**מבוא לבינה מלאכותית**

**סמסטר חורף תשפ"ה**

**מטלה 1**תאריך הגשה: 03.12.24 23:55

# הנחיות

* שאלות בנושא מטלה זו יש לשאול דרך המודל, בפורום "מטלה 1".
* הוראות להגשת המטלה מופיעים בסוף מסמך זה.
* הקבצים הנדרשים להרצת הקוד הינם:
  + heuristics.py
  + main.py
  + grid\_robot\_state.py
  + search.py
  + search\_node.py
* העבודה להגשה בזוגות בלבד אלא אם כן המגישים קיבלו אישור להגשה שאינה בזוגות.
* לפני שניגשים לממש את המטלה מומלץ לעיין רבות בהסברים וכן בקוד הקיים.
* פתרון המטלה שתגישו ייבדק מול שאר ההגשות על ידי תוכנת העתקות.
* **מי שימצא כי העתיק יכשל בקורס וכן יועבר לוועדת משמעת אוניברסיטאית**.
* הפרויקט נכתב וייבדק בשפת התכנות python.
* מסמך זה בנוי באופן הבא: תיאור המטלה, בעיית החיפוש, מרחב הבעיה, שאלות המטלה, פונקציות שיש להשלים והסבר על פונקציות קיימות.

# תיאור המטלה

סגל הקורס, מבוא לבינה מלאכותית, החליטו לפתוח סטרטאפ להחלפת מנורות בשם "יש לי רעיון". לשם כך, הם פיתחו רובוט שאוסף ומרכיב מדרגות כדי להגיע למנורה בתקרה. הרובוט זז בחדר ב-4 כיוונים מבלי להתקע במכשולים ויכול להרים מדרגות כאשר הוא מגיע למשבצת בן הן נמצאות. כמו כן, אם הרובוט מחזיק כבר מדרגות, הוא יכול להגיע למשבצת עם עוד מדרגות ולהרכיב את המדרגות ליחידה אחת (הסבר מפורט בהמשך). עזרו לסגל הקורס לכתוב אלגוריתם **\*A** המוצא את הפתרון האופטימלי בהינתן עלות של כל פעולה כמפורט בהמשך.

# בעיית חיפוש

בהינתן מיקום הרובוט, מפת החדר עם המכשולים, מיקום המנורה וגובהה ,מיקומי המדרגות וגובהם. עלינו למצוא רצף מצבים שבעזרתן הרובוט יגיע למיקום המנורה ויהיה שם מספר מדרגות כגובה המנורה. שימו לב: המדרגות צריכות להיות מחוברות כיחידה אחת כדי להחשיב את המצב כמצב סופי, כמו כן, אין חשיבות אם הרובוט מחזיק את המדרגות או שהן על הקרקע.

# מרחב הבעיה

* מפה - מערך דו מימדי של משצבות בגודל . חלק מהמשבצות פנויות וחלק חסומות על ידי מכשול (לא מדרגות).
* מיקום המנורה על המפה .
* גובה המנורה .
* מיקומי מדרגות על המפה .   
  מספר המדרגות קטן או שווה למקומות הפנויים במפה וקיימות מספיק מדרגות כדי להגיע למנורה.
* גובה של המדרגות *.   
  גובה של מדרגות במשבצות שונות יכול להיות שונה.*
* *מיקום הרובוט על המפה .*

מצב התחלתי: מפה שבה מצויין, מיקום המכשולים, מיקום המדרגות וגובהם. בנוסף לכך(לא על המפה), נתון מיקום ההתחלתי של הרובוט, מיקום המנורה וגובה המנורה. הרובוט לא מחזיק מדרגות בתחילת הבעיה.

על המפה –

0 מציין משבצת ריקה.

ערך גדול מ-0 מציין מדרגות בגובה הערך המצויין.

1-(מינוס אחד) מציין מכשול – לא ניתן לזוז למשבצת זו.

מצב סופי: מיקום הרובוט שווה למיקום המנורה וגם ישנה יחידה אחת של מדרגות בגובה המנורה במיקום המנורה.

אופרטורים:

הרמת מדרגות על ידי הרובוט- הרובוט יכול להרים (ולהחזיק) את המדרגות אם הוא נמצא באותה משבצת של המדרגות והוא לא מחזיק כבר מדרגות. ברגע שהרובוט הרים את המדרגות, המדרגות כבר לא נמצאות במשבצת זו אלא אם יניח אותן בחזרה. עלות פעולה זו הינה 1.

הנחת מדרגות על ידי הרובוט- הרובוט יכול להניח את המדרגות (במידה והוא כבר מחזיק אותן) על משבצת פנויה בלבד. אם יש כבר יש מדרגות במשבצת זו, הרובוט יכול לחבר את המדרגות שהוא מחזיק והמדרגות במשבצת ליחידה אחת. ברגע שהרובוט הניח את המדרגות, הן נשארות שם עד שהוא ירים ויזיז אותן שוב. עלות פעולה זו הינה 1.

חיבור מדרגות על ידי הרובוט- הרובוט יכול לחבר מדרגות ליחידה אחת במידה והוא נמצא במשבצת שבה יש מדרגות ובנוסף לכך הוא מחזיק מדרגות (נוספות). לאחר החיבור הרובוט נשאר באותו המיקום כאשר הוא מחזיק את היחידה החדשה. לא ניתן לחבר מדרגות שסכומן גדול מגובה המנורה. עלות פעולה זו הינה 1.

תזוזה של הרובוט למשבצת שהיא לא מכשול- הרובוט יכול לזוז משבצת אחת ב-4 כיוונים (למעלה, למטה, ימינה ושמאלה) בתנאי שהמשבצת השכנה אינה מכשול או מחוץ לגבולות המפה. עלות פעולה זו היא 1+ גובה המדרגות אותו הוא מחזיק בצעד הנתון. עבור צעד בלי החזקת מדרגות עלות הפעולה היא 1. לדוגמה: אם הרובוט זז ימינה ומחזיק יחידת מדגרות בגובה 3 עלות הפעולה תהיה 1+3=4.

A black rectangular object with white text

Description automatically generated

בדוגמה הזאת (משבצות בצבע שחור מהוות מכשול):

הרובוט ילך למדרגות בגובה 3 (8 צעדים בעלות 8\*1)

הרובוט ירים את המדרגות (בעלות של 1)

הרבוט יקח את המדרגות למיקום המדרגות בגובה 4 (5 צעדים בעלות 5\*4)

הרובוט יחבר את המדרגות ליחידה אחת בגובה 7 (בעלות של 1)

הרבוט יקח את המדרגות למיקום המנורה (4 צעדים בעלות 4\*8)

הרובוט יניח את המדרגות על הרצפה (בעלות של 1)

סה"כ עלות הפתרון: 8+1+20+1+32+1 =63

# שאלות המטלה

1. עזרו לסגל הקורס להשלים את הפונקציות החסרות (מצויינות בהמשך).
2. שנו את היוריסטיקה base heuristic כך שתחשב את מרחק מנהטן בין מיקום הרובוט והמנורה.
3. שנו את היוריסטיקה advanced\_heuristic כך שתחשב היוריסטיקה של מצב מסויים בבעיה. על היוריסטיקה לשפר את היוריסיטיקה מסעיף ב' (ימדד בזמן ריצה) ולשמור על אופטימליות הפתרון.

הזוג שהריץ את הטסטים של סעיף ג' בזמן הממוצע הקצר ביותר (ימדד בהרצת הטסטים במודל ואולי בטסטים נוספים רק כדי להכריע את הבונוס), יקבל **2 נקודות בונוס לציון הסופי**.

שימו לב שכל הטסטים צריכים לעבור כדי שהזמן יחשב.  
בנוסף, כדי למדוד את פונקציית היוריסטיקה בצורה מדוייקת יותר, הוספנו בטסטים מימוש של קובץ ה search ושם נקרא לפונקציות היוריסטיקה (והגדרת המצבים) שלכם כדי למדוד את זמן הריצה עם אותו מימוש של מבנה נתונים ולוגיקת חיפוש. גם כאן ינתנו **2 נקודות בונוס לציון הסופי.**

זוגות שרוצים להשתתף בבונוס צריכים לרשום את תעודות הזהות שלהם והזמן שקיבלו בהרצה [בקבוץ הזה](https://docs.google.com/spreadsheets/d/1yaRxoGDln3bC4kn_MCxMAC7J9nIfvCaQ6LvqcHDMvjM/edit?usp=sharing).

# הפונקציות שיש להשלים למימוש כל אחד מאלגוריתמי החיפוש

heuristics.py

base\_heuristic(\_grid\_robot\_state)

הפונקציה מקבלת אובייקט מסוג \_grid\_robot\_state ומחזירה את ערך היוריסטיקה עבור הקלט הנוכחי. (יש למחוק את השורה שמחזירה 0)

advanced\_heuristic(\_grid\_robot\_state)

הפונקציה מקבלת אובייקט מסוג \_grid\_robot\_state ומחזירה את ערך היוריסטיקה עבור הקלט הנוכחי. (יש למחוק את השורה שמחזירה 0)

grid\_robot\_state.py

get\_neighbors(self)

הפונקציה לא מקבלת קלט ומחזירה רשימה של סדורות (tuples) כך שבכל סדורה יש את אובייקט מסוג \_grid\_robot\_state המייצג את אחד השכנים ועלות המעבר אל אותו שכן.

[(grid\_robot\_state1, cost1), (grid\_robot\_state2, cost2)…]

is\_goal\_state(\_grid\_robot\_state)

פונקציה סטטית המקבלת מצב מסוג grid\_robot\_state ומחזירה האם המצב הינו מצב סופי.

search.py

create\_open\_set()

הפונקציה אינה מקבלת ערכים, ומחזירה מבני נתונים מתאים לאלגוריתם החיפוש.

create\_closed\_set()

הפונקציה אינה מקבלת ערכים, ומחזירה מבני נתונים מתאים לאלגוריתם החיפוש.

add\_to\_open(vn, open\_set)

הפונקציה מקבלת כקלט את מבני הנתונים open ומצב vn ומכניסה את vn ל-open.

open\_not\_empty(open\_set)

הפונקציה מקבלת כקלת את מבני הנתונים open ומחזירה אמת אם הוא לא ריק, אחרת, מחזירה שקר.

get\_best(open\_set)

הפונקציה מקבלת כקלט את מבני הנתונים ומחזירה את הקודקוד הטוב ביותר, על פי אלגוריתם החיפוש.

add\_to\_closed(vn, closed\_set)

הפונקציה מקבלת כקלט את מבני הנתונים closed ומצב vn ומכניסה את vn ל- closed.

duplicate\_in\_open(vn, open\_set)

הפונקציה מקבלת כקלט את מבני הנתונים open ומצב vn ומחזירה אמת אם קיים קודקוד ב open עם מצב זהה לvn.state וערך הg שלו קטן או שווה ל vn.g. אם vn.state לא קיים ב open או ערך הg של המצב בopen גדול מvn.g , הפונקציה תחזיר שקר.

duplicate\_in\_close(vn, closed\_set)

הפונקציה מקבלת כקלט את מבני הנתונים closed ומצב vn ומחזירה אמת אם קיים קודקוד ב closed עם מצב זהה לvn.state וערך הg שלו קטן או שווה ל vn.g. אם vn.state לא קיים בclosed או ערך הg של המצב בclosed גדול מvn.g , הפונקציה תחזיר שקר.

# מידע על פונקציות ממומשות

search\_node.py

get\_neighbors(self)

הפונקציה לא מקבלת קלט ומחזירה את כל המצבים השכנים של המצב הנוכחי.

search.py

print\_path(path)

הפונקציה מקבלת מערך של קודקודים מסוג search\_node ומדפיסה את המחרוזת המייצגת את המצב של מיקום הרובוט(ניתן להוסיף עוד מידע).

search(start\_state, heuristic, goal\_state)

הפונקציה מקבלת את המצב ההתחלתי, פונקצית היוריסטיקה ומצב סופי ומחזירה מערך של search\_node המייצגים את המסלול האופטימלי מהמצב ההתחלתי למצב הסופי. בתא הראשון במערך יהיה המצב ההתחלתי בתא הבא יהיה השכן שלו במסלול שהוחזר וכך הלאה עד המצב הסופי בתא האחרון.

# הגשת המטלה

* יש להגיש דרך מערכת המודל את ארבעת הקבצים הבאים:
  + heuristics.py
  + grid\_robot\_state.py
  + search.py
  + search\_node.py

הגשת המטלות תתבצע ישירות מול מערכת המודל בצורה אלקטרונית.

שימו לב: ישנה אפשרות להגיש את המטלה התכנותית מספר פעמים(אין הגבלת הגשה) ובכל הגשה לקבל חיווי, כלומר תקבלו באופן מיידי את הציון לביצוע התרגיל.

ניתן לראות את הפידבק להרצה (קומפילציה, מספר טסטים שעברו, שגיאות זמן ריצה וכו'...).

לאחר סיום ההרצה יתקבלו התוצאות. ישנם מספר טסטים הבודקים את הפתרון המוגש למטלה. אם התקבלה שגיאת קומפליציה יש עליכם להעלות קובץ חדש בכדי לקבל ציון לאחר התיקונים. בקבצים המקוריים שקיבלתם עם המטלה עלולות להיות שגיאות ריצה עבור פונקציות לא ממומשות. לא יבדקו עוד טסטים שישפיעו על צינוכם במטלה מעבר לטסטים בהגשה אך יכול להיות שיהיו עוד טסטים עבור הכרעה של נקודונות הבונוס.

בהצלחה!