

# תרגיל בית 2

## מבוא

מטרת התרגיל היא להתנסות בפתרון בעיות בתנאי ידע חלקי.

עקרון המשחק נשאר דומה לזה שראיתם בתרגיל הראשון - עדיין עליכם להוביל גיבורים למציאת זהב. בשונה מהתרגיל הראשון, הידע הנגיש לכם על מצב העולם הוא חלקי בלבד. כמו כן, חלק מהאובייקטים שהיו קיימים בחלק הראשון, לא יופיעו בחלק זה.

## הבעיה

קקלט תקבלו את המצב ההתחלתי של המשחק. זוהי מטריצה  $M \times N$  שמייצגת חלקית את מצב המשחק. מיקום הבורות והמפלצות לא זמין לכם. בדומה לתרגיל ראשון, לסוכן מותר לזוז ולירות חצים ל-4 כיוונים (ימינה, שמאלה, למעלה ולמטה).

בשונה מהתרגיל הראשון, לא יהיו דלתות ומפתחות, אלא רק גיבורים, מפלצות, קירות, בורות וזהב. מיקום המפלצות והבורות מוסתר מכם, והמידע שניתן לקבל על הימצאותם מוגבל, ונאסף בהדרגה לאורך המשחק. כמובן, אין ביכולתכם לתכנן פתרון כל התרגיל מראש, ותצטרכו להחליט על הפעולה הבאה רק לאחר קבלת תצפיות על מצב העולם - אלו יתקבלו לאחר ביצוע כל פעולה.

## תיאור המשימה

לתרגיל זה מצורף קוד שמתאר את המצב האמיתי של העולם (שאין באפשרותכם לגשת אליו ישירות), ומעביר את המצב העולם החלקי למחלקה `WumpusController` שעליכם לממש. בפרט יש לממש בתוך המחלקה הזאת את הפונקציה `get_next_action` שמקבלת את התצפיות של העולם האמיתי, ומחזירה פעולה שברצונכם לבצע במהלך הבא.

מצב העולם החלקי מורכב ממיקום גיבורים, קירות ותצפיות המעידות על הימצאות אובייקטים אחרים. התצפיות האפשריות כוללות:

- בריזה (breeze) - ברגע שאחד הגיבורים מגיע לתא הנמצא ליד בור (מימין, משמאל מלמעלה או מלמטה), הוא מרגיש בריזה.
- סרחון (stench) - ברגע שאחד הגיבורים מגיע לתא הנמצא במרחק מנהטן 2 או פחות ממפלצת חיה הוא מרגיש סירחון. הסרחון כל כך חזק, שהוא מורגש אפילו אם יש קירות בין הגיבור לבין המפלצת. אם המפלצת נהרגת, הסרחון (באורח פלא) נעלם.
- נצנוץ (glitter) - ברגע שאחד הגיבורים מגיע לתא הנמצא ליד הזהב, הוא רואה נצנוץ.

עליכם לגלות בעזרת תצפיות אלה ושיטות היסק לוגי את מיקום האובייקטים החסרים, ולהגיע עם אחד הגיבורים לאחד התאים עם הזהב.

## פעולות

הפעולות המותרות זהות לאלו של תרגיל 1, עם אותם כללים ותוצאות. לפעמים לא תוכלו לראות מיידית את תוצאת הפעולות שעשיתם - למשל אם יריתם חץ על מפלצת רחוקה לא תוכלו להבחין מיידית שהמפלצת נהרגה.

## ייצוג הקלט

הקלט לפונקציות שעליכם לממש מורכב מ-2 חלקים עיקריים - מפה ותצפיות. ייצוגי האובייקטים הנראים במפה נשארים זהים לאלו של תרגיל 1. ה-tuple של התצפיות מורכב מזוגות בצורה (coordinate, observation)

למשל, אם יש גיבור בתא (1, 1) שמרגיש סירחון, הזוג שמייצג את התצפית הזו יהיה:

((1, 1), 'stench')

הקידוד של שאר אובייקטים על המפה נשאר זהה לזה של תרגיל 1. כל מקום שיכול להיות בור או מפלצת יופיע כמעבר.

## חבילות מותרות לשימוש

מותר להשתמש בחבילות בספריה הסטנדרטית של פייתון, ובכל קטע קוד שפרסמנו. כמו כן, בשביל שיהיה ניתן למחזר את הקוד של תרגיל בית 1, ניתן להשתמש בספריית numpy.

בתרגיל זה אסור להשתמש ב-threading, למרות שהוא נמצא בספריה הסטנדרטית.

כמו כן, אסור לייבא קובץ checker.py או inputs.py, או לנסות לגשת בכל דרך אחרת למפה האמיתית.

## הסבר קצר על הקוד המצורף

הקוד מורכב מ-4 קבצי קוד:

1. ex2.py - קובץ שבו נעשית העבודה העיקרית שלכם, הקובץ היחיד שעליכם לשנות ולהגיש. אמור להכיל את המחלקה WumpusController, המכילה פונקציה get\_next\_action.
2. checker.py - קובץ המכיל את הקוד שממדל את התנהגות העולם כתוצאה מהפעולות שאתם מבצעים.
3. inputs.py - קובץ המכיל מפות שונות לצורך בדיקה עצמית.
4. utils.py - קובץ עזר המכיל הרבה פונקציות שימושיות.

## בדיקת תרגיל

התרגיל ייבדק באופן אוטומטי, ולכן חשוב להקפיד על שמות מדויקים של קבצים, מחלקות ופעולות. הבדיקה תתבצע על ידי הרצת קובץ checker.py.

זמנים שהקוד חייב לעמוד בהם - ה-constructor חייב להסתיים ב-60 שניות. פונקציית get\_next\_action אמורה להסתיים תוך 5 שניות. כמו כן, כל ריצת הקוד מהתחלה עד להשגת מטרה צריכה להסתיים ב- $(5+3M+3N)$  צעדים, כש-M ו-N הינם מימדי המפה.

נא להוריד את כל ההדפסות לפני הגשת התרגיל.

## הגשה, בדיקה, וציונים

- ההגשה ביחידים או בזוגות בלבד
- מספיק שאחד מבני הזוג יגיש את התרגיל, אין צורך ששני בני הזוג יגישו.
- את ת.ז. של המגישים יש לרשום במשתנה ids בקובץ ex2.py בתוך strings בתוך רשימה קיימת. אם המגיש יחיד, הרשימה תכלול string בודד.
- יש להגיש רק את הקובץ ex2.py. אין להגיש קבצים אחרים, או בפורמט אחר (zip וכו').
- 80% מהציון יינתן על עמידה בדרישות בסיס, ו-20% על בסיס תחרות בין המגישים
- דרישות בסיס - פתרון כל תרגיל כך ששום מימד של מטריצת הקלט לא עולה על 6
- נוסחה לציון בתחרות זהה לנוסחה מתרגיל ראשון