב על צעד עדכון עדיף לL?

נשנה את האלגוריתם בצורה הבאה :

תשובה:

נדאג לכך שבצעד העדכון של L, נמנע ממנו למצוא פתרון באותו עומק שוב ושוב ע"י כך שנעדכנהו להיות עומק הפתרון שמצאנו באיטרציה הנוכחית פחות 1. כלומר בהינתן ועומק הפתרון שחזר בריצה מסוימת הוא d', נעדכן כך: $L \leftarrow d' - 1$.

שאלה 6 - UCS (4 נק׳):

השאלות בחלק זה מתבססות על הלוח "8x8" שמופיע במחברת אלא אם נכתב אחרת.

1. יבש (1 נק׳): עבור אילו בעיות חיפוש אלגוריתם UCS ואלגוריתם BFS יפעלו באותו האופן? הסבירו.

תשובה:

במקרים בהם משקל הצמתים בלוח אחידים.

כאשר מחיר הצמתים אחיד, מחיר ההגעה לכל אחד מהם הוא אחיד (כלומר מחיר כל קשת הוא אחיד). במקרים כאלו, מסלול יהיה קצר ביותר אם"ם הוא קל ביותר. לכן, כל צומת ש - BFS יבחר להתקדם אליו גם UCS יבחר כיוון ששניהם מסתמכים בבחירה זו על המרחק/המחיר (בהתאמה) שהם שקולים ועל הסדר המוגדר בבעייה ($D \rightarrow R \rightarrow U \rightarrow L$).

2. יבש (1 נק׳): האם בבעיית החיפוש שלנו, עבור לוח NxN, האלגוריתם הוא שלם? האם הוא קביל?

תשובה:

האלגוריתם שלם וקביל.

ראינו בתרגול כי במקרים בהם מחיר כל קשת חסום מלמטה ע"י קבוע חיובי האלגוריתם שלם וקביל. בבעיית החיפוש שלנו, מחיר כל קשת חסום מלמטה ע"י $\delta = 1$.

3. יבש (2 נק׳): שאדי טעה במימוש של אלגוריתם UCS ובטעות בדק בעת יצירת הצומת האם היא צומת מטרה במקום בפיתוח שלה. הביאו דוגמה לגרף חיפוש שעבורו שאדי יחזיר בכל זאת את המסלול הקל ביותר, ודוגמה לגרף חיפוש שעבורו שאדי לא יחזיר את המסלול הקל ביותר. עבור כל דוגמה הסבירו מה המסלול והעלות ש-UCS השגוי החזיר, ומה המסלול והעלות שהאלגוריתם הנכון היה מחזיר. נדגיש שגרף החיפוש לא בהכרח צריך לייצג את בעיית כדור הדרקון. אתם יכולים לתת דוגמה לגרף שמייצג בעיית חיפוש אחרת. הגרף צריך להכיל קשתות מכוונות ואת העלות של כל קשת.

תשובה:

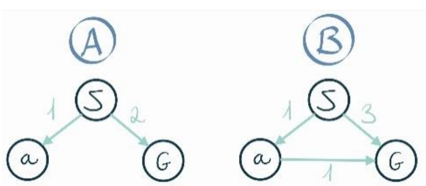
נתבונן בשני הגרפים המובאים בציור:

<u>גרף A:</u> בגרף זה שאדי יחזיר את המסלול הקל ביותר. שני האלגוריתמים יחזירו את המסלול $S \rightarrow G$ ואת העלות 2.

<u>גרף B:</u> בגרף זה שאדי לא יחזיר את המסלול הקל ביותר.

המסלול שהאלגוריתם של שאדי יחזיר הוא $S \rightarrow G$ עם עלות 3 כיוון שמיד עם יצירת G הוא יבדוק האם היא מטרה ולא ימתין לפיתוח שלה.

לעומת זאת, האלגוריתם הנכון יחזיר את המסלול $S \rightarrow a \rightarrow G$ ואת העלות 2. זאת כיוון שלפני שהוא בודק את G הוא מפתח את a (הוא קודם בערמת המינימום) ומעדכן את G ורק אז מפתח את G ובודק אותו.



שאלה 7 - יוריסטיקות (8 נק׳):

יהי מרחב חיפוש (S,O,I,G) , נסתכל על בעיית הניווט לכדור דרקון יחיד. המטרה היא למצוא מסלול זול ביותר מהמוצא I ליעד יחיד G .
פונק׳ העלות מוגדרת כאורך הכביש המחבר בין שתי נקודות, יהיה חסם תחתון $\delta > 0$ על אורך הכבישים .ניתן להניח כי העולם שטוח .
מלבד זאת, לא ניתן להניח דבר נוסף על מרחב החיפוש.

הגדרה : יוריסטיקה h היא ε-קבילה אם קיים $\varepsilon \geq 1$ כך שלכל מצב $s \in S$ מתקיים $h(s) \leq \varepsilon \times h^*(s)$.
נזכיר כי $h^*(s)$ הינה פונקציית המחיר המסלול האופטימאלי מ-s לצומת היעד .

עבור כל אחת מהיוריסטיקות הבאות קבעו האם קיים $\varepsilon \geq 1$ כך שהיוריסטיקה תהיה ε-קבילה . אם כן מצאו את ה-ε ההדוק ביותר המקיים את זאת. נמקו היטב .

1. יבש (1 נק׳): מרחק מנהטן : $h_{MD}(p) = |P-G|_1 = |G_x - P_x| + |G_y - P_y|$

תשובה: עבור $\varepsilon = \sqrt{2}$, היא ε-קבילה.

נימוק:
נרצה להסתכל על המקרה בו המסלול הקצר ביותר שקיים הוא בעל העלות הנמוכה ביותר שאפשרית. המסלול הקצר ביותר שייתכן בין צומת x כלשהו לבין G הוא דרך האלכסון, כלומר דרך קשת שמחברת בינו לבין היעד, אם קיימת (אי-שוויון המשולש לסכום). כלומר, העלות הנמוכה ביותר האפשרית היא המרחק אוקלידי. נדרוש זאת על ϵ:

$$\forall p: h(p) \leq \varepsilon \times h^*(p) \underset{\text{as written above}}{\Longleftrightarrow} h_{MD}(p) = \left|\underbrace{a}_{d_x} + \underbrace{b}_{d_y}\right| \leq \epsilon * \sqrt{a^2+b^2} \underset{\substack{\text{both sides positive} \\ \text{raise in the power of 2}}}{\Longleftrightarrow}$$

$$a^2 + 2ab + b^2 \leq \epsilon^2 * (a^2+b^2) \Leftrightarrow$$

$$2ab \leq (\epsilon^2-1)a^2 + (\epsilon^2-1)b^2 \underset{\epsilon^2-1 \stackrel{\text{def}}{=} t}{\Longleftrightarrow}$$

$$2ab \leq t(a^2+b^2) \Leftrightarrow 0 \leq ta^2 - 2ab + tb^2 \Leftrightarrow 0 \leq a^2 - 2ab + b^2 + (t-1)(a^2+b^2) \Leftrightarrow$$

$$0 \leq \underbrace{(a-b)^2}_{\geq 0} + (t-1)\underbrace{(a^2+b^2)}_{\geq 0}$$

לכן, כדי שזה יתקיים לכל p, עלינו לדרוש כי $t - 1 \geq 0$ שזה אם"ם $t \geq 1$ $\stackrel{\text{def}}{=}$ $\varepsilon^2 - 1$ שזה אם"ם $\varepsilon \geq \sqrt{2}$.

2. יבש (1 נק׳): $h(p) = |P-G| = \min\{G_x - P_x \ , \ G_y - P_y\}$

תשובה: עבור $\varepsilon = 1$, היא ε-קבילה.

נימוק:
נשווה ל - $|P-G|_2$ $h^*(p) \geq |P-G|_2$ (כפי שהוסבר לעיל). כיוון ש - $|G_x - P_x|, |G_y - P_y| \leq |G-P|_2$ (במשולש י"ז היתר הוא הגדול ביותר) אזי גם $\min\{|G_x - P_x|, |G_y - P_y|\} \leq |G-P|_2 \leq h^*(p) = \epsilon * h^*(p)$ לכל p.

3. יבש (1 נק׳): L^3 : $h(p) = |P-G|_3 = \sqrt[3]{|G_x - P_x|^3 + |G_y - P_y|^3}$

תשובה: עבור $\varepsilon = 1$, היא ε-קבילה.

נראה כי $L_2 \geq L_3$ ולכן לפי סעיף קודם מתקיים כי: $\forall p: h(p) \leq |P-G|_2 \leq \varepsilon \times h^*(p)$.

ראשית:

$$y > x \geq 0 \underset{x^2 \geq 0}{\Longrightarrow} yx^2 \geq xx^2 \underset{y^3 \geq 0}{\Longrightarrow} y^3yx^2 \geq y^3xx^3 \underset{=}{\Rightarrow} y^4x^2 \geq y^3x^3 \underset{3>2}{\Longrightarrow} 3y^4x^2 \geq 2y^3x^3$$

נניח בה"כ $a \geq b$. מתקיים:

$$a \geq b \geq 0 \Rightarrow 3a^4b^2 \geq 2a^3b^3 \Rightarrow 3a^4b^2 + \underbrace{3a^2b^4}_{\geq 0} \geq 2a^3b^3$$

כעת:

$$\forall a,b \geq 0: \quad 3a^4b^2 + 3a^2b^4 \geq 2a^3b^3 \underset{+a^6+b^6}{\Longrightarrow}$$

$$a^6 + 3a^4b^2 + 3a^2b^4 + b^6 \geq a^6 + 2a^3b^3 + b^6 \underset{formula}{\Longrightarrow}$$

$$(a^2+b^2)^3 \geq (a^3+b^3)^2 \Rightarrow \sqrt[6]{(a^2+b^2)^3} \geq \sqrt[6]{(a^3+b^3)^2} \Rightarrow \sqrt{(a^2+b^2)} \geq \sqrt[3]{a^3+b^3}$$

4. יבש (1 נק'): נתונות יוריסטיקות h_1 , h_2 שהן ε_1 , ε_2 קבילות בהתאמה וכי ε_1 , ε_2 הם האפסילונים ההדוקים ביותר.
הראו כי $h_3 = h_1 + h_2$ היא ε_3 – קבילה , מצאו את ε_3 ההדוק ביותר והוכיחו .

תשובה: $\varepsilon_3 = \varepsilon_1 + \varepsilon_2$. נוכיח:
לפי הנתון:
$\forall p: h_1(p) \le \varepsilon_1 \times h^*(p),\ h_2(p) \le \varepsilon_2 \times h^*(p)$.

לכן:
$$\forall p: h_3(p) \stackrel{\text{def}}{=} h_1(p) + h_2(p) \le \varepsilon_1 \times h^*(p) + \varepsilon_2 \times h^*(p) = (\varepsilon_1 + \varepsilon_2) \times h^*(p)$$

לכן h_3 היא ε_3-קבילה. נראה כעת כי הוא ההדוק היותר.
נניח בשלילה שהוא לא ההדוק ביותר. כלומר, קיים $1 \le \varepsilon' < \varepsilon_3$ המקיים זאת ("הדוק יותר"), אזי:
$$\forall p: \quad h_1(p) + h_2(p) \le \varepsilon' \times h^*(p)$$
נגדיר $\varepsilon_1' \stackrel{\text{def}}{=} \varepsilon' - \varepsilon_2$. נראה כי הדוק יותר עבור h_1 ובסתירה לנתון:

$$\forall p: \quad h_1(p) = h_3(p) - h_2(p) \le \varepsilon' * h^*(p) - \varepsilon_2 * h^*(p) = (\varepsilon' - \varepsilon_2) h^*(p) = \varepsilon_1' * h^*(p)$$
סתירה. לכן ε_3 הוא ההדוק ביותר.

נגדיר יוריסטיקה חדשה :

- D היא קבוצת כדורי הדרקון , $D = \{d1, d2\}$.

$$h_{MSAP}(s) = \min\{h_{Manhatan}(s,g) | g \in G \cup D\}$$

הערה: בנוסחת המרחק מתייחסים למיקום של צומת.
שימו לב שבמקרה זה אנחנו לוקחים את המינימום על פני כל צמתי היעד.

5. יבש (1 נק'): האם היוריסטיקה h_{MSAP} קבילה על כל לוח? אם כן הסבר, אם לא הבא קודמה נגדית.

תשובה: כן.
כיוון שהתנועה היא בצעדים אופקיים/אנכיים ומחיר כל צעד הוא לפחות 1, לכל צומת p ולכל צומת g מתקיים:
$h^*(p) \ge h_{MD}(s,g)$.

לכן:
$$h^*(p) \ge \min\{h_{MD}(s,g) \mid g \in G\} \ge \min\{h_{MD}(s,g) \mid g \in D \cup G\}$$
כאשר נכונות הצעד האחרון נובעת מכך שכשלוקחים מינימום על קבוצה, אם מגדילים את הקבוצה זה לא יכול להגדיל את האיבר המינימלי. לכן:
$$\forall p: \ h_{MSAP}(p) \stackrel{\text{def}}{=} \min\{h_{MD}(s,g) \mid g \in D \cup G\} \le h^*(p)$$

6. יבש (1 נק'): האם היוריסטיקה h_{MSAP} עקבית על כל לוח? אם כן הסבר, אם לא הבא דוגמה נגדית.(לחשוב אם היא עקבית ולתקן בהתאם)

תשובה:
כן.
יהי p ויהי $p' \in succ(p)$.
ראשית נבחין כי ייתכנו 3 מצבים עבור הצעד $p \to p'$.
מקרה ראשון בו לא שיפרנו וערך היוריסטיקה נשאר זהה (יש כדור או מטרה אחרים שנותנים את אותו ערך יוריסטיקה במצב החדש). אז: $h(p) - h(p') = 0$.
מקרה שני בו הרענו. זה יכול לקרות כאשר לא התקרבנו אל עבר כדור/מטרה קרובים יותר. ואז: $h(p) - h(p') = -1$.
מקרה שלישי בו שיפרנו. התקרבנו אל אותו כדור/מטרה או אל עבר אחד אחר שנותן ערך יוריסטיקה טובה יותר. ואז:
$h(p) - h(p') = 1$. (הערה: לא ייתכן כי שיפרנו ביותר, כיוון שההתקדמות היא בצעד אחד כל פעם ואם נניח כי שיפרנו ביותר מ – 1 נגיע לסתירה לערך $h(p)$ שלמעשה יהיה קטן יותר.)
לכן:
$$h(p) - h(p') \le 1 \le cost(p,p')$$

נגדיר יוריסטיקה חדשה :

- D היא קבוצת כדורי הדרקון , $D = \{d1, d2\}$.

$$h_{new}(s) = \max\{h_{Manhatan}(s, g) | g \in G \cup D\}$$

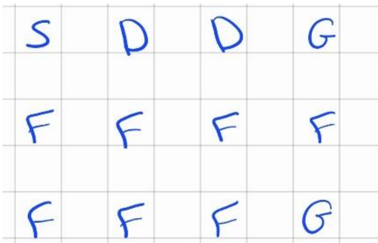
7. יבש (1 נק׳): האם היוריסטיקה h_{new} קבילה על כל לוח? אם כן הסבר, אם לא הבא קודמה נגדית.

תשובה:

היוריסטיקה אינה קבילה על כל לוח. נראה דוגמה נגדית ע"י ציור:

בדוגמה הנגדית שלפנינו: $h(s) = 5$, התקבל מהמרחק שלו אל צומת המטרה בפינה הימנית תחתונה.

ואולם: $h^*(s) = 1 + 1 + 1 = 3$. יוצא כי: $h(s) > h^*(s)$.



8. יבש (1 נק׳): האם היוריסטיקה h_{new} עקבית על כל לוח? אם כן הסבר, אם לא הבא דוגמה נגדית.

תשובה:

היא אינה עקבית על כל לוח.

דוגמה נגדית לכך היא הדוגמה הנגדית מהסעיף הקודם. היוריסטיקה אינה קבילה וזה גורר כי אינה עקבית.

שאלה 8 – Greedy Best First Search (3 נק׳):

השאלות בחלק זה מתבססות על הלוח "8x8" שמופיע במחברת אלא אם נכתב אחרת.

1. יבש (1 נק׳): האם האלגוריתם שלם? האם הוא קביל?

תשובה:

האלגוריתם שלם כיוון שמרחב המצבים סופי וללא מעגלים (כפי שראינו בהרצאה).

אך כפי שראינו בהרצאה, לא מובטח כי הוא קביל.

2. יבש (2 נק׳): תנו יתרון וחיסרון של אלגוריתם Greedy Best first Search לעומת Beam Search.

תשובה:

כפי שראינו בתרגול – טיב הפתרון אל מול זמן ומקום נמוכים.

יתרון של Greedy: כיוון שהוא לא זורק צמתים מ - $Open$, הוא לא יפספס צמתים שעלולים להיות על המסלול האופטימלי אל היעד.

חיסרון של Greedy: צורך יותר זמן ריצה ויותר מקום כיוון שהוא שומר יותר מ – K צמתים ב - $Open$ ברגע נתון.

שאלה 9 – W-A* (2 נק׳):

השאלות בחלק זה מתבססות על הלוח "8x8" שמופיע במחברת.

1. רטוב: ממשו את החלקים החסרים באלג׳ W-A* בקובץ ע״פ ההנחיות המופיעות שם. עליכם להשתמש ביוריסטיקה h_{MSAP}.

2. (יבש 2 נק׳) בהינתן $1 \le w_1 < w_2$, נסמן את המסלולים המחוזרים על ידי W-A* תחת הפורמולציה $f = g + w \cdot h$ בp_1, p_2 עבור w_1, w_2 בהתאמה. אזי $cost(p_1) < cost(p_2)$ עבור:

    a. יוריסטיקה קבילה h. אם כן הסבירו. אם לא, ספקו דוגמה נגדית.

    הטענה אינה נכונה. נראה דוגמה נגדית:

    ניקח את $h(p) \equiv 0$.

    היוריסטיקה h קבילה כיוון שלכל p מתקיים: $0 \le h(p) = 0 \le h^*(p)$.

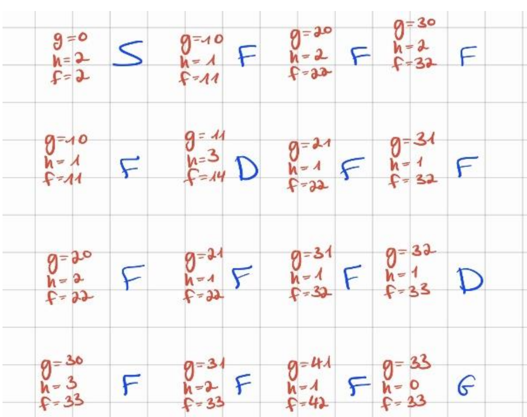
    ואולם, מתקיים כי $f_1 = g + w_1 * h \underbrace{=}_{h \equiv 0} g + w_2 * h = f_2$. לכן, $p_1 = p_2$ ולכן $cost(p_1) = cost(p_2)$.

b. יוריסטיקה כללית (לא בהכרח קבילה) h. אם כן הסבירו. אם לא, ספקו דוגמה נגדית.
הטענה אינה נכונה. נראה דוגמה נגדית:
ניקח את $h(p) \equiv 0$.
היוריסטיקה h היא קבילה ובפרט כללית.
ואולם, מתקיים כי $f_2 = g + w_2 * h \underbrace{=}_{h\equiv 0} g + w_1 * h = f_1$. לכן, $p_1 = p_2$ ולכן $cost(p_1) = cost(p_2)$.

שאלה 10 – *IDA (2 נק׳):

1. יבש (1 נק׳): ספקו יתרון וחסרון של *IDA ביחס ל*A. באילו מקרים הייתם מעדיפים להשתמש בכל אחד מהם?
<u>תשובה:</u>
יתרון של IDA^*: משתמש בחתימת זיכרון נמוכה יותר, כיוון שהוא מגביל את גודלו של $Open$.
חיסרון של IDA^*: עלול להגדיל את זמן הריצה, כיוון שהוא עלול להריץ את A^* מספר פעמים על מסלולים דומים באיטרציות השונות.
במקרה בו ניתן להניח כי צמתים שנמצאים על המסלול לפיתרון לא ייכנסו ל - $Open$ באיטרציותץ נמוכות (בגלל העומק שלהם מהמקור או בגלל ערכי f_value הגבוהים שיקבלו), כדאי להריץ את A^* ולא סתם "לבזבז" זמן על האיטרציות הנמוכות.
במקרים האחרים, נעדיף להריץ את IDA^*.

2. יבש (1 נק׳): ספק המחשה שלב אחר שלב של אלגוריתם *IDA על הלוח (4x4) שמופיע במחברת, המראה כיצד החיפוש מתקדם באמצעות העמקה איטרטיבית ?
<u>תשובה:</u>
האלגוריתם יחזיר את המסלול הבא: $15 \to 11 \to 10 \to 9 \to 5 \to 4 \to 0$.
האלגוריתם יבצע את איטרציות DFS החל מהצומת המקור, לפי ערכי $newLimit$, f_{limit} כפי שנלמד בתרגול, כאשר סדר ערכי f_{limit} שנגלה הוא: 2,11,14,22,32,33.
בכל פעם ישמור f_{max} חדש ויתחיל שוב את הריצה ויפתח יותר בנים עד שיגיע ל - $f_{limit} = 33$ (מגיע מ $G -$). כאשר זה יקרה, ריצת ה - DFS תרוץ ללא הגבלה ותגלה את המסלול הקצר ביותר שאוסף את שני הכדורים (מה שתיארנו לעיל).
להלן ציור של הגרף עם הערכים הסופיים הרלוונטיים:



שאלה 11 – A* epsilon (6 נק׳):

1. רטוב: ממשו את החלקים החסרים באלג׳ W-A* בקובץ ע״פ ההנחיות המופיעות שם. עליכם להשתמש ביוריסטיקה h_{MSAP}.

2. יבש (2 נק׳): תנו יתרון וחיסרון של A*-epsilon לעומת A*.

תשובה:
חיסרון של $A^* - epsilon$: עלות הפיתרון עשויה להיות גבוהה יותר – פתרון פחות איכותי (ראינו בהרצאה כי הוא מבטיח חסם מעט פחות טוב כתלות באפסילון).
יתרון של $A^* - epsilon$: הזמן שייקח לו להחזיר את הפתרון (ה"כמעט אופטימלי" לפי החסם שראינו) עשוי להיות קצר יותר.

3. יבש (3 נק׳): תנו הצעה ליוריסטיקה כדי לבחור את הצומת הבאה לפיתוח מתוך FOCAL. תארו את היוריסטיקה והציגו השוואה בין השימוש ביוריסטיקה זו לעומת השימוש ב-$g(v)$, מבחינת מספר פיתוחים, מסלול שנבחר ועלות המסלול שנבחר.

תשובה:
נציע את היוריסטיקה הבאה:

$$h_f(p) = Number\ of\ dargon\ balls\ left\ in\ state\ p, \quad h_f(p) \in \{0,1,2\}$$

$$\forall p \in goalStates: h_f(p) = 0$$

לאחר מימוש באמצעות היוריסטיקה הזו והשוואת התוצאות אל השימוש ביוריסטיקה המקורית:
כאשר השתמשנו ביוריסטיקה המקורית, עבור ערכי אפסילון שונים (גדולים וקטנים) קיבלנו עלות זהה (103) ומספר צמתים שפותחו זהה (81). לעומת זאת בשימוש ביוריסטיקה שהצענו, ניתן לראות כי ככל שאפסילון עולה, מספר הצמתים שפותחו עולה וכמו כן עלות הפתרון עולה משמעותית (למשל עבור אפסילון שווה למאה, קיבלנו עלות של 142 ומספר צמתים שפותחו 220).
לכן ניתן להניח כי היוריסטיקה המקורית עדיפה (לפחות עבור מפה זו).

4. יבש (1נק׳): אם נגדיר שאפסילון שווה לאינסוף איך תהיה ההתנהגות של האלגוריתם עם סביבת כדורי הדרקון.

תשובה:
עבור אפסילון שווה אינסוף, אין כל משמעות ל – open בתור תור אלא בתור set. בכל פעם, נגדיר בתור Focal להיות למעשה Open ונבחר לפי ערך ה - $g(v)$. לכן, ההתנהגות תהיה שקולה ל - UCS.

שאלה 12 – Benchmarking (2 נק׳):

בשאלה זאת נשווה בין אלגוריתמי חיפוש שונים על בעיות שונות. הריצו את החלק הרלוונטי במחברת ותיראו שנוצר קובץ csv. (ניתן לפתוח עם Excel).

1. רטוב: הריצו את החלק הרלוונטי במחברת ותיראו שנוצר קובץ csv. (ניתן לפתוח עם Excel).

2. יבש (2 נק׳): הסבירו את התוצאות. האם הן תואמות לציפיות שלכם? האם התוצאות היו משתנות עם יוריסטיקה יותר מיודעת? נתחו והסבירו את התוצאות במונחים של מספר פיתוחים, מסלול מוחזר ומחיר הפתרון. שימו לב שבסעיף זה אין תשובה נכונה או לא נכונה אבל נדרש ממכם לספק הסבר מפורט ומבוסס.

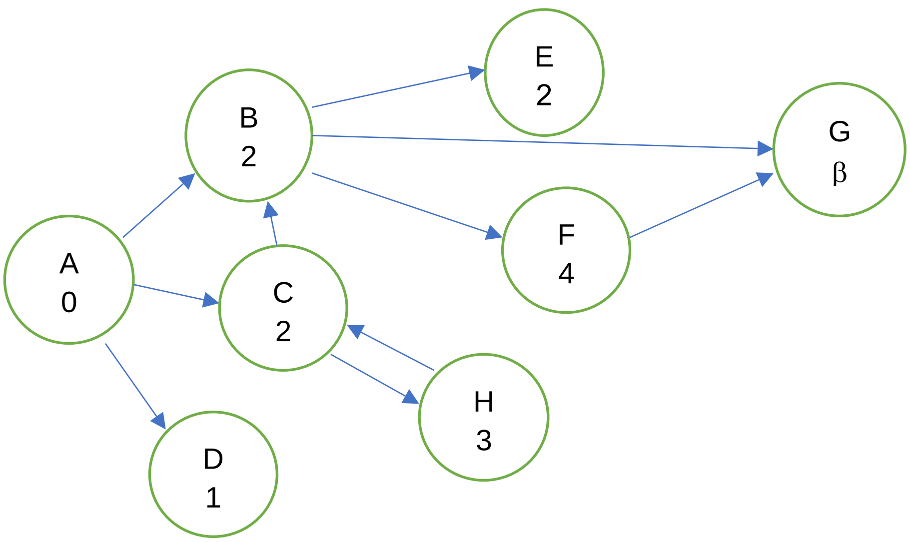
תשובה:
להלן התוצאות:

map	BFS-G cost	BFS-G expanded	WA* (0.5) cost	WA* (0.5) expanded	WA* (0.7) cost	WA* (0.7) expanded	WA* (0.9) cost	WA* (0.9) expanded
map 12x12	140	445	118	224	118	200	118	240
map 15x15	215	858	178	651	178	604	195	707
map 20x20	203	1045	188	684	188	587	188	1002

נפרט לפי 3 הפרמטרים שצויינו:
<u>מספר פיתוחים:</u> ציפינו כי האלגוריתמים המיודעים יפתחו פחות צמתים וככל שהאלגוריתם מיודע יותר (נותן יותר משקל ליוריסטיקה) כך גם יפתח פחות צמתים (כיוון שהיוריסטיקה מונעת "התפתחות לכל הכיוונים"). ואמנם, BFS הלא מיודע פיתח משמעותית יותר צמתים, אך דווקא מתן יותר משקל ליוריסטיקה לא הוריד את מספר הצמתים המפותחים ואף העלה (בקפיצה מ – 0.7 ל – 0.9).
<u>מסלול מוחזר:</u> ניתן לראות (במחברת אך לא כאן), כי עבור שתי המפות האחרונות, BFS החזיר מסלול קצר יותר (על אף שיותר "יקר") וזה כמצופה כיוון שהוא מוצא את המסלול הקצר ביותר.
<u>מחיר הפתרון:</u> ניתן לראות כי מחיר הפתרון של BFS גדול מאשר זה של האלגוריתמים האחרים וזאת כפי שציפינו כיוון ש BFS לא מתחשב במשקלים וכי הוא אופטימלי רק אם הגרף בעל מחירים אחידים. ואולם, ציפינו כי עבור WA*, ככל ש - h יגדל, כך הדיוק יקטן והמחיר יגדל, כיוון שהוא נהיה יותר גרידי וכבר לא מובטחת האופטימליות כמו עבור $0 \le h \le 0.5$. אך בפועל זה קרה רק פעם אחת (map15X15: $h = 0.7$ to $h = 0.9$).

שאלה 13 – Local Search (5 נק׳):

בהינתן מרחב המצבים הבא, כאשר a הינו המצב ההתחלתי, $U: S \to \mathbb{R}^+$ הינה פונקציית ערך והערך עבור כל מצב מצוין בצומת. המטרה שלנו היא למצוא מצב שממקסם את ערך U.



נשתמש באלגוריתם <u>Stochastic Hill Climbing</u>.
כמו כן ידוע כי $\boldsymbol{\beta > 3}$.

1. יבש (1 נק׳): מה ההסתברויות למעבר מהצב ההתחלתי לכל אחד מהמצבים b, c, d. רשמו את $p(d|a).p(b|a), p(c|a)$

<u>**תשובה:**</u>

$$\Delta U(b) = 2, \Delta U(c) = 2, \Delta U(d) = 1$$
$$\Rightarrow p(b|a) = p(c|a) = \frac{2}{5}, p(d|a) = \frac{1}{5}$$

2. יבש (1 נק׳): מה הוא מספר הצעדים המקסימלי שהאלגוריתם יכול לבצע? צעד מוגדר כמעבר בין מצבים.

<u>**תשובה:**</u>

נפריד למקרים:
אם $3 < \beta \le 4$: אין מעבר מ – F ל – G. לכן מספר הצעדים המקסי' הוא 2. למשל עבור המסלול: $A \to B \to F$.
אם $\beta > 4$: יש מעבר מ – F ל – G. לכן מספר הצעדים המקסי' הוא 3. זה עבור הארוך ביותר: $A \to B \to F \to G$.

3. יבש (1 נק׳): בהינתן שבצעד הראשון האלגוריתם עבר למצב c. האם האלגוריתם יתכנס למקסימום הגלובלי?

<u>**תשובה:**</u>

לא.

הסבר: שהמקסימום הגלובלי מקיים $M \geq U(F) = 4$. כיוון שהאלגוריתם לא עובר ממצב למצב שאינו משפר (גם אם יש להם אותו ערך U), בהכרח הוא יתכנס ל M – $U(H) = 3 < M$ ולכן לא יתכנס למקסימום הגלובלי.

4. יבש (1 נק׳): מה ההסתברות שהאלגוריתם יתכנס לפתרון לא אופטימלי (שאינו מקסימום גלובלי)?
יש שני מקרים :
תשובה:
ראשית, מתקיים:
$$p(A \to B \to F) = \underbrace{\frac{2}{5}}_{A \to B} * \underbrace{\frac{4-2}{(\beta-2)(4-2)+(2-2)}}_{B \to F} = \frac{4}{5\beta}.$$
בנוסף, אין הליכה ל E – כיוון שהוא לא משפר את B.
נפריד למקרים לפי β:
אם $3 < \beta \leq 4$: האופטימלי הוא F. וברגע שהגענו אליו אין ממנו הליכה ל G –. לכן:
$$p(U_{final} < M) = 1 - p(F) = 1 - \frac{4}{5\beta}.$$
אם $\beta > 4$: האופטימלי הוא G. לכן, ברגע שהגענו אל B, בוודאות נגיע ל G –, או ישירות אליו או דרך F. לכן:
$$p(U_{final} < M) = p(A \to B) = \frac{2}{5}.$$

5. יבש (1 נק׳): עבור אילו ערכים של β ההסתברות להגיע מהמצב ההתחלתי למקסימום הגלובלי תוך <u>בדיוק 3 צעדים</u> גדול מ - $\frac{1}{5}$?
תשובה:
מסלול כזה יכול להתקיים אם"ם $\beta > 4$ וגם המסלול הוא $A \to B \to F \to G$. בסעיף הקודם חישבנו את ההסתברות שלו:
$$p(A \to B \to F \to G) = P(A \to B \to F) = \frac{2(\beta-2)}{5\beta}.$$

לכן:
$$\frac{1}{5} < \frac{4}{5\beta} \underset{\beta>0}{\Longleftrightarrow} \beta < 4$$
<u>לכן, לא קיימים ערכים כאלה עבור β.</u>

הוראות הגשה:
עליכם להגיש קובץ יחד בשם AI1_<id1>_<id2>.zip (בלי הסוגריים המשולשים) המכיל:
1. קובץ בשם AI1_<id1>_<id2>.pdf שמכיל את התשובות לחלק היבש.
2. קובץ בשם Algorithms.py המכיל את המימוש לאלגוריתמי החיפוש.