

תרגיל בית 3:

Checkers

מטרות התרגיל

- בניית שחקן עבור משחק שני שחקנים.
- התנסות בניתוח ובנייה של פונקציות יריסטיות.
- התנסות באלגוריתם *Minimax* ומימוש וריאציות עליו.
- ביצוע מחקר המשווה בין ביצועי שחקנים שונים.

הערות

- תאריך הגשה: 5/1/17.
- את המטלה יש להגיש **בזוגות בלבד!**
- שאלות בנוגע לתרגיל ובקשות **מוצדקות** לדחייה יש לשלוח לנעם: noamrzd@cs.technion.ac.il
- קראו היטב את ההסברים וההוראות במסמך זה, מטרתם לסייע לכם בהבנת הדרישות של התרגיל. שימו לב להוראות ההגשה המצורפות בסוף התרגיל.
- התעדקנו ברשימת ה-FAQ באתר הקורס בתדירות גבוהה, לפני פנייה בשאלות דרך המייל ולפני הגשת התרגיל. ההערות שתתפרסמנה באתר הקורס מחייבות את כלל הסטודנטים בקורס!
- הקוד שלכם ייבדק על-ידי *Unit – Tests*, לכן עליכם לעקוב בתשומת לב רבה אחר הוראות ההגשה המצורפות במהלך התרגיל ובסופו לפני הגשתו.
- בתרגיל זה, כמו גם בתרגילים הבאים בקורס, הרצת הניסויים עשויה לקחת זמן רב ולכן מומלץ מאוד להימנע מדחיית העבודה על התרגיל לרגע האחרון. **לא תינתנה דחיות על רקע זה.**



מבוא והנחיות

במטלה זו תתכננו ותממשו שחקנים למשחק *Checkers* (דמקה). את חוקי המשחק המחייבים לצורך התרגיל ניתן למצוא כאן: https://www.itsyourturn.com/t_helptopic2030.html. הקפידו לקרוא אותם בקפידה טרם העבודה על התרגיל.

לצורך התרגיל, נגדיר מספר חוקים החורגים מההגדרות המופיעות באתר:

1. ביצוע "קפיצה" (*capture*) – אם למשתמש קיימת אפשרות לבצע קפיצה, עליו לקחת אותה. אם קיימות מספר אפשרויות לקפיצה (למשל ע"י כלים שונים), הוא רשאי לבחור כל אחת מהפעולות. אם בסיום הקפיצה קיימת אפשרות לקפיצה נוספת בכיוון התנועה של הכלי (כלומר קדימה בלבד לשחקנים פשוטים), על השחקן לבצע את הקפיצה הנוספת. כך ימשיך לבצע כל קפיצה נוספת עד שיעצר בנקודה בה לא ניתן לבצע עוד קפיצה. למען הנוחות, נקרא למהלך כזה "קפיצה מרובה".
2. סיום המשחק ב"תיקו" – האופן היחיד בו משחק יסתיים בתיקו הוא כאשר מס' התורות מאז הקפיצה האחרונה יעלה על 20. לצורך זה כל זוג מהלכים (אחד מכל שחקן) יחשב לתור. לשחקנים לא קיימת האפשרות להציע תיקו על דעת עצמם.

השחקנים שתממשו, בהינתן מצב לוח, יחזירו את הפעולה לביצוע. בחירת הפעולה תעשה באמצעות ואריאציות שונות לאלגוריתם *Minmax* עם גיזום $\alpha\beta$. על כן, מומלץ לחזור על שקפי ההרצאות והתרגולים בנושא "משחקים" לפני תחילת העבודה על התרגיל. בנוסף, השחקן יהיה מחויב לשתי מגבלות זמן:

1. זמן אתחול (*setup*): מגבלה על זמן הריצה של הבנאי (*constructor*). בזמן זה יוכל השחקן להתכונן למשחק, לאתחל מבני נתונים וכו'. לצורך התרגיל (והתחרות) נקבע זמן זה להיות 2 שניות.
2. זמן עבור k מהלכים: מגבלה על הזמן הלוקח לשחקן לבחור את המהלך הבא. לכל k מהלכים במשחק יש אותה מגבלת זמן, כאשר על השחקן לבחור איך לחלק את הזמן בין המהלכים. זמן שאינו מנוצל בתום k מהלכים אינו נצבר לטובת k המהלכים הבאים. בתרגיל (ובתחרות): $k = 5$.

בעוד שזמן האתחול הוא קבוע, הזמן עבור k מהלכים עשוי להשתנות בין משחקים, ולכן השחקנים שתממשו יפעלו במתכונת anytime-contract עם מדיניות ניהול זמנים לבחירתכם.

לפני שאתם ניגשים לתרגיל, מומלץ להתנסות מעט במשחק על-מנת לקבל הבנה טובה יותר שלו. ניתן למצוא גרסה חינוכית של המשחק כאן: <http://www.247checkers.com>. כמו כן, בקוד שסופק לכם קיים מנוע להרצת המשחק, שחקן אינטראקטיבי (המחכה לקלט מהמשתמש לשם ביצוע מהלך), וכן שחקן רנדומלי ושחקן עם היוריסטיקה פשוטה, אשר ניתן לשחק מולם.

חלק א' – הבנת המשחק (18 נק')

בשלב ראשון, נרצה לקבל הבנה בסיסית להגדרת מרחב המצבים מעליו נפעל. שימו לב כי בחלק זה (חלק א') אין צורך לכתוב קוד.

1. פתחו את הקובץ `checkers/board.py` והסתכלו על המחלקה `GameState`. מהו המבנה של כל מצב (`state`) במרחב המצבים של המשחק?
2. במחלקה `GameState`, הפונקציה `get_possible_moves` מחזירה את כל המהלכים (אופרטורים) שניתן לבצע מהמצב הנוכחי. כל מהלך מוחזר כאובייקט `GameMove` המוגדר בקובץ `checkers/moves.py`. איזה מידע נחוץ כדי לייצג כל אופרטור? איך ניתן להבדיל מעל מבנה זה בין תנועה רגילה (`movement`) לפעולות קפיצה (`capture`)?

בשאלות 3-4 בלבד, נתבונן בווריאציה הבאה של המשחק:

כאשר המשתמש יבחר לבצע "קפיצה מרובה", נסמן x_0, x_1, \dots, x_n את סדרת המיקומים על לוח המשחק שבהם השחקן רוצה לעבור, כאשר x_0 הוא המיקום ההתחלתי של השחקן, ומתקיים $n \geq 2$. רצף הקפיצות יתבצע באופן הבא, עבור $p \in [0,1)$:

- א. אתחול - $i \leftarrow 1$
 - ב. בצע את הקפיצה מ x_{i-1} לעבר x_i
 - ג. בהסתברות p עצור את מהלך הקפיצה והשאר את השחקן במשבצת x_i .
- אחרת, אם $i < n$ עדכן $i \leftarrow i + 1$ וחזור לשלב ב'.

נרצה לערוך את מרחב המצבים שלנו להתאים לווריאציה זו. בפרט, נרצה להוסיף מצב הסתברותי **יחיד** בין צומת max לצומת min עוקב ולהיפך (בין כל צומת min לצומת max עוקב) עבור כל אופרטור המייצג "קפיצה מרובה".

3. איזה שינויים יש לבצע בייצוג המצבים במשחק (כפי שתואר בשאלה 1)? הסבירו.
4. אילו מהלכים יוחזרו ע"י פונקציית העוקב (`successor`) של מצב הסתברותי המתאר "קפיצה מרובה" באורך n ? מה מקדם הסיעוף של מצב זה? מה תהיה ההסתברות לכל אחד מהמהלכים?

חלק ב' – הבנת השחקן הפשוט (10 נק')

קראו את הקוד של השחקן הפשוט המסופק כדוגמא ב `players/simple_player`. שימו לב כי בחלק זה (חלק ב') אין צורך לכתוב קוד.

5. מהי הגישה (הנאיבית) בה נוקט השחקן לשם חלוקת הזמן בין כל k מהלכים?
6. ציינו חסרון אפשרי של גישה זו והסבירו כיצד שחקן חכם יותר יוכל להתגבר עליו.
7. מהי הפונקציה היוריסטית בה משתמש השחקן? הסבירו את המוטיבציה ליוריסטיקה זו.

חלק ג' – שיפור השחקן (24 נק')

בחלק זה של התרגיל תדרשו לממש שלושה שחקנים. על כל שחקן להיות ממומש בתיקיה נפרדת על פי שמו. מחלקת השחקן תקרא *Player* ועליה לרשת מהמחלקה *AbstractPlayer* (או מהמחלקה *Player* של *simple_player*). ניתן למצוא הסבר מפורט על הפורמט הנכון לשמירת השחקנים בהוראות ההגשה בעמוד האחרון של התרגיל.

8.

- i. הגדירו באופן מפורש יוריסטיקה משלכם להערכת מצבי המשחק. בפרט, יש לספק נוסחה עבור כל פרמטר של המשחק הנלקח בחשבון לשם חישוב היוריסטיקה.
- ii. הסבירו את המוטיבציה להגדרה זו (ולכל פרמטר המופיע בה). מדוע אתם צופים שהיא תשפר את ביצועי השחקן ביחס ליוריסטיקה של *simple_player*?
- iii. ממשו שחקן בשם *better_h_player* הנעזר ביוריסטיקה שהגדרתם.

כעת נרצה לממש שיפור לאלגוריתם $\alpha\beta$ - *Minmax* בכדי לשפר את ביצועי השחקן שלנו. יש לענות רק על **אחת מהשאלות 9-10**, ולציין **במפורש** בדו"ח על איזו שאלה עניתם. את מימוש השחקן המשופר מהשאלה שבחרתם (ללא היוריסטיקה משאלה 8) יש לשמור בשם *improved_player*.

9. ניהול זמנים:

- i. הגדירו שיטה אינטליגנטית לחלוקת הזמן של השחקן בין כל k מהלכים. שיטה נאיבית ניתן, כאמור, למצוא בשחקן המסופק לכם. יש להציג נוסחה כללית במקרה הצורך, ונוסחה עבור כל פרמטר של המשחק הנלקח בחשבון לשם חישוב זה.
- ii. הסבירו את המוטיבציה מאחורי השיטה שמימשתם. באילו מצבים השיטה משקיעה פחות זמן חיפוש, ובאילו יותר? מדוע לדעתכם פעולה זו תהיה משמעותית לשיפור השחקן הפשוט?
- iii. ממשו שחקן עם מדיניות ניהול הזמנים שהגדרתם.

10. העמקה סלקטיבית:

- העמקה סלקטיבית היא הבחירה לאפשר לאלגוריתם החיפוש $RB - \alpha\beta$ להעמיק את החיפוש ממצבים מסוימים בעץ המשחק מעבר לעומק D (מגבלת העומק של האלגוריתם). בכיתה למדנו כי אחת הסיבות לבצע העמקה סלקטיבית היא כאשר אנו מזיהים מצב "לא שקט".
- i. הציעו קריטריון לצמתים בהם נרצה להמשיך להעמיק את החיפוש מעבר למגבלת העומק. יש להגדיר במפורש את אופן חישוב הקריטריון, ובמקרה הצורך להציג נוסחה עבור כל פרמטר של המשחק הנלקח בחשבון לשם חישוב זה.
 - ii. תארו את הרציונל מאחורי הקריטריון שהגדרתם. מדוע יש להעמיק יותר במצבים העומדים בקריטריון הנ"ל? מדוע לדעתכם פעולה זו תהיה משמעותית לשיפור השחקן הפשוט?
 - iii. ממשו שחקן הנעזר בקריטריון שהגדרתם להגבלת עומק החיפוש של האלגוריתם. לצורך זה, ניתן להיעזר בפונקציה *selective_deepening_criterion* הנמצאת במחלקה *Player.simple_player*.

11. ממשו שחקן בשם *improved_better_h_player* המשתמש ביוריסטיקה ובשיפור שהגדרתם בשאלות 8-10.

חלק ד' – ניסויים, תוצאות ומסקנות (48 נק')

נרצה להשוות בין שחקני אלפא-בטא הנעזרים ביוריסטיקות השונות שראינו, עם ובלי שיפורים, תחת מגבלות זמן שונות. בפועל, ההשוואה תיעשה בין 4 שחקנים:

(1) *simple_player* המסופק (שחקן baseline)

(2) *better_h_player* שמימשתם בשאלה 8

(3) *improved_player* שמימשתם בשאלה 9 או 10

(4) *improved_better_h_player* שמימשתם בשאלה 11

נסמן $T = \{2, 10, 50\}$. עבור כל זוג שחקנים (סה"כ $\binom{4}{2}$ צמדים) הריצו סדרת משחקים עבור כל מגבלת

זמן $t \in T$ (בשניות). להזכירכם, t מגביל את זמן החישוב של השחקן עבור k מהלכים (כפי שהוסבר במבוא). בכל מגבלה, הריצו 2 משחקים, אחת לכל חלוקת צבעים לשחקנים. סה"כ עליכם להריץ 36 משחקים.

לצורך ניתוח התוצאות, נשתמש בשיטת הניקוד הבאה:

- הניקוד למשחק יחיד - ניצחון = 1 נק', תיקו = 0.5 נק', הפסד = 0 נק'.
- הניקוד הכולל של שחקן יחושב כסכום הנקודות שלו מכל המשחקים בהם הוא לקח חלק.

12. הגישו קובץ בשם *experiments.csv* שיכיל את תוצאות המשחקים שהרצתם. כל שורה בקובץ תכיל 5 עמודות: שם השחקן האדום (מבין 4 השחקנים המשתתפים בניסוי), שם השחקן השחור, מגבלת הזמן בשניות ל $k = 5$ תורות, ניקוד השחקן האדום, ניקוד השחקן השחור. לנוחיותכם מסופקת שורה לדוגמא:

simple_player, improved_player, 10, 0.5, 0.5

אין לכלול כותרות בקובץ, אלא רק את תוצאות הניסויים.

13. נרצה להשוות את ביצועי השחקנים על פני מגבלות הזמן השונות:

א. הציגו גרף המתאר עבור כל אחת ממגבלות הזמנים את הניקוד הכולל של כל שחקן

כפונקציה של מגבלת הזמן. הקפידו להשתמש בצבעים או בסוגי קו שונים עבור שחקנים

שונים. לנוחיותכם מצורף גרף לדוגמא (*).



II. עבור גרף זה, צרפו גם את טבלת הנתונים שיצרו אותו. למשל (*):

$t = 50$	$t = 10$	$t = 2$	
1	2.5	5	<i>simple_player</i>
4	4	2	<i>better_h_player</i>
3	3	4	<i>improved_player</i>
4	2.5	1	<i>improved_better_h_player</i>

- III. השוו את ביצועי השחקנים עם היוריסטיקה הפשוטה (שחקנים (1) ו(3)) לאלו עם היוריסטיקה שהצעתם בשאלה 8 (שחקנים (2) ו(4)) ביחס למגבלת הזמן ע"י ניתוח של מגמות השיפור והדעיכה בניקוד הכולל של כל אחד מהשחקנים. באילו מקרים היוריסטיקה שלכם מתגברת על היוריסטיקה הפשוטה? הציעו הסבר לממצאים הנ"ל.
- IV. השוו את ביצועי השחקנים בעלי השיפור משאלה 9 או 10 (שחקנים (3) ו(4)) לאלו ללא שיפור זה (שחקנים (1) ו(2)) ביחס למגבלת הזמן ע"י ניתוח של מגמות השיפור והדעיכה של כל אחד מהשחקנים. באילו מקרים השיפור שלכם משפר את ביצועי השחקן? הציעו הסבר לממצאים הנ"ל.

(*) שימו לב כי הגרפים והטבלאות שהצגנו כאן נוצרו על סמך ערכים שרירותיים שקבענו לצורך **המחשה בלבד**. לא מובטח שהתוצאות שתקבלו בהרצת הניסויים שלכם תהיינה דומות לערכים המוצגים בגרף ובטבלה שלעיל.

14. **בהסתמך על תוצאות הניסוי:** הציעו אלגוריתם לשחקן חמישי, טוב יותר מארבעת השחקנים שבניסוי, המתבסס על השחקנים שנבנו בשאלות הקודמות. הציעו שיפור אחד לפחות. הסבירו על אילו תוצאות הסתמכתם ומדוע אתם מסיקים מהן שהשחקן המשופר יגלה יתרון על ארבעת הקודמים. שיפורים כלליים, שאינם נובעים מתוצאות הניסוי, **לא יקבלו את הנקודות לשאלה**.

חלק ה' – התחרות

נרצה לבחון את הידע שצברתם במשחקים בעזרת תחרות. על כן, כל זוג נדרש לממש שחקן יחיד שייצג אותו במשחק דמקה. צוות הקורס יקיים טורניר בין השחקנים שיוגשו, והסטודנטים ששחקניהם יצטיינו בתחרות יזכו בבונוס מכובד לציון הסופי של הקורס (ובתהילת עולם).

בחירת השחקן:

אתם רשאים להשתמש באחד השחקנים שמימשתם בסעיפים הקודמים או לחילופין לממש שחקן חדש כרצונכם, אך ההשתתפות בתחרות היא **חובה**. למטלה זו אין משקל בניקוד תרגיל בית 3 עם סייג אחד: **במידה והשחקן שתגישו לא ירוץ, תיתכן הורדת נקודות על התרגיל.**

מסגרת הזמן:

בתחרות שנקיים בין השחקנים שתגישו, יאכפו שתי מגבלות זמנים:

- מגבלת זמן של 2 שניות לשחקן עבור שלב האתחול
- מגבלת זמן של מספר שניות לשחקן עבור כל 5 מהלכים.

מגבלות אלו זהות לאלו שבחנתם במהלך התרגיל. שימו לב שהדבר מאפשר לכם חופש פעולה בהגדרת הזמן שברצונכם להקצות לחישוב כל אחד מהמהלכים. הסוכנים ייבחנו תחת 3 מגבלות הזמנים שהוגדרו בשלב הניסוי. זמן החישוב הכולל של כל k מהלכים צריך להיות נמוך מהמגבלה המוגדרת. שחקן שיחרוג מהזמן המוקצה, יפסיד טכנית במשחק הספציפי בו קרתה החריגה.

חוקים:

- מימוש השחקן יופיע בתיקיה בשם `competition_player`. על מחלקת השחקן להקרא `Player` ולרשת מהמחלקה `AbstractPlayer` (או מהמחלקה `Player` של `simple_player`). ניתן למצוא הסבר מפורט על הפורמט הנכון לשמירת כלל השחקנים בהוראות ההגשה בעמוד האחרון של התרגיל.
- אתם רשאים לבחור עבור השחקן שלכם שם יצירתי כרצונכם ע"י דריסת הפונקציה `__repr__` במחלקה `Player` באופן הבא:

```
def __repr__(self):
    return '{} {}'.format(abstract.AbstractPlayer.__repr__(self),
                           'Senorita_Analphabeta')
```

- אסור לשחקן להשתמש ברשת האינטרנט או ברשת מקומית כלשהי.
- אסור לשחקן להשתמש בכל סוג של קוד מקבילי.
- צריכת הזיכרון של השחקן שלכם מוגבלת ל-512 MB.
- כל שחקן שינסה לרמות או לשבש את קוד היריב או המערכת ייפסל.
- חל איסור להשתמש במידע שעובד מראש (`pre-processed`). ניתן לבצע פעולות אתחול ו/או עיבוד מקדים בבנאי (`__init__`) של השחקן בלבד, וזאת במסגרת מגבלת זמן האתחול.
- אין להשתמש ב-C extensions.

פרסים:

מקום ראשון: 5 נק'. 

מקום שני: 4 נק'. 

מקום שלישי: 3 נק'. 

מקום רביעי: 2 נק'. 

מקום חמישי: 1 נק'. 

לקבלת הבונוס, נבקש מחמשת המנצחים להגיש מסמך PDF קצר בן כעמוד (לא יותר משניים), המתאר את השחקן שלהם. במסמך נבקשכם להתייחס לפחות לחמש השאלות המנחות הבאות:

1. מה התכונות הייחודיות של השחקן שלכם לעומת שחקן אלפא-ביתא גנרי? בפרט חשוב להתייחס להגדרת היוריסטיקות ולכל שיפור או שינוי שמימשתם באלגוריתם.
2. מה להערכתכם היתרון היחסי של השחקן שלכם לעומת השחקן המסופק ושחקנים אחרים שסטודנטים אחרים אולי מימשו? באילו מצבים ו/או סוגי משחקים אתם צופים שהשחקן שלכם יהיה טוב מאחרים?
3. מהם החסרונות או נקודות החולשה של השחקן שלכם, להערכתם? מדוע?
4. תארו תובנה אחת מעניינת שהגעתם אליה במהלך הפיתוח בהקשר למשחק זה, לאסטרטגיות בו, לאלגוריתם אלפא-ביתא או כל דבר אחר. האם תובנה זו השפיעה על המימוש של שחקן התחרות, אם כן - כיצד, אם לא – מדוע?
5. אם היה לכם זמן נוסף בלתי מוגבל להמשיך ולפתח את השחקן שלכם, כיצד הייתם מנצלים אותו על מנת לשפרו?

הוראות הגשה

- הגשת התרגיל תתבצע **אלקטרונית בלבד** דרך אתר הקורס.
- כהכנה להגשה, אתם מתבקשים ליצור בתוך התיקייה *players* שבקוד המסופק **תיקייה יחידה** בשם *AI3_{id1}_{id2}* (ללא הסוגריים המשולשים), שבתוכה הקבצים הבאים:
 - קובץ בשם *readme.txt* בפורמט הבא:


```
name1 id1 email1

name2 id2 email2
```
 - קובץ ריק בשם *__init__.py*
 - 4 תיקיות ששמן: *better_h_player, improved_player, improved_better_h_player, competition_player*
- עבור 3 השחקנים שמימשתם בחלק ג' והשחקן אותו אתם מגישים לתחרות. על כל אחת מהתיקיות הללו להכיל קובץ *__init__.py* עם קוד השחקן שלכם (או *import לקוד*). מחלקת השחקן צריכה להיקרא *Player* ולרשת מהמחלקה *AbstractPlayer* (או מהמחלקה *Player* של *simple_player*) שבקוד המסופק.
 - כל קובץ עזר שכתבתם, אשר קוד השחקנים משתמש בו.
 - קובץ בשם *requirements.txt* שיכיל כל חבילה חיצונית שאינה מותקנת כחלק מ-*Anaconda Python*. על אחריותכם לוודא כי ניתן להתקין כל חבילה כנ"ל באמצעות הרצת השורה: *pip install -r requirements.txt*
 - קובץ בשם *AI_HW3.pdf*, המכיל את התשובות לחלק היבש והערות מיוחדות הנוגעות לקוד במידה ויש צורך בכך.
- על-מנת לוודא שמבנה הקבצים הנ"ל תקין, בדקו שאתם מסוגלים להריץ את הפקודה הבאה:


```
python run_game.py 2 3 5 y AI3_{id1}_{id2}.competition_player simple_player
```
- פקודה זו תריץ משחק בין שחקן התחרות שלכם (בתור השחקן האדום) לבין השחקן הפשוט. את התיקייה *AI3_{id1}_{id2}* יש לקבץ לארכיון בשם *AI3_{id1}_{id2}.zip*, **שאותו (ואותו בלבד) עליכם להגיש**.
- אין להעתיק את הקבצים המסופקים לכם אל תוך תיקיית ההגשה. הניחו כי קבצים אלו יהיו זמינים בעת בדיקת התרגיל.
- שימו לב שכל הפנייה למיקום קובץ/תיקייה כלשהם בקוד תהיה רלטיבית (*relative path*) ולא אבסולוטית, כך שהקוד יעבוד כפי שהוא על כל מחשב בכל מיקום שנבחר לתיקיית הפרויקט. הקפידו לבדוק זאת לפני ההגשה!
- הקפידו על קוד **ברור, קריא ומתועד!** עליכם לתעד כל חלק שאינו טריוויאלי בקוד שלכם. בפרט, אם התשמשתם בקוד שנמצא ברשת וביצעתם בו שינויים, עליכם לתעד זאת.
- "המצאת" נתונים לצורך בניית הגרפים **אסורה** ותוביל לדיון בבית הדין המשמעתי של הטכניון.
- אתם רשאים לעשות שימוש בכל קוד שתמצאו ברשת, אך כל קוד חיצוני מצריך הצהרה מפורשת על המקור שלו בקובץ *AI_HW3.pdf*.

בהצלחה!