בס"ד

בינה מלאכותית פתרון תרגיל 4

1. במשחק הפקמן החדש ישנו פקמן הנמצא במבוך בגודל 10X10 משבצות, המכיל קירות חוסמים **אך ללא אויבים וסכנות וללא נקודות על מאכלים**. בתחילת המשחק, הפקמן נמצא במקום רנדומאלי על הלוח. הפקמן צריך להגיע למשבצת [5,6]. בכל שלב הפעולות האפשריות לפקמן הן ללכת צעד אחד לכיוון North, South, East, West, Stop. בהנחה שאין קיר חוסם בכיוון זה.

עלינו למצוא תוכנית עבודה בה יגיע הפקמן ליעד במספר הצעדים הנמוך ביותר.

א. [20 נקודות] ייצגו את הבעיה כבעיית חיפוש **<u>באופן פורמאלי</u>** תוך התייחסות ל: מהי הגדרת מצב כללי, מהי פונקציית העוקב, מהו מבחן המטרה, מהי עלות צעד.

ב. [10 נקודות] הצע/י פונקציה היוריסטית המעריכה את מספר הצעדים עד למצב הפתרון ממצב כלשהו במשחק.

ג. [10] האם הפונקציה שהצעת היא קבילה?(אופטימית) נמקו בקצרה.

ד. [10נקודות] האם הפונקציה שהצעת היא מונוטונית? נמקו בקצרה.

א. ייצג/י את הבעיה כבעיית חיפוש **<u>באופן פורמאלי</u>**

<u>מצב כללי</u> הינו מטריצה 10 על 10 המתארת את מיקום הלוחות ומיקום הפקמן שייצוג ע"י נקודה [x,y].

<u>פונקציית העוקב</u>, היא הפעלת כל אחת מהפעולות האפשריות על פקמן, בהנחה שאין קיר חוסם באותה משבצת. כלומר

S([x,y])= [a,b],where:

1. [a,b] in { [x,y],
   [x,y+1] if y<10,
   [x, y-1] if y>1,
   [x+1,y] if x<10,
   [x-1, y] if x>1} and there is no wall between [x,y] and [a,b].

<u>מבחן המטרה</u> הפקמן הגיע ליעד כלומר x=5 and y=6

<u>עלות כל צעד</u> – 1- עלות שווה לכל תזוזה .

ב. הצע/י פונקציה היוריסטית המעריכה את מספר הצעדים עד למצב הפתרון ממצב כלשהו במשחק. אם נגדיר את הבעיה עם הקלה – שאין קירות בתוך הלוח, הרי שהמרחק שיצטרך הפקמן לעבור יהיה מרחק מנהטן בין מיקומו לבין [6, 5] כלומר |y-6| + |x-5|

ג. הפונקציה קבילה כיוון שאם אין קירות במסלול, זה בדיוק מספר הצעדים שיבוצע כיוון שזה המרחק הקצר ביותר כשלא ניתן ללכת באלכסון. אם יש קירות בדרך צריך לעקוף אותם, ולכן ללכת מספר צעדים גדול יותר.

ד. פונקציה עקבית (consistent) גורמת שציון f של מצב יהיה קטן או שווה לציון של בנו.

אם מתקדמים צעד לקראת היעד – עלות הצעד אחד ומרחק המנהטן מתקצר באחד כלומר f של מצב ההורה ו fשל מצב הילד זהה. אם המצב הבא מתרחק מהיעד, בשל קיר, אזי מרחק המנהטן של המצב החדש גדל וגם נוסף ל g של צעד נוסף כך שה f של הילד גדול יותר מה f של ההורה. לכן הפונקציה שהגדרנו היא עקבית.

2. [50 נקודות] נתון מצב התחלתי של 8-puzzle הבא

2		3
1	8	4
7	6	5

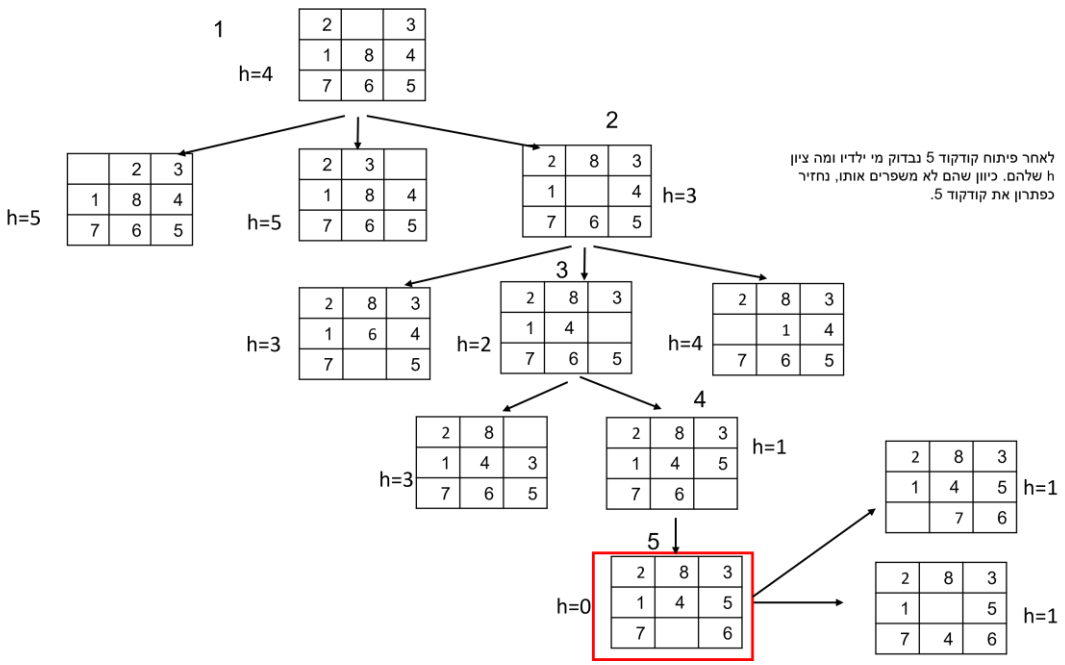
נרצה לפתור את הבעיה על ידי שימוש ב hill climbing כאשר הציון של קודקוד הוא מספר האריחים שאינם במקומם ומצב המטרה הוא:

2	8	3
1	4	5
7		6

<u>הראו את עץ הפיתוח שמייצר האלגוריתם (למרות שלא שומר את העץ)</u>

<u>חובה למספר הקודקודים שיפתחו</u> לפי סדר הפיתוח וחובה לכתוב את החישובים הנעשים לכל קודקוד.

<u>חובה לצייר המצבים בצורת מטריצה, כפי שמופיעים בשאלה</u>



שאלות לא להגשה:

נכון או לא נכון? חובה לנמק בקצרה

1. המצב ממנו מתחילים חיפוש Hill Climbing משפיע על היכולת של האלגוריתם להגיע לפתרון עם ערך אופטימלי.

- **נכון**. יש מצבים שהפעלת אלגוריתם חמדני שרק רוצה לשפר את הציון של המצב מגיע למקסימום גלובלי ויש מצבים שהפעלת אלגוריתם זה תוביל למקסימום מקומי, לכן יש השפעה של המצב ההתחלתי על ההצלחה להגיע לפתרון אופטימלי.

2. אם h פונקציה היוריסטית שתמיד (לכל קודקוד) גדולה מ 'h פונקציה היוריסטית אחרת לאותה בעיה, אז שימוש ב h יאפשר ל*A לפתח פחות קודקודים מאשר שימוש ב'h.

– **נכון .** בהנחה ששתי הפונקציות הן קבילות, עדיף להשתמש בפונקציה שמחזירה ערך גדול יותר, כי ככל שהפונקציה ההיוריסטית מחזירה ערך קרוב יותר (אך עדיין קטן) למרחק האמיתי כך יפתח *A פחות קודקודים ויתמקד במסלול האופטימלי עצמו.

הערה -אם הן לא קבילות, אין עדיפות של אחת על השניה- לא ניתן להבטיח ש*A יחזיר פתרון אופטימלי.

3. *A Iterative deepening צריך זמן עבודה כמו *A בדומה לכך ש iterative deepening צריך זמן עבודה דומה ל/BFS

- **לא נכון.** *A Iterative deepening יכול לפתח באופן חוזר קודקודים רבים, אם למשל יש ערכים שונים לכל הקודקודים, ולכן עלול לעבוד זמן רב יותר מ*A. זאת בניגוד ל Iterative deepening שמפתח באופן חוזר מספר קטן של

קודקודים (רמות עליונות בעץ) ולכן הגורם המשמעותי בזמן העבודה שלו הוא פיתוח הרמה של הפתרון אותה מפתח פעם אחת.

4. השימוש ב *A תמיד עדיף על פני שימוש בחיפוש מקומי לפתרון בעיה.
   - לא נכון. אם לא צריך מסלול לפתרון, אז חיפוש מקומי יכול למצוא פתרון כשל*A יקח הרבה מאד זמן להגיע לפתרון.
   - אם יש צורך במסלול לפתרון אז *A עדיף , מה דגם שהוא מתחייב למצוא פתרון בעל עלות אופטימלאלית וחיפוש מקומי לא יכול להתחייב על כך, אך לרוב ראחר מספר הרצות מוצא פתרון אופטימאלי.

5. טיפוס גבעה מתאים לכל בעיות החיפוש.
   - לא נכון . חיפוש גבעה הוא חיפוש מקומי שאינו מחזיר מסלול אל הפתרון ולכן לא מתאים לכל בעיית חיפוש.