

# מבוא לבינה מלאכותית

סמסטר חורף תשפ"ו

## מטלה 3 - Domain-Independent Planning

תאריך הגשה: 09/12/2025



### הנחיות

- שאלות בנושא מטלה זו יש לשאול דרך המודל, בפורום "מטלה 3".
- הוראות להגשת המטלה מופיעות בסוף מסמך זה.
- הקבצים הנדרשים להרצת הקוד הינם:
  - community\_garden.pddl
  - community\_garden\_problem\_v1.pddl
  - community\_garden\_problem\_v2.pddl
  - community\_garden\_problem\_v1\_solution.txt
  - community\_garden\_problem\_v2\_solution.txt
  - community\_garden\_problem\_v3.pddl
  - community\_garden\_problem\_v4.pddl
  - community\_garden\_problem\_v5.pddl
- העבודה להגשה בזוגות בלבד אלא אם כן המגישים קיבלו אישור מיוחד.
- 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, גרסה 3.10.
- מסמך זה בנוי באופן הבא: תיאור המטלה, בעיית תכנון, מרחב הבעיה, שאלות המטלה, הנחיות לביצוע השאלות, והסבר על הגשת המטלה.

## תיאור המטלה

השמש הציצה מעל האופק והטילה זוהר חמים על עיירת הרמוניויל. היום היה שבו חלקת האדמה המוזנחת מאחורי בניין העירייה תיהפך לגינת הקהילה התוססת של הרמוניה. שלושה מתנדבים מסורים, שלכל אחד מהם תפקיד חשוב, הגיעו נלהבים להתחיל.

כדי לעשות זאת עליכם לעזור למתנדבים (תושבי הרמוניויל) המסורים לאסוף את הכלים הדרושים לגינון ואז לשתול, להשקות ולטפח את גינת הקהילה החדשה!

כל זאת תעשו על ידי יצירת PDDL המתאר את היכולות והמשימות של המתנדבים, ולאחר מכן פתרון בעיות במיקומים שונים בעיירה כך שכל אזור יהפוך לחלק פורח ופעיל בגינה.

כחלק מהמטלה נתונים לכם ה-predicates הטיפוסים של האובייקטים בבעיות והגדרות הפעולות. עליכם להשלים את ה-preconditions וה-effects על מנת לאפשר לסוכן התכנון לייצר אוסף פעולות שיאפשרו למתנדבים לאסוף ציוד ולבסוף להפוך את שטח האדמה המוזנח לגינת קהילה שוקקת חיים. שימו לב שעירת הרמוניויל מיוצגת כ-grid, כאשר בכל תא יש בדיוק דבר אחד.

## מרחב הבעיה

### מוגדרים הטיפוסים הבאים:

gardeningTool location volunteer - object

tiller seeds wateringCan - gardeningTool

### בנוסף אנחנו מגדירים את הקבוע הבא:

gardenPlot - location

המגדיר את מיקום הגינה המיועד.

### מוגדרים הפרדיקטים הבאים:

1. (at ?o - object ?loc - location) - מציין שהאובייקט ?o נמצא במיקום ?loc.
2. (has-tiller ?v - volunteer ?t - tiller) - מציין שהמתנדב ?v, השיג את המחרשה ?t.
3. (has-seeds ?v - volunteer ?s - seeds) - מציין שהמתנדב ?v, השיג את הזרעים ?s.
4. (has-watering-can ?v - volunteer ?wc - wateringCan) - מציין שהמתנדב ?v, השיג את משפך ההשקיה ?wc.
5. (is-cultivator ?v - volunteer) - מציין שהמתנדב ?v מתמחה בעיבוד הקרקע, תפקידו להכין את האדמה לגידול ולשתילה.
6. (is-planter ?v - volunteer) - מציין שהמתנדב ?v מתמחה בשתילה, תפקידו לזרוע את הזרעים במקומם המתאים בגינה.
7. (is-waterer ?v - volunteer) - מציין שהמתנדב ?v מתמחה בהשקיה, תפקידו לדאוג שהצמחים יקבלו את כמות המים הדרושה להם לצמיחה.
8. (soil-is-tilled) - מציין שהקרקע עובדה והוכנה לשתילה.
9. (seeds-are-sown) - מציין שהזרעים נזרעו באדמה.
10. (garden-is-watered) - מציין שהגינה הושקתה.
11. (garden-is-thriving) - מציין שמטרת המשימה הושגה, כלומר הגינה פורחת ומשגשגת.

### עליכם להגדיר את ה- preconditions וה- effects לפעולות הבאות:

**move** – הפעולה מזיזה את המתנדב v מהמקום from למקום to.

- כדי שהפעולה תפעל המתנדב צריך להיות במקום from לפני ביצוע הפעולה.
- לאחר ביצוע הפעולה המתנדב עובר מלהיות ב- from ללהיות במיקום to.

**get-tiller** – הפעולה מאפשרת למתנדב לקחת מחרשה.

- **רק** מתנדבים שהם מוגדרים כמתמחים בעיבוד קרקע יכולים לבצע את הפעולה.
- המיקום של המתמחה **חייב** להיות זהה למיקום המחרשה אותה הוא רוצה לקחת.
- לאחר ביצוע הפעולה, המתמחה השיג את המחרשה והיא כבר לא נמצאת במיקום בו היא הייתה.

**באופן דומה** מוגדרות הפעולות get-seeds עבור מתנדב שמתמחה בשתילה תשיג זרעים, ו- get-

watering-can עבור מתנדב המתמחה בהשקיה תשיג משפך השקיה.

לאחר שכל אחד מהמתנדבים אוסף את הכלי שלו, הם צריכים להשתמש בכלים בגינה המיועדת.

**till-soil** – הפעולה חורשת את אדמת הגינה.

- ניתן לבצע את הפעולה **רק** אם המתנדב אכן מומחה בעיבוד קרקע **והוא אסף** את המחרשה.
- המיקום של המתנדב **חייב** להיות במיקום הגינה.
- לאחר חרישת האדמה, האדמה הופכת להיות חרושה (soil-is-tilled).

**באופן דומה** מתמחה בשתילה ישתול זרעים בגינה ומתמחה בהשקיה ישקה את הגינה.

**באופן שונה** קיימות תלויות בין הפעולות, הפעולה sow-seeds מחייבת בנוסף לתנאים הבסיסים שגם הקרקע תייה מוכנה לפני הפעלתה (soil-is-tilled), והפעלת water-garden מחייבת בנוסף לתנאים הבסיסים שגם יהיו זרעים באדמה (seeds-are-sown).

הפעולה האחרונה אותה יש להשלים הינה **celebrate-garden-opening**. בפעולה זו כלל המתנדבים מתכנסים יחד בחלקת הגינה כדי לחגוג את פתיחתה הרשמית.

- ניתן לבצע את הפעולה רק אם האדמה עובדה, הזרעים נזרעו, והגינה הושקתה.
- כל שלושת המתנדבים — המעבד, השותל והמשקה חייבים להיות נוכחים באותה חלקת גינה.
- לאחר ביצוע הפעולה מטרת המשימה הושגה והגינה פורחת ומשגשגת.

## הנחיות לביצוע העבודה

- עבור כל אופרטור יש להשלים את שתי הרשימות:

**preconditions -**

- הפרדיקטים הנדרשים עבור הפעלת אופרטור זה.

**effects -**

לפני פרדיקטים שמסירים יש לשים **not**.

מומלץ לעבוד עם vscode עם הפלאגין של PDDL של Jan Doljsi. זה מאפשר syntax highlights כמו גם אינפורמציה על שגיאות בכתיבת ה-domain.

### **:Domain independent planner**

מזורף לכם קובץ פייטון בשם uni\_planner.py, הוא ישים אתכם כ domain independent planner ואין לשנותו.

לפני השימוש בו עליכם לפתוח סביבת פייטון ובה לבצע את ההתקנה הבאה:

```
pip install unified-planning[engines]==1.2.0
```

לאחר מכן השימוש בו נעשה בעזרת הפקודה הבאה:

```
python uni_planner.py --domain_file <domain-path> --problem_file <problem-path>
```

שימו לב, כחלק מבדיקת התרגיל אתם תיבדקו גם על נכונות ה PDDL domain אותו אתם מייצרים וגם על איכות הפתרון שנוצר.

לטובת כך ישנם **שלושה** קבצי בעיה community\_garden\_problem\_v(3-5).pddl אשר עליכם לספק את הפתרונות לבעיות אלו על ידי הרצת ה-planner והוספת הפתרון כחלק מההגשה. בדיקת התרגיל תתבצע כולה ב-VPL וייתן לכם משוב על הפתרון אותו אתם מספקים.

# בהצלחה!