תרגיל בית 3 – MDP

מגישות:

לי בנימין **207502444**

שירה לביא 208607689

**חלק א׳ – MDP (44 נק׳)**

**רקע**

בחלק זה נעסוק בתהליכי החלטה מרקובים, נתעניין בתהליך עם **אופק אינסופי** (מדיניות סטציונרית).

**חלק א׳ - חלק היבש** [Writing Hand on Apple ](https://emojipedia.org/apple/ios-14.6/writing-hand/)

1. בתרגול ראינו את משוואת בלמן כאשר התגמול ניתן עבור המצב הנוכחי בלבד, כלומר , למתן תגמול זה נקרא "תגמול על הצמתים" מכיוון שהוא תלוי בצומת שהסוכן נמצא בו.   
   בהתאם להגדרה זו הצגנו בתרגול את האלגוריתמים Value iteration ו-Policy Iteration למציאת המדיניות האופטימלית.

כעת, נרחיב את ההגדרה הזו, לתגמול המקבל את המצב הנוכחי והמצב אליו הגיע הסוכן, כלומר: , למתן תגמול זה נקרא "תגמול תוצאתי". לצורך שלמות ההגדרה, נגדיר שאם לכל   
 מתקיים - אז .

1. (1 נק') התאימו את הנוסחה של התוחלת של התועלת מהתרגול, עבור התוחלת של התועלת המתקבלת במקרה של ״ תגמול תוצאתי ״, אין צורך לנמק.
2. (1 נק') כתבו מחדש את נוסחת משוואת בלמן עבור המקרה של ״ תגמול תוצאתי ״, אין צורך לנמק.

בסעיפים הבאים התייחסו גם למקרה בו , והסבירו מה לדעתכם התנאים שצריכים להתקיים על הסביבה\ על מנת שתמיד נצליח למצוא את המדיניות האופטימלית.

1. (2 נק') נסחו את אלגוריתם Value Iteration עבור המקרה של ״ תגמול תוצאתי ״.

***במקרה בו תנאי העצירה יהיה . עבור ערך זה, לא יהיה הפרש בין הערכים באיטרציה הנוכחית לבין האיטרציה הקודמת ויהיה ניתן לסיים את הריצה.***

1. (2 נק') נסחו את אלגוריתם Policy Iteration עבור המקרה של ״ תגמול תוצאתי ״.

עבור אלגוריתם זה ייתכנס למדיניות האופטימלית רק כאשר מרחב המצבים סופי, מרחב הפעולות סופי, קיים מצב סופי, פונקציית התגמולים חסומה והסוכן מקבל reward שלילי בכל פעולה שהוא מבצע.

נתון הגרף הבא:



נתונים:

* (Discount factor) .
* אופק אינסופי.
* – קבוצת המצבים – מתארים את מיקום הסוכן בגרף.
* – קבוצת המצבים הסופיים.
* קבוצת הפעולות לכל מצב (על פי הגרף), לדוגמא: .
* תגמולים ("תגמול תוצאתי"):
* מודל המעבר הוא דטרמיניסטי, כלומר כל פעולה מצליחה בהסתברות אחת.

1. (יבש 2 נק') הרץ את האלגוריתם Value iteration שכתבת על הגרף הנתון. ומלא את הערכים בטבלה הבאה, כאשר . (ייתכן שלא צריך למלא את כולה).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | 5 | 5 | 5 | 5 | 0 |  |
|  |  |  |  | 7 | 7 | 7 | 7 | 0 |  |
|  |  |  |  | 1.5 | 1.5 | 1.5 | -1 | 0 |  |
|  |  |  |  | 2.5 | 2.5 | 2.5 | -1 | 0 |  |
|  |  |  |  | 2.5 | 2.5 | 2.5 | -1 | 0 |  |
|  |  |  |  | 0.25 | 0.25 | -1.5 | -1 | 0 |  |
|  |  |  |  | 0.25 | 0.25 | -1.5 | -1 | 0 |  |

1. (יבש 2 נק') הרץ את האלגוריתם Policy iteration שכתבת על הגרף הנתון. ומלא את הערכים בטבלה הבאה, כאשר המדיניות ההתחלתית מופיעה בעמודה הראשונה בטבלה. (ייתכן שלא צריך למלא את כולה).

הניחו שבמידה ולא קיים שיפור, האלגוריתם יבחר תמיד להשאיר את הפעולה הקודמת.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

1. (יבש 2 נק') חיזרי על הסעיף הקודם. הפעם עם **אופק סופי כאשר** (שימי לב, המדיניות לא חייבת להסתיים במצב מסיים, ישנם מצבים שלא יכולים להגיע למצב מסיים עם אופק זה. ישנם צמתים עם מספר תשובות נכונות, נקבל את כולם).

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. (1 נק') ללא תלות בשינוי של הסעיף הקודם. אם , מה מספר המדיניות האופטימליות הקיימות? נמקו.

**אם , התועלת של כל מצב תהיה זהה לגמול על הפעולה. עבור תהיה פעולה אחת אופטימלית כל פעם. עבור שאר הצמתים הגמול על כל פעולה הוא 1- אז כל צעד הוא שווה ולכן כולם נחשבים אופטימליים, כלומר מספר המדיניות האופטימליות לצומת כזה הוא כתלות במספר האפשרויות לצעד. סה"כ נקבל שמספר המדיניות האופטימליות הינו:**

***.***

1. (1 נק') ללא תלות בשנוי של הסעיף הקודם, הסבירי מה היה קורה אם

בתשובתך, התייחסי גם לערכי התועלות של כל צומת וגם לשינוי במדיניות, אין צורך לחשב.

**עבור החל מהצעד השני, בו המדיניות כבר לא אפס, למדיניות היה משקל גבוה יותר מאשר לתגמול ולכן השפעה גדולה יותר על המדיניות שתתקבל. צמתים 1 ו-2 היו בעלי מדיניות גבוהה יותר החל מהצעד השני וכן גם עבור יתר הצמתים.**

**השינוי ב- יגרום לכך שערכי התועלת בכל צומת יעלו בכל איטרציה בגלל שערכי התגמול חיוביים ובעצם האלגוריתם ייכנס ללולאה.**

**חלק ב׳ - היכרות עם הקוד**

חלק זה הוא רק עבור היכרות הקוד, עבורו עליו במלואו ווודאו כי הינכם מבינים את הקוד.

mdp.py – **אתם לא צריכים לערוך כלל את הקובץ הזה**.   
בקובץ זה ממומשת הסביבה של ה-mdp בתוך מחלקת MDP. הבנאי מקבל:

* board - המגדיר את המצבים האפשריים במרחב ואת התגמול לכל מצב, תגמול על הצמתים בלבד.
* terminal\_states – קבוצה של המצבים הסופיים (בהכרח יש לפחות מצב אחד סופי).
* transition\_function – מודל המעבר בהינתן פעולה, מה ההסתברות לכל אחת מארבע הפעולות האחרות. ההסתברויות מסודרות לפי סדר הפעולות.
* gamma – discount factor המקבל ערכים - .

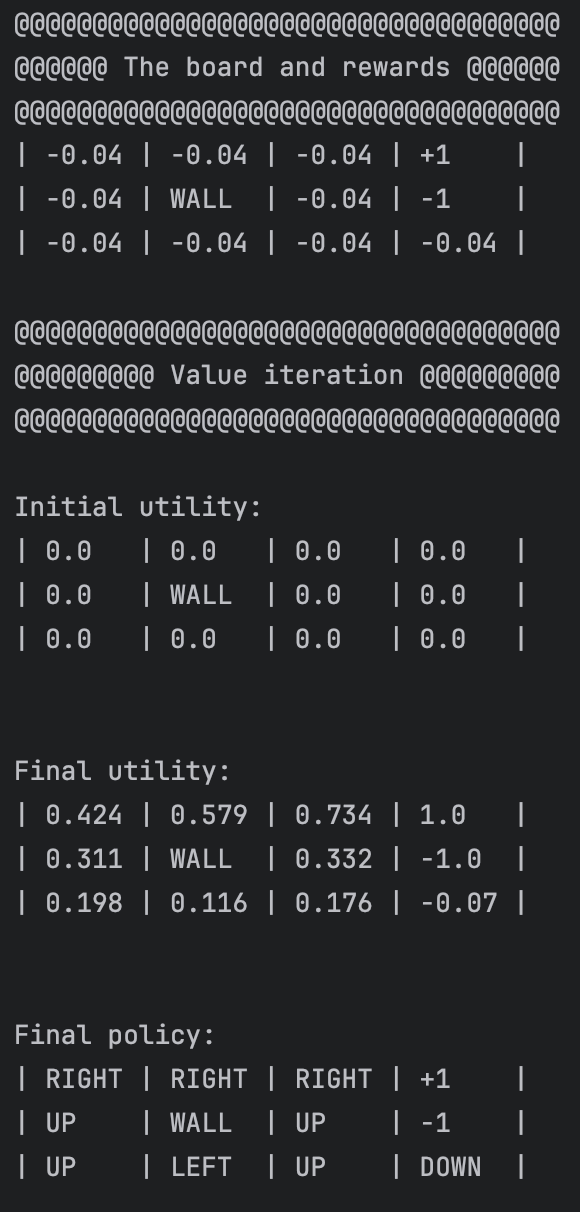
בתרגיל זה לא נבדוק את המקרה בו .

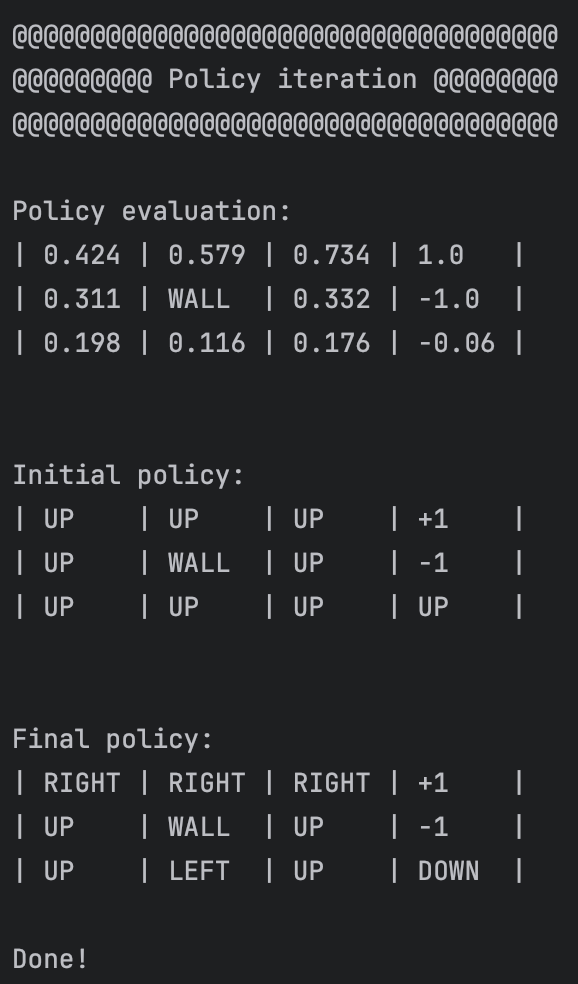
הערה: קבוצת הפעולות מוגדרת בבנאי והיא קבועה לכל לוח שיבחר.

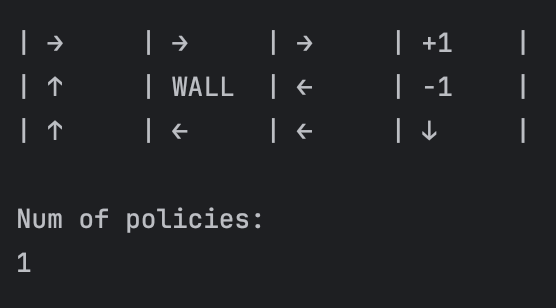
למחלקת MDP יש מספר פונקציות שעשויות לשמש אתכם בתרגיל.

* print\_rewards() – מדפיסה את הלוח עם ערך התגמול בכל מצב.
* print\_utility(U) – מדפיסה את הלוח עם ערך התועלת U לכל מצב.
* print\_policy(policy) – מדפיסה את הלוח עם הפעולה שהמדיניות policy נתנה לכל מצב שהוא לא מצב סופי.
* step(state, action) – בהינתן מצב נוכחי state ופעולה action מחזיר את המצב הבא באופן דטרמיניסטי. עבור הליכה לכיוון קיר או יציאה מהלוח הפונקציה תחזיר את המצב הנוכחי state.

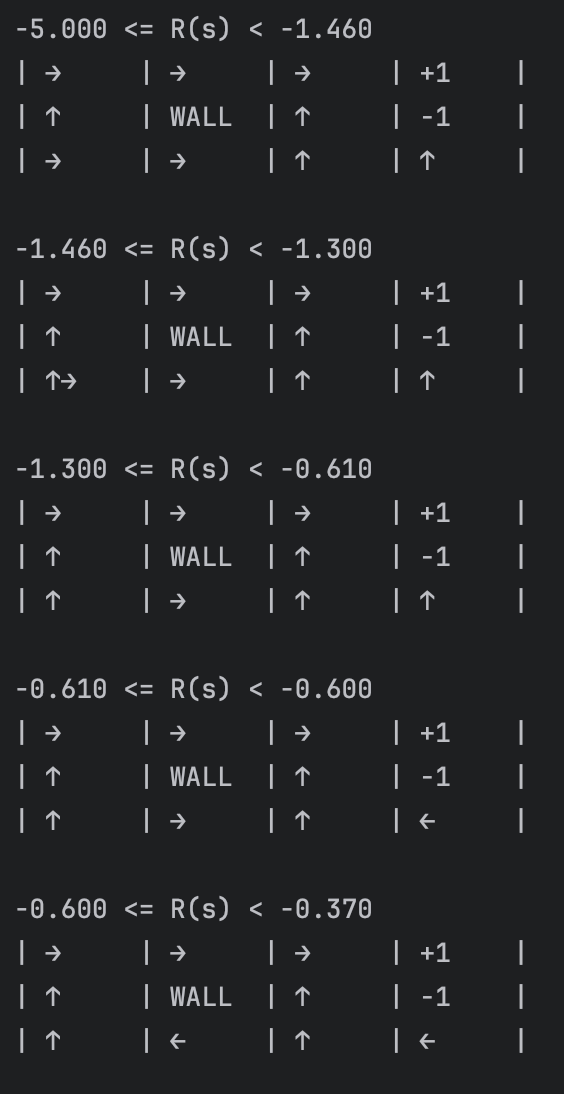
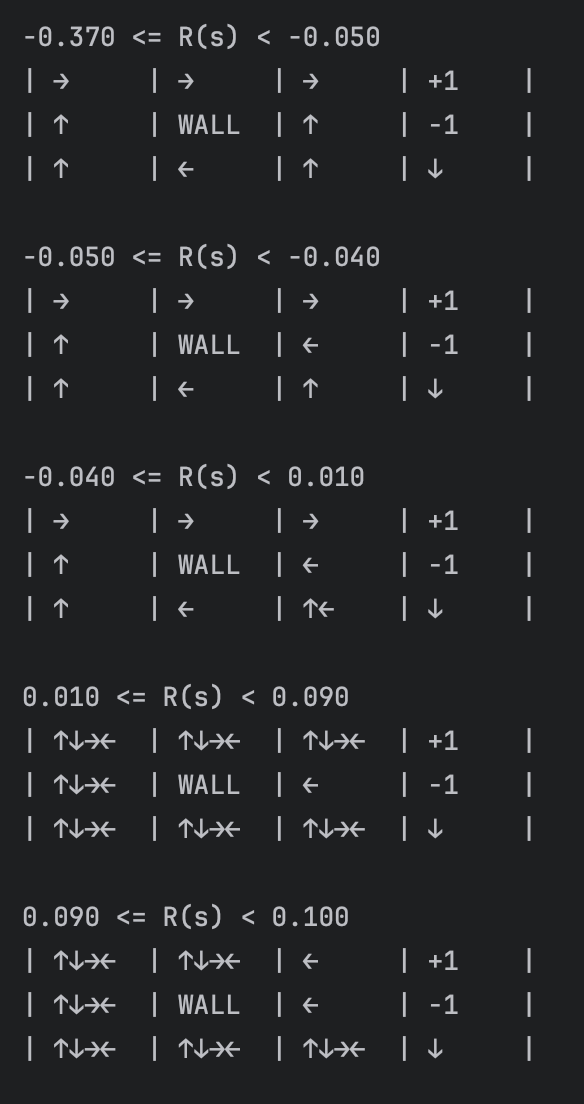
**חלק ג׳ – רטוב**  
מימשנו את הפונקציות הנדרשות בקובץ mdp\_implementation.py המצורף.

הדפסות:



עבור הפונקציה get\_all\_policies(mdp, U, epsilon=10\*\*(-3)) :

עבור הפונקציה get\_policy\_for\_different\_rewards(mdp, epsilon) :

****

****