

PyWar

תוכן עניינים

1	<u>פרק ראשון: עולם המשחק</u>
1	הקדמה
1	מהלך התרגיל
1	מה עושים?
2	משחק המלחמה
2	מטרת המשחק
2	העולם
3	הצבא
5	איך משחקים?
5	חלוקה לצוותים
5	המטרה של הצוותים
5	העבודה של הצוותים
6	איך נראית העבודה עצמה?
6	האתגר האסטרטגי והאלגוריתמי
6	הכלים
6	טנקים
7	ארטילריה
7	מטוס
7	מסוק
8	נ"ט
8	כיפת ברזל
8	בונקרים
9	מרגלים
9	מגדלי תצפית
9	לויני ריגול
9	בנאים
10	סיכום
10	רמות נראות של משבצות
11	מהלך קרב
11	מצבים של כלים לוחמים
12	איך נראה קרב?

12	שלב א': זיווג
12	שלב ב': נלחמים
12	שלב ג': תוצאות
12	מה קורה לכלים כשהמשבצת שלהם נכבשת?
13	<u>פרק שני: הפעלת המשחק</u>
13	ה-API-ים
13	ה-API הטקטי
13	המחלקות ב-API הטקטי
16	ה-API האסטרטגי
17	המחלקות ב-API האסטרטגי
17	מימוש ה-API האסטרטגי
18	שימוש ב-API האסטרטגי
18	משהו בסיסי לדוגמה
19	הערכה למשתתף המתחיל
19	מה קיבלתם ומה עוד תקבלו?
19	קיבלתם
20	תקבלו
20	שרת ההרצה של המשחק - הסבר כללי
20	חיבור לשרת
20	שרת ההרצה - Code Versions
20	מידע כללי
20	כשאין גרסאות קוד
20	כשיש גרסאות קוד
20	סימון הקוד הראשי
21	מחיקת גרסת קוד
21	העלאת גרסת קוד חדשה
22	העלאה בעזרת האתר
23	העלאה בעזרת <code>upload_script.py</code>
24	שרת ההרצה - Bots
24	מידע כללי
25	שרת ההרצה - Runs
25	מידע כללי
25	כשאין הרצות
25	כשיש הרצות
25	יצירת הרצה חדשה (אחרי בחירת מפה ושחקנים)
25	צפייה במידע על הרצה
27	צפייה בהרצה עצמה
28	מחיקת הרצה
28	שרת ההרצה - Tournaments
28	מידע כללי
29	כשאין טורנירים

29	כשיש טורנירים
29	יצירת טורניר חדש
31	לוגים וטיפול בשגיאות
31	שגיאות נפוצות
32	תוצאות השגיאות
32	הדפסה לקבצי הלוג

פרק ראשון : עולם המשחק

הקדמה

ברוכים הבאים לתרגיל PyWar! משחר ימי ארזים, תרגיל זה עובר למחזורי הארזים באוניברסיטה בסוף סדנאות הקיץ. זה הזמן שלכם להכיר קצת את האנשים שילוו אתכם במשך השנה הקרובה, ללמוד ולהינות!

מהלך התרגיל

מה עושים?

עבור התרגיל אתם תתחלקו לקבוצות. כל קבוצה היא קבוצת שכירי חרב ואתם הולכים לשחק במשחק מלחמה ממוחשב באמצעות בוטים. המטרה שלכם, בתור קבוצה, היא לבנות את הבוט הכי טוב שאתם יכולים ולנצח את הבוטים של שאר הקבוצות.

משחק המלחמה

מטרת המשחק

המשחק הוא משחק לוח המבוסס תורות. הבוט שלכם ינהל צבא שיכיל כל מני כלי מלחמה שחיים במשבצות בלוח. תאספו כסף, תבנו כלים, תשמידו את צבא היריב ותכבשו כמה שיותר משבצות - במטרה להיות הצבא ששולט בכמות המשבצות הגדולה ביותר בלוח.

העולם

במשחק, אתם נמצאים בתוך עולם ריבועי שמחולק למשבצות. ניתן להתייחס לכל משבצת כזוג קואורדינטות שלמות, ולכל משבצת יש לכל היותר ארבע משבצות שכנות (מלמעלה, מלמטה, מימין ומשמאל). למשבצת יש כמה פרמטרים:

- שייכות למדינה: משבצת יכולה להיות שייכת למדינה מסוימת או לא להיות שייכת לאף מדינה.
- כלי מלחמה: משבצת יכולה להכיל כלי מלחמה של כל אחד מהשחקנים.
- כסף: משבצת יכולה להכיל סכום כסף כלשהו.

העולם נקבע בתחילת המשחק, והוא ישתנה בין ההרצות השונות בהן הבוטים שלכם יתחרו. גודל הלוח, כמות הכסף ההתחלתית¹ ומצב הכלים ההתחלתי למעשה מהווים את התנאים השונים בין קרב לקרב.

הצבא

הצבא שלכם מחולק לארבעה חלקים: צבא התקפי, צבא הגנתי, מודיעין ובנייה.

¹ נקבעת לכל משבצת בנפרד.

בשלב זה, נציג את הכלים במשחק אבל נפרט על הפעולות שלהם ועל התפקיד שלהם מאוחר יותר.

- הצבא ההתקפי מכיל את הכלים הבאים: טנקים, ארטילריה, מטוסים ומסוקים.
- הצבא ההגנתי מכיל את הכלים הבאים: נ"ט [נגד טנקים], כיפות ברזל ובונקרים.
- המודיעין מכיל את הכלים הבאים: מרגלים, מגדלי תצפית ולוויני ריגול.
- הבנייה מכילה את הכלים הבאים: בנאים.

איך משחקים?

חלוקה לצוותים

לתרגיל יש שני API-ים: API אסטרטגי ו-API טקטי [במקרה זה, API הוא רשימת פקודות שצד אחד מממש אותן והצד השני משתמש בהן]. הקבוצה שלכם תתחלק לשני צוותים:

- הצוות הטקטי: ישתמש ב-API הטקטי (שאותו אנחנו מימשנו כבר) כדי לממש את ה-API האסטרטגי.
- הצוות האסטרטגי: ישתמש ב-API האסטרטגי כדי לבנות את הבוט.

העבודה תתנהל באופן היררכי, כך שלכל צוות יהיה ראש צוות ועל שני הצוותים יפקד ראש מדור (רמ"ד). נצלו את הכישורים שלכם והיעזרו אחד בשני בדרככם לבנות יחד את הבוט האידיאלי. שימו לב שכדי לבנות בוט טוב ה-API האסטרטגי צריך להיות ממומש בצורה חכמה וצריך להשתמש בו בצורה חכמה. רק עבודה אלגוריתמית חכמה של שני הצוותים תוביל אתכם לניצחון!

המטרה של הצוותים

המטרה העיקרית של הצוותים היא להתאמן על כתיבת קוד גדול וכתיבת קוד נכונה. המטרה העיקרית של המפקדים היא להתאמן על ניהול פרויקטים.

העבודה של הצוותים

- הצוות הטקטי עובד על קובץ שמממש את ה-API האסטרטגי.
 - הצוות האסטרטגי עובד על קובץ שמגדיר מה יעשה הבוט.
- את שני הקבצים האלו אתם צריכים להגיש וככה נוכל להריץ את הבוטים שלכם. ניתן ואף מומלץ לפצל את הקוד שלכם לקבצים נוספים ולהגיש גם אותם. הקוד שאתם כותבים ילווה אתכם ביומיים הקרובים - נסו להקדיש תשומת לב לתכנון נכון וחשיבה קדימה.

נזכיר כי הצבא מחולק לארבעה חלקים, ועל כן גם ה-API האסטרטגי מחולק לפי ארבעת החלקים האלו. שמרו את העבודה בצוות הטקטי לפי ארבעת החלקים האלה ותנו לאנשים שונים לממש חלקים שונים ב-API. שימו לב שאת התיאום בין החלקים הטקטיים יבצע הצוות האסטרטגי בלבד!

איך נראית העבודה עצמה?

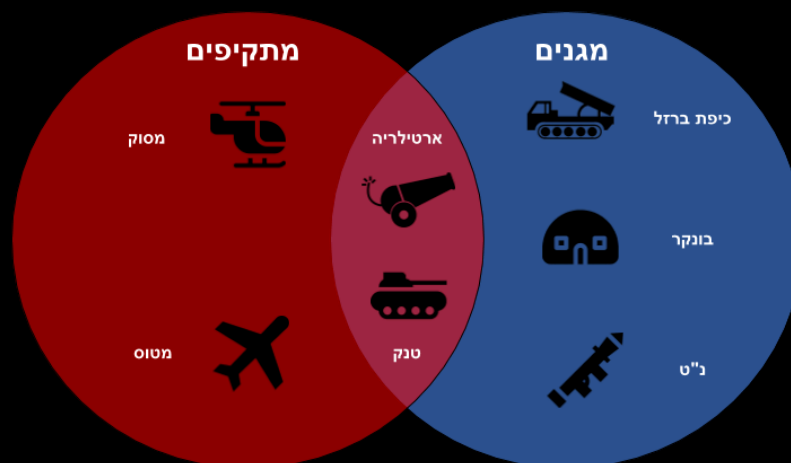
לאחר שתסיימו לקרוא את ההוראות, תתחילו לעבוד. פעם בכמה זמן תתרחש הרצה של הבוטים. אתם תגישו את הבוט העובד הכי טוב שיש לכם באותו רגע, נריץ את הבוטים שלכם אחד נגד השני ונראה מי מנצח. נזכיר כי בכל סיבוב העולם יהיה שונה. הקבוצה שתנצח בהכי הרבה סיבובים תנצח במשחק!

האתגר האסטרטגי והאלגוריתמי

כשאתם כותבים בוטים שמשחקים את המשחק, אחד האתגרים המרכזיים שלכם הוא האתגר האלגוריתמי. המטרה היא לא רק לכתוב קוד, אלא לכתוב אותו בצורה טובה. לדוגמא, אפשר לכבוש משבצת עם טנק על ידי הגרלת טנק, הגרלת משבצת שאינה של המדינה שלי, שליחת הטנק למשבצת ותקיפה עם הטנק. לעומת זאת מימוש טוב יותר אלגוריתמית הוא בחירה של טנק שהכי קרוב לגבול, לבחור את משבצת הגבול הכי קרובה ולתקוף שם.

לכל תור מוקצה זמן ריצה של עד 5 שניות. אם בזמן הזה התור לא נגמר, התור לא יתבצע. בנוסף, המשחק עצמו לא ירוץ יותר משתי דקות. מעבר לכך אין משמעות ליעילות. שימו לב שיש כאן הרבה שכבות של אבסטרקציה שיכולות להעמיס על הריצה, ונסו להשתדל שהקוד שלכם ירוץ בזמן סביר.

הכלים




נתאר כעת את התפקוד של כלי המלחמה השונים. שימו לב כי בעת קרב האינטראקציה של כל אחד מהם עם השני מתוארת בטבלת הקרב. בכל התיאורים כאן המרחקים כתובים ב"מרחק מנהטן" (נורמת L1), כלומר סכום המרחק האנכי והאופקי. זהו גם מספר הצעדים המינימלי הדרוש להגעה ממשבצת אחת לאחרת כאשר אסור לנוע באלכסון (כך הכלים שלכם זזים). כאשר כתוב טווח 0 הכוונה היא למשבצת שבה הכלי נמצא בלבד.

טנקים

תיאור כללי: טנק הוא כלי טווח קרוב שיכול לתקוף ולהגן על המשבצת בה הוא נמצא. הוא הכלי היחיד שיכול לכבוש משבצת. טנק מגן כל עוד הוא לא קיבל פקודה לתקוף.

פקודות: לזוז, לתקוף.

טווח תקיפה:  0.

טווח הגנה:  0.

טווח ראייה:  1.


מהירות:  1.


מחיר:  8.

ארטילריה


תיאור כללי: ארטילריה היא כלי טווח רחוק שיכול לתקוף ולהגן על משבצות במרחק ממנו, ולא על המשבצת שלו. כל עוד הארטילריה לא תוקפת, היא מגנה על כל המשבצות בטווח ההגנה שלה. בקרב מול ארטילריה במשבצת בה היא נמצאת, נתייחס אליה כאל "ארטילריה פאסיבית".

פקודות: לזוז, לתקוף.

טווח תקיפה:  3.

טווח הגנה:  2 * (לא כולל את המשבצת בה היא נמצאת).

טווח ראייה:  1.


מהירות:  1.

מחיר:  8.

מטוס

תיאור כללי: מטוס הוא כלי טווח קרוב שיכול רק לתקוף. הוא זז מהר מאוד, חסין נגד טנקים ונ"ט אבל פגיע למטוסים אחרים וארטילריה. מטוס יכול להיות על האדמה או באוויר; הוא יכול לזוז ולתקוף רק כשהוא באוויר, והוא יכול להיות באוויר לכל היותר 16 תורות ברצף. מטוס שנחת בשטח אויב עובר צד.

פקודות: לזוז, לתקוף, להמריא, לנחות.

טווח תקיפה:  0.

טווח הגנה:  -.

טווח ראייה:  1.

מהירות:  8.


מחיר:  20.

מסוק


תיאור כללי: מסוק הוא כלי טווח קרוב שמסייע בקרבות ולא יכול להיפגע. הוא לא יכול להגן, וכדי לסייע הוא צריך להצטרף לטנקים, מטוסים או לארטילריה. מסוק יכול לסייע ולזוז רק כאשר הוא באוויר, והוא יכול להיות באוויר לכל היותר 8 תורות ברצף. מסוק יעבור צד אם המשבצת שלו נכבשת או שהוא נוחת בשטח שאינו שלו.

פקודות: לזוז, לתקוף, להמריא, לנחות.

טווח תקיפה:  1.

טווח הגנה:  -.

טווח ראייה:  1.

מהירות:  2.

מחיר:  16.

נ"ט


תיאור כללי: נ"ט מגן על שטח מפני טנקים תוקפים. הוא הורס רק טנקים, ולא נפגע בקרב איתם. הוא תמיד מגן על המשבצת שהוא נמצא בה. אם שטח שהנ"ט מגן עליו נכבש, הנ"ט נעלם.

פקודות: לזוז.

טווח תקיפה:  -.

טווח הגנה:  0.

טווח ראייה:  1.


מהירות:  1.

מחיר:  10.

כיפת ברזל

תיאור כללי: כיפת ברזל היא כלי הגנתי שלא יכול לתקוף. יש לה טווח הגנה גדול בו היא מגנה מפני אויבים אוויריים ואף יכולה ליירט ירי של ארטילריה. היא לא יכולה להשמיד כלי קרקע (ובפרט, ארטילריה). לכיפת ברזל שני מצבים - הגנה פועלת והגנה כבויה. כאשר ההגנה פועלת היא לא יכולה לזוז, וכאשר ההגנה כבויה היא יכולה לזוז (וכמובן לא להגן).

פקודות: לזוז, להפעיל הגנה, לכבות הגנה.

טווח תקיפה:  -.

טווח הגנה:  10.

טווח ראייה:  1.

מהירות:  1.

מחיר:  32.





בונקרים


תיאור כללי: בונקר הוא כלי הגנתי סטטי. הוא לא יכול לזוז, והוא יכול לספוג לכל היותר 5 פגיעות בכל קרב. אם הוא ספג 5 פגיעות בקרב, הוא יושמד; לעומת זאת, אם ספג פחות מחמש פגיעות הוא ימשיך לסיבוב הבא בו שוב ויכל לספוג לכל היותר 5 פגיעות. אם המשבצת בה בונקר יושב נכבשת, הוא עובר צד.

פקודות: -.

טוח תקיפה:  -.

טוח הגנה:  0.

טוח ראייה:  1.

מהירות:  -.

מחיר:  10.





מרגלים

תיאור כללי: מרגל הוא כלי מודיעיני שזז בעולם ומאפשר למדינה לדעת יותר. מרגל חושף את כל המידע שיש במשבצות בטוח הראייה שלו (ראה: "רמת נראות של משבצות"), והוא הכלי היחיד שיכול לראות מרגלים אחרים ולווינים. מרגל לא יכול לתקוף או להגן על עצמו, ואם הוא נמצא במשבצת בה צבא זר מנצח, המרגל מת.

פקודות: לזוז.

טוח תקיפה:  -.

טוח הגנה:  -.

טוח ראייה:  1.

מהירות:  1.

מחיר:  20.





מגדלי תצפית


תיאור כללי: מגדל תצפית לא יכול לזוז, ויש לו טוח ראייה גדול בו הוא מספק נראות רגילה (ראה: "רמת נראות של משבצות"). כאשר המשבצת עליה הוא יושב נכבשת, מגדל התצפית עובר צד.

פקודות: -.

טוח תקיפה:  -.

טוח הגנה:  -.


טוח ראייה:  5.

מהירות:  -.

מחיר:  16.

לווני ריגול

תיאור כללי: לווין ריגול יכול לזוז במהירות גבוהה, ונותן נראות מלווין על המשבצות בטווח הראייה הגדול שלו (ראה: "[רמת נראות של משבצות](#)"). ניתן לראות לווין רק בעזרת מרגל. פקודות: לזוז.

טווח תקיפה:  -.

טווח הגנה:  -.


טווח ראייה:  10.

מהירות:  32.

מחיר:  64.

בנאים

תיאור כללי: בנאי מתרוצץ לו ברחבי העולם, אוסף כסף ובונה כלי מלחמה נוספים. בכל תור בנאי יכול לאסוף לכל היותר 5 כסף מהמשבצת בה הוא נמצא, ובכל רגע נתון הוא יכול להחזיק עליו עד 100 כסף. אם המשבצת שבה נמצא בנאי נכבשת, הוא עובר צד עם כל הכסף שלו. פקודות: לזוז, לקחת כסף ממשבצת, להניח כסף במשבצת, לבנות כל כלי.

טווח תקיפה:  -.

טווח הגנה:  -.

טווח ראייה:  1.

מהירות:  1.

מחיר:  20.

סיכום

8  1  1  *2  3 		8  1  1  0  0 	
16  2  1  -  1 		20  8  1  -  0 	
32  1  1  10  - 		10  1  1  0  - 	
20  1  1  -  - 		10  -  1  0  - 	
64  32  10  -  - 		16  -  5  -  - 	
		20  1  1  -  - 	

רמות נראות של משבצות

ישנן 4 רמות נראות שונות למשבצת. נתאר אותן בטבלה:

רמת נראות	מתי זה קורה?	מה אני יודע על המשבצת?
ללא נראות	המשבצת לא שייכת לי וגם אין לי כלי שרואה אותה	שייכות למדינה
נראות מלווין	המשבצת לא שייכת לי ולווין שלי רואה אותה	שייכות למדינה וכלים הנמצאים במשבצת
נראות רגילה	המשבצת שייכת לי או שיש לי כלי שאינו לווין שרואה אותה	שייכות למדינה, כלים הנמצאים במשבצת וכמות כסף
נראות שקופה	מרגל רואה את המשבצת	שייכות למדינה, כלים הנמצאים במשבצת, כמות כסף וכן לווין/מרגלים במשבצת

מהלך קרב

קרב הוא סבב לחימה במשבצת מסוימת, המתרחש בסוף תור בין כל כלי הלחימה שתוקפים את המשבצת בתור הזה לבין כל כלי הלחימה שהגנו על המשבצת בתור הזה. בקרב משתתפים בו זמנית כלים של כל המדינות התוקפות/מגנות על המשבצת.

בקרב, לכל כלי יש מצב מסוים - תקיפה, הגנה, או פאסיביות. הכלים יזווגו לקרבות אחד-על-אחד ושם ישפיעו אחד על השני לפי התיאור בטבלת הקרב. נתחיל בלתאר את המצבים השונים, ולאחר מכן את מהלך הקרב עצמו.

מצבים של כלים לוחמים

כאמור, במהלך קרב לכל כלי יש מצב מסוים. כלים תוקפים הם כלים שקיבלו באותו התור פקודת תקיפה. הם תוקפים יעד ספציפי, והם ישתתפו בקרב רק במשבצת אותה תקפו ובמצב תקיפה. אם כלי לא קיבל פקודת תקיפה בתור כלשהו, הוא יהיה דיפולטית במצב הגנה (למעט כלים שלא יכולים להגן: מטוס ומסוק). כלי במצב הגנה ישתתף בכל הקרבות בטווח ההגנה שלו.

למען הסדר הטוב והיחס החם ליוצאים מן הכלל, הנה פירוט על כל הכלים ומתי הם נמצאים בכל מצב:

- טנק יהיה במצב תוקף אם הוא קיבל פקודה לתקוף, אחרת הוא יהיה במצב מגן.
- ארטילריה תהיה במצב תוקף אם היא קיבלה פקודה לתקוף, אחרת היא מגנה ובכל מקרה ארטילריה היא פסיבית במשבצת שלה.

- מטוס ומסוק יהיו במצב תוקף אם הם באוויר וניתנה להם פקודה לתקוף, אחרת הם פאסיביים.
- נ"ט תמיד יהיה במצב מגן.
- כיפת ברזל תהיה במצב מגן כשהיא דולקת ופסיבית כשהיא כבויה.
- בונקר תמיד יהיה במצב מגן.
- מרגל, מגדל תצפית, לווין ו**בנאי** לא משתתפים בקרבות.

איך נראה קרב?

נתאר לוגית את הדרך בה מתנהל קרב במשבצת כלשהי.

שלב א': זיווג

נניח כי יש לנו קרב בין n מדינות. נבנה טבלה גדולה בה לכל מדינה יש עמודה, ובעמודה נכתוב את כל הכלים של המדינה שרלוונטיים לקרב במשבצת הנוכחית - כלומר כלים של המדינה שתוקפים את המשבצת הזאת, כלים של המדינה שמגנים על המשבצת הזאת, וכלים של המדינה שפאסיביים במשבצת הזאת (כל כלי בשורה משל עצמו, חוץ מבונקר שיופיע חמש פעמים). הסדר בו נכתוב את הכלים בעמודה הוא אקראי. כעת, כל שורה בטבלה מכילה מספר (לא בהכרח קבוע) של כלי מלחמה, לכל היותר אחד של כל מדינה. כל שורה היא זיווג של כלי מלחמה שנלחמים אחד בשני.

שלב ב': נלחמים












כל שורה בטבלה מכילה מספר כלשהו בין 1 ל- n של כלי מלחמה, שהולכים להילחם זה בזה. נריץ קרב בין כל זוג כלי מלחמה בשורה. את התוצאה של הקרב אפשר לראות בטבלת הקרב. נשמור את התוצאות של הקרב בין כל זוג, אך האפקט שלהן עוד לא קורה. כלי מלחמה המופיעים בשורה לבד לא עושים שום דבר, חוץ מאחד חשוב - אם טנק מופיע בשורה לבד והוא במצב התקפה, הוא יכבוש את המשבצת. גם זו תוצאה שנזכור ולא נבצע את האפקט שלה מיד.

שלב ג': תוצאות

כל התוצאות הנובעות מהקרבות שהיו בשלב ב' מתמשות כעת, והקרב נגמר.

מה קורה לכלים כשהמשבצת שלהם נכבשת?

נפרט פה מה קורה לסוגי כלים שונים כשהמשבצת שלהם נכבשת:

- טנק:  כלום
- ארטילריה:  מחליפה צד
- מטוס:  מחליף צד אם הוא על הקרקע וכלום אם הוא באוויר
- מסוק:  מחליף צד אם הוא על הקרקע וכלום אם הוא באוויר
- נ"ט:  מת
- כיפת ברזל:  מתה
- בונקר:  מחליף צד
- מרגל:  מת
- מגדל תצפית:  מחליף צד
- לוויין:  כלום
- בנאי:  מחליף צד עם כל הכסף שלו

פרק שני : הפעלת המשחק

ה-API-ים

הערה כללית לגבי מימוש הקוד:
כל האובייקטים נוצרים מחדש בכל תור, ולכן אם ברצונכם לשמור מידע בין תור לתור, ניתן לעשות זאת באמצעות משתנים גלובליים.

ה-API הטקטי

ה-API הטקטי מוגדר בתיקיה `info` בקובץ `tactical_api.txt` וניתן לראות אותו בצורה נחמדה אם פותחים את הקובץ `tactical_api.m.html` בדפדפן (גם הקובץ הזה נמצא ב-`info`). דוגמת שימוש ב-API הטקטי נמצאת בתיקיה `code` בקובץ `tactical_example_with_all_commands.py`.
המשחק מתנהל בתורות כשלכל כלי מותר לבצע פעולה אחת מה-API הטקטי בכל תור.

המחלקות ב-API הטקטי

ה-API הטקטי מספק מחלקה לכל סוג כלי, כאשר לכל מחלקה יש פונקציות המתאימות לסוג הכלי. המחלקות הללו יורשות ממחלקת הבסיס `BasePiece`, לפי ההיררכיה הבאה:

- `BasePiece`
 - `Antitank` (נ"ט)
 - `Artillery` (ארטילריה)
 - `Builder` (בנאי)
 - `Bunker` (בונקר)
 - `FlyingPiece`
 - `Airplane` (מטוס)
 - `Helicopter` (מסוק)
 - `IronDome` (כיפת ברזל)
 - `Satelite` (לוויין)
 - `Spy` (מרגל)
 - `Tank` (טנק)
 - `Tower` (מגדל)

התכונות המשותפות לכל הכלים שמוגדרות במחלקת `BasePiece`:

- הפונקציה `move` שמזיזה כלי ממקום למקום.
- השדה `country` המציין את המדינה אליה שייך הכלי.
- השדה `id` שהוא מזהה הכלי במערכת.
- השדה `tile` המציין את מיקום הכלי (הסבר בהמשך).
- השדה `type`, המכיל מחרוזת עם סוג הכלי.

- התכונות המשותפות לכל הכלים המעופפים, שמוגדרות במחלקה `FlyingPiece`:
- הפונקציות `take_off` ו-`land`, שמטרתן להורות לכלי להמריא ולנחות, בהתאמה.
 - השדה `in_air`, שערכו `True` אם ורק אם הכלי כרגע באוויר.
 - השדה `time_in_air`, שערכו מספר התורות מאז שהכלי היה על הקרקע בפעם האחרונה.

ה-API הטקטי מייצג משבצות משחק באמצעות המחלקה `Tile`, שחושפת את השדות הבאים:

- השדה `coordinates`, שמייצג את הקואורדינטות של המשבצת. לשדה יש תת שדות בשמות `x` ו-`y`.
- השדה `country` המציין את המדינה אליה שייכת המשבצת.
- השדה `money` המכיל את כמות הכסף במשבצת.
- השדה `pieces` שהוא רשימה של הכלים הנמצאים במשבצת כרגע, ככל הידוע למדינה.

מחלקה חשובה נוספת ב-API הטקטי היא `TurnContext`, המספקת מידע על התור הנוכחי. אובייקטים מסוג `TurnContext` נבנים על ידי מנוע המשחק ולא על ידי השחקנים עצמם.

המחלקה חושפת את השדות הבאים:

- השדה `tiles`: מפה מקואורדינטה (x, y) לאובייקט מסוג `Tile`, לפי המצב הידוע למדינה זו בתחילת התור הנוכחי.
- השדה `my_pieces`: מפה הממפה את אובייקטי הכלים של מדינה זו לפי מזהה הכלי.
- השדה `all_pieces`: מפה הממפה את אובייקטי כל הכלים שמדינה זו יודעת על קיומם לפי מזהה הכלי.
- השדה `game_width`: רוחב לוח המשחק.
- השדה `game_height`: גובה לוח המשחק.
- השדה `my_country`: השם של מדינה זו.
- השדה `all_countrys`: רשימת השמות של כל המדינות המשתתפות במשחק.

כמו כן, המחלקה מגדירה מספר פונקציות שימושיות. כדאי לעיין במדריך המלא של ה-API.

ה-API האסטרטגי

ה-API האסטרטגי הוא ממשק התקשורת בין הצוות האסטרטגי לצוות הטקטי. הצוות האסטרטגי משתמש בו כדי לתת הוראות לצוות הטקטי. הצוות הטקטי מקבל הוראות כקריאות ל-API, ומבצע את ההוראות בצורה של מימוש ה-API.

ה-API האסטרטגי מוגדר בתיקייה `info` בקובץ `strategic_api.txt` וניתן לראות אותו בצורה נחמדה אם פותחים את הקובץ `strategic_api.m.html` בדפדפן (הנמצא באותה תיקייה).

שימו לב: לא ניתן לבצע שינויים ב-API הזה!

המחלקות ב-API האסטרטגי

- ה-API האסטרטגי מגדיר שתי מחלקות להעברת מידע בין הצוותים:
- המחלקה `StrategicPiece`, שהיא ייצוג מצומצם יותר של כלי משחק, המכילה שני שדות: `id` (מזהה) ו-`type` (סוג).
 - המחלקה `CommandStatus`, שמיועדת להעברת מידע אודות ביצוע הוראה מהצוות הטקטי לצוות האסטרטגי.
- אובייקטים של המחקה בונים באמצעות אחת מהדרכים הבאות:
- כדי לייצג הוראה שנמצאת כרגע בביצוע, יש להשתמש בפונקציה `CommandStatus.in_progress` ולהעביר אליה את מזהה ההוראה, מספר התורות שחלפו מאז שניתנה ההוראה, והערכה לכמות התורות הנותרים להשלמת ביצוע ההוראה.
 - כדי לייצג הוראה שביצועה הושלם בהצלחה, יש להשתמש בפונקציה `CommandStatus.success`, ולהעביר אליה את מזהה ההוראה.
 - כדי לייצג הוראה שביצועה נכשל, יש להשתמש בפונקציה `CommandStatus.failed`, ולהעביר אליה את מזהה ההוראה.
- ה-API האסטרטגי עצמו מוגדר במחלקה `StrategicApi`. מומלץ בחום לעיין בפונקציות המוגדרות בו. אין חובה לממש את כל הפונקציות של ה-API האסטרטגי. אולם אם נעשה שימוש בפונקציה שאינה ממומשת, מדינה זו לא תבצע כל מהלך בתור הנוכחי.

מימוש ה-API האסטרטגי

הקובץ שמממש את הפונקציות של ה-API האסטרטגי יקרא הקובץ הטקטי. הקובץ הטקטי חייב להכיל פונקציה בשם `get_strategic_implementation`, שמקבלת ארגומנט יחיד - אובייקט `TurnContext` שבאמצעותו ניתן לבצע פעולות טקטיות, ומחזירה אובייקט שמממש את ה-API האסטרטגי (כלומר אובייקט של מחלקה שיורשת מ-`StrategicApi`). הפונקציה הזו נקראת בכל תור מחדש, והיא אמורה לקבל הוראות באמצעות ה-API האסטרטגי, ולתרגמן לפקודות של ה-API הטקטי. שימו לב: אם תקראו לפקודות ב-API הטקטי עם פרמטרים שאינם מתאימים, הפעולה לא תבצע. הערה חשובה: כל כלי יכול לקבל פקודה אחת בתור. אם תתנו לכלי כמה פקודות באותו תור, ההתנהגות איננה מוגדרת.

שימוש ב-API האסטרטגי

הקובץ שבו כותבים את הבוט יקרא הקובץ האסטרטגי.
הקובץ האסטרטגי לא יכול להשתמש בפונקציות מה-API הטקטי אלא רק מה-API האסטרטגי (אל תהיו ילדותיים ותנסו לעקוף את החוק הזה, זה הורס את מטרת התרגיל והורס את הכיף לכולם).
הקובץ האסטרטגי חייב להכיל פונקציה בשם `do_turn`, שמקבלת ארגומנט יחיד - אובייקט שמממש את ה-API האסטרטגי (זה יהיה הקוד שממומש בקובץ הטקטי).
הפונקציה `do_turn` נקראת בכל תור מחדש, והיא אמורה לבצע פעולות באמצעות ה-API האסטרטגי.

משהו בסיסי לדוגמה

ניקח לדוגמה את האסטרטגיה הבאה:
בכל תור, ננסה לכבוש שטח אויב. אם אין שטח אויב, ננסה לכבוש שטח פנוי.

כדי לממש את האסטרטגיה הזאת, יש צורך להשתמש בפונקציות של ה-API האסטרטגי: (ניתן לקרוא עליהן בעיון במסמך התיעוד שלו)

- `attack`, באמצעותה הצוות האסטרטגי מורה לצוות הטקטי איזו משבצת לתקוף, ובאמצעות איזה כלי.
- `estimate_tile_danger`, באמצעותה הצוות הטקטי מדווח לצוות האסטרטגי מה מצבה של משבצת מסוימת.
- `get_game_height` ו-`get_game_width`, באמצעותן הצוות הטקטי מדווח לצוות האסטרטגי מה גודלו של לוח המשחק.
- `report_attacking_pieces`, באמצעותה הצוות הטקטי מדווח לצוות האסטרטגי מה מצבו של כל כלי תוקף.

את הקוד האסטרטגי אנחנו נכתוב בקובץ פיתוח ייעודי. בהתאם לדרישות המערכת, נממש בו פונקציה בשם `do_turn`, שמקבלת ארגומנט יחיד - אובייקט המממש את ה-API האסטרטגי. נשתמש באובייקט הזה כדי להעביר הוראות לצוות הטקטי.

ראשית, הצוות האסטרטגי נדרש למפות את העולם בעזרת הצוות הטקטי. לכן, הצוות האסטרטגי נדרש לדעת מה גודל לוח המשחק, ומה מצבה של כל משבצת. לאחר מכן, נעדיף משבצות אויב, ורק אז משבצות פנויות. בתוך כל קטגוריה, נבחר משבצות באופן אקראי. את המשימה הזאת נעשה בפונקציה שנקרא לה `get_sorted_tiles_for_attack`.

כעת, כשיש לנו את מפת לוח המשחק, נבדוק מה מצב הכלים שלנו (לפי הדיווח של הקוד הטקטי), ונשלח כלים לתקיפה לפי סדר המשבצות שקיבלנו מ-`get_sorted_tiles_for_attack`.
המימוש המלא של הקוד האסטרטגי נמצא בקובץ `simple_strategic.py` שבתיקייה `.code`.

את הקוד הטקטי אנחנו נכתוב בקובץ פייתון ייעודי. בהתאם לדרישות המערכת, כל מודול טקטי צריך להכיל את ההגדרות הבאות:

- מחלקה היורשת מ-StrategicApi ומממשת את הפונקציות הנדרשות ממנו. נקרא לה MyStrategicApi בדוגמה זו.
- פונקציה שנקראת get_strategic_implementation, המקבלת ארגומנט יחיד - context מסוג TurnContext, ומחזירה אובייקט מהמחלקה שהגדרנו לעיל. פונקציה זו תיקרא בכל תור מחדש.

לצורך ההסבר, נתמקד רק במימוש הבנאי של MyStrategicApi מה מעניין בבנאי? קודם כל, החתימה שלו. חשוב לשים לב שהחתימה שלו, וכן השורה הראשונה (שקוראת לבנאי של מחלקת הבסיס) יהיו בדיוק כך:

```
def __init__(self, *args, **kwargs):
    super(MyStrategicApi, self).__init__(*args, **kwargs)
```

זה יאפשר לנו לגשת מאוחר יותר ל-TurnContext בתור self.context. דבר נוסף שמעניין בבנאי הוא העובדה שהבנאי נקרא בתחילת כל תור, ללא קריאה מיוחדת מצד הצוות האסטרטגי. זה מאפשר לנו להמשיך לבצע פקודות אסטרטגיות שניתנו לנו בתורות קודמים באופן עצמאי.

איך עושים את זה? המידע שמעניין אותנו מהתורות הקודמים שמור במשתנים גלובליים. במקרה שלנו, מדובר במשתנים tank_to_coordinate_to_attack, tank_to_attacking_command ו-tank_to_attacking_command.

המימוש המלא של הקוד הטקטי נמצא בקובץ simple_tactical.py שבתיקייה .code

הערכה למשתתף המתחיל

מה קיבלתם ומה עוד תקבלו?

קיבלתם

קיבלתם תיקייה עם הדברים הבאים:

- קובץ הוראות (הקובץ הזה)
- קובץ טבלת קרב (נספח של קובץ ההוראות הזה)
- קובץ להעלאה מהירה של גרסאות קוד (בהמשך נסביר [איך משתמשים בו](#))
- תיקיית code שמכילה דוגמאות קוד שימושיות
- תיקיית info שמכילה את ה-API-ים

תקבלו

כתובת url לשרת ההרצה של המשחק וסיסמה איתה תתחברו לשרת.

שרת ההרצה של המשחק - הסבר כללי

- אחרי שתתחברו לשרת תוכלו לראות שהאתר בנוי מארבעה חלקים עיקריים:
1. **Code Versions** - החלק שבו אתם מנהלים את את גרסאות הקוד שלכם.
 2. **Bots** - החלק בו מופיעים הבוטים בהם אתם יכולים להשתמש לאימון.
 3. **Runs** - החלק שבו אתם מריצים וצופים במשחקים בודדים.
 4. **Tournaments** - החלק שבו אתם מריצים וצופים בטורנירים.
- ככה נראה העמוד הראשי של האתר:


PyWar

Welcome to pyWar!

Run named a deleted successfully

Code Versions

No code has been uploaded yet.

 [Upload new](#)

Bots

Bots are builtin codes that cannot be removed, but can be ran.

Advanced

Empty

Simple

Runs

New run

city

-

2-players

Submit

Existing runs

No runs are available.

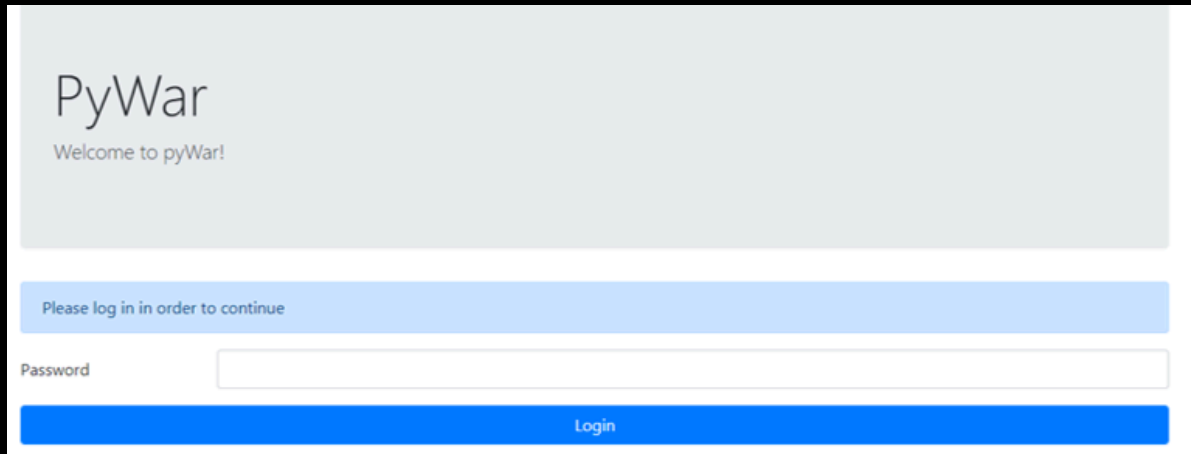
Tournaments

No tournaments are available.

[New tournament](#)

חיבור לשרת

כדי להתחבר לשרת, פתחו דפדפן אינטרנט וגלשו ל-url שניתן לכם.
כשתתחברו לשרת בפעם הראשונה אתם תראו את העמוד הבא:

A screenshot of the PyWar login page. At the top, it says "PyWar" in a large font, followed by "Welcome to pyWar!" in a smaller font. Below this is a light blue box with the text "Please log in in order to continue". Underneath that is a label "Password" next to a text input field. At the bottom is a blue button with the text "Login".

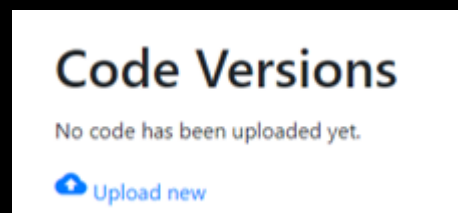
בעמוד זה תצטרכו להכניס את הסיסמה לשרת בתיבה ליד המילה "Password" וללחוץ על הכפתור הכחול "Login".

שרת ההרצה - Code Versions

מידע כללי

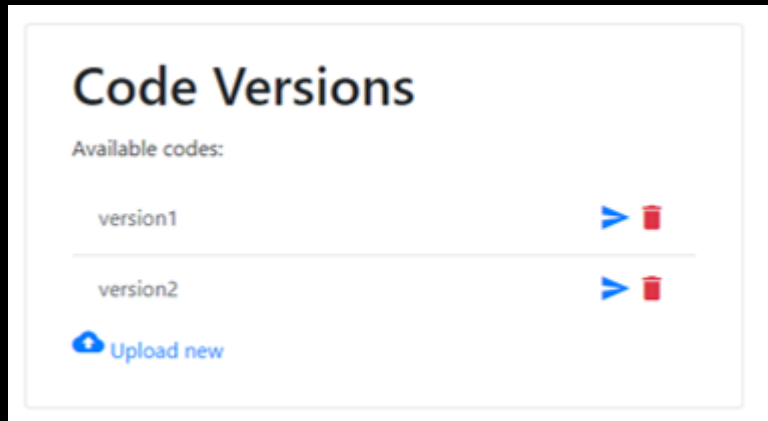
בחלק זה של האתר אתם יכולים להעלות גרסאות קוד חדשות, לבחור את הגרסה אותה תרצו להגיש ולמחוק גרסאות מיותרות.

כשאין גרסאות קוד



בהתחלה יהיה כתוב בחלק זה "No code has been uploaded yet", זה אומר שאין לכם גרסאות קוד שלכם. בשביל להעלות קוד חדש, לחצו על "Upload new" כפי שמופיע בתמונה למעלה.

כשיש גרסאות קוד



אחרי שהעלתם גרסאות קוד, הן יופיעו ברשימה של "Available codes". שימו לב שאתם יכולים להמשיך ולהעלות עוד גרסאות בעזרת "Upload new". על כל גרסת קוד יש שתי פעולות שאתם יכולים לעשות:

1. לסמן אותה כגרסה הכי עדכנית (זאת שתוגש אוטומטית לריצה הבאה)
2. למחוק את גרסת הקוד.

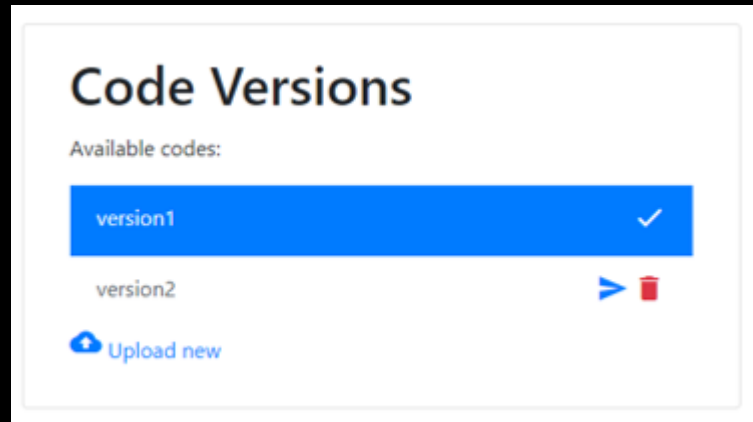
סימון הקוד הראשי

בשביל לסמן את גרסת הקוד הראשית שלכם (שתשתף בתחרות הקרובה) לחצו על החץ הכחול שליד הגרסה אותה תרצו להגיש. שימו לב שתוכלו לשנות את הגרסה הראשית שלכם על ידי לחיצה על החץ הכחול ליד גרסה אחרת. כדאי לכם שבכל שלב תהיה לכם גרסה ראשית כדי שתמיד ירוץ קוד כלשהו שלכם בתחרות.

כאשר תסמנו גרסת קוד בהצלחה תופיע ההודעה הבאה:

Code named version1 is now submitted

והקוד המסומן שלכם יופיע מסומן:



מחיקת גרסת קוד

בשביל למחוק חרסת קוד לחצו על הפח האדום שליד הגרסה אותה תרצו למחוק. כאשר תמחקו גרסת קוד בהצלחה תופיע ההודעה הבאה:

Code named version2 deleted successfully

שימו לב שאתם לא יכולים למחוק את גרסת הקוד שכרגע מסומנת.

העלאת גרסת קוד חדשה

ישנן שתי דרכים להעלות גרסת קוד חדשה:

1. דרך האתר

2. דרך הסקריפט `upload_script.py`

העלאה בעזרת האתר

אפשר להעלות גרסת קוד חדשה על ידי לחיצה על הכפתור "Upload new". לאחר מכן יפתח לכם העמוד הבא:

[Main page](#) > Upload code

Upload Code

Code name

version1

Tactical module

tactical

The tactical implementation module name should not contain the file extension (like `.py`).

Strategic module

strategic

The strategic implementation module name should not contain the file extension (like `.py`).

Code tarball

Choose File

all_my_scripts.zip

The code in the provided tarball should not be packed under a subdirectory.

☐ Overwrite existing code

Submit

בעמוד זה תצטרכו להכניס את הפרטים הבאים:

- Code name - שם הגרסה שאתם מעלים (בדוגמה: version1)
- Tactical module - שם הקובץ הטקטי בלי ה-py. (בדוגמה: tactical)
- Strategic module - שם הקובץ האסטרטגי בלי ה-py. (בדוגמה: strategic)
- קובץ tarball שהוא (.tar.gz) או קובץ zip. המכיל את כל קבצי הקוד שלכם (בדוגמה: all_my_scripts.zip)
- סמנו את הקופסה למטה רק אם אתם רוצים לדרוס גרסה עם אותו שם.

כשסיימתם להכניס את כל הפרטים ואת הקובץ, לחצו על הכפתור הכחול של "Submit" והקוד יעלה.

אם גרסת הקוד עלתה, תקבלו את ההודעה הבאה:

Code uploaded successfully

העלאה בעזרת upload_script.py

בשביל להעלות קבצים עם הסקריפט, פתחו shell או cmd בתיקיה והריצו את הסקריפט upload_script.py עם הדגלים הבאים:

1. --directory DIR כאשר DIR זה שם התיקיה של הקוד.
2. --tactical-module TACT כאשר TACT זה שם הקובץ הטקטי.
3. --strategic-module STRAT כאשר STRAT זה שם הקובץ האסטרטגי.
4. --server IP כאשר IP זה ה-IP של השרת שלכם.
5. --port PORT כאשר PORT זה מספר הפורט של השרת שלכם.
6. --name NAME כאשר NAME זה שם גרסת הקוד שאתם מעלים.
7. --password PASS כאשר PASS זה הסיסמה לשרת שלכם.

דוגמה לשורה תקינה:

```
python upload_script.py --directory code --tactical-module empty_tactical --strategic-module empty_strategic --server 127.0.0.1 --port 57548 --name version1 --password "(5BD0FJE1D40,W08<$fH"
```

אם הקוד עלה אתם אמורים לקבל את התשובה Success.

שרת ההרצה - Bots

מידע כללי

בחלק זה של האתר מופיעה לכם רשימת הבוטים שאתם יכולים להתמודד מולם בהרצות שלכם.

בשלב זה יש לכם 3 בוטים להתמודד מולם:

1. Empty - בוט שלא עושה כלום.
2. Simple - בוט שמקיים את האסטרטגיה בקבצי הדוגמא simple שקיבלתם.
3. Advanced - בוט עם אסטרטגיה טובה יותר מאשר Simple.

החלק של Bots באתר נראה כך:

Bots

Bots are builtin codes that cannot be removed, but can be ran.

Advanced

Empty

Simple

שרת ההרצה - Runs

מידע כללי

בחלק זה של האתר אתם יכולים להריץ משחקים בודדים ולצפות במשחקים שהרצתם ובתוצאות שלהם.

כשאינן הרצות

כאשר אין לכם הרצות לצפות בהן החלק של Runs יראה כך:

Runs

New run

city - 2-players Submit

Existing runs

No runs are available.

בשביל ליצור הרצה חדשה, בחרו את סוג המפה שאתם רוצים להריץ ואת כמות השחקנים ולחצו על הכפתור הכחול של "Submit".

כשיש הרצות

Runs

New run

city

 -

2-players

Submit




Existing runs

run1

Wed, 07 Oct 2020 13:14:25

100 turns on city-4-players.

Winner: **berzerkistan**

כאשר יש הרצות, אתם תראו אותם בחלק זה. על כל הרצה אתם תוכלו לראות את הפרטים הבאים:

- שם ההרצה (בדוגמה: run1)
- הזמן שבו היא התרחשה
- כמות התורות של ההרצה (בדוגמה: 100)
- המפה בה התרחשה ההרצה (בדוגמה: city)
- כמות השחקנים (בדוגמה: 4)
- המנצח של ההרצה (בדוגמה: berzerkistan)

עבור כל הרצה תוכלו לבצע את הפעולות הבאות:

1. לצפות בפרטי ההרצה – על ידי לחיצה על כפתור העין הכחול
2. לצפות בהרצה עצמה – על ידי לחיצה על כפתור המשולש הכחול (play)
3. למחוק את ההרצה – על ידי לחיצה על כפתור הפח האדום

שימו לב שאתם עדיין יכולים להריץ עוד משחקים על ידי התהליך שראינו קודם.

יצירת הרצה חדשה (אחרי בחירת מפה ושחקנים)

אחרי שבחרתם מפה וכמות שחקנים ולחצתם על "Submit" יפתח לכם העמוד הבא:

[Main page](#) > [New run](#)

New Run

Run name	<input type="text" value="run1"/>
Map name	city-4-players
absurdistan	<input type="text" value="version1"/>
berzerkistan	<input type="text" value="Advanced"/>
cobrastan	<input type="text" value="Empty"/>
derkaderstan	<input type="text" value="Simple"/>
Max game turns	<input type="text" value="100"/>
Max game time	<input type="text" value="120"/>

Submit

בו תצטרכו להכניס את הפרטים הבאים:

- שם ההרצה (בדוגמה: run1)
- לבחור גרסת קוד או בוט לכל אחת מהמדינות המשתתפות
- מספר תורות מקסימלי (ברירת המחדל: 100)
- מספר השניות המקסימלי שיקח המשחק (ברירת המחדל: 120)

לאחר שמילאתם את כל הדברים לחצו על הכפתור הכחול "Submit" והריצה תתחיל.

כשהריצה תסיום, תחזרו לעמוד הראשי ותקבלו את ההודעה הבאה:

Run completed successfully

תוכלו לראות את הריצה בחלק של Runs.

צפייה במידע על הרצה

בשביל לצפות במידע על הרצה לחצו על כפתור העין הכחול שליד ההרצה ואז יפתח לכם העמוד הבא:

[Main page](#) > [Tournament details](#)

Run run1

Basic Information

Run time: Wed, 07 Oct 2020 13:14:25

Map: city-4-players

Max turns: 100

Countries

Code version1 as **absurdistan**

Code Advanced as **berzerkistan**

Code Empty as **cobrastan**

Code Simple as **derkaderstan**

Actions

[▶ Play](#) [🗑 Delete run](#)

Results

Winner: **berzerkistan**

Actual turns: 100

Termination reason: Game over due to turns limit

Game duration: 11.67625641822815 seconds

Statistics

berzerkistan has 285 tiles and 65 pieces.

derkaderstan has 20 tiles and 4 pieces.

cobrastan has 11 tiles and 4 pieces.

absurdistan has 10 tiles and 4 pieces.

Logs

למעלה בגדול יופיע שם ההרצה (בדוגמה: run1) בחלק Basic Information יופיע מידע בסיסי על ההרצה. בחלק Countries יופיע באיזה גרסת קוד או בוט השתמשה כל מדינה. בחלק Actions תוכלו לצפות בהרצה עצמה או למחוק את ההרצה. בחלק Results אתם תראו את המנצח, כמות התורות בפועל, סיבת הסיום של המשחק והזמן בפועל של המשחק. בחלק Statistics תוכלו לראות כמה משבצות וכמה כלים היו לכל מדינה בסוף המשחק. למטה יופיעו הלוגים של כל מדינה תחת החלק Logs:

Logs

absurdist an

No logs for absurdistan

berzerkist an

No logs for berzerkistan

cobrastan

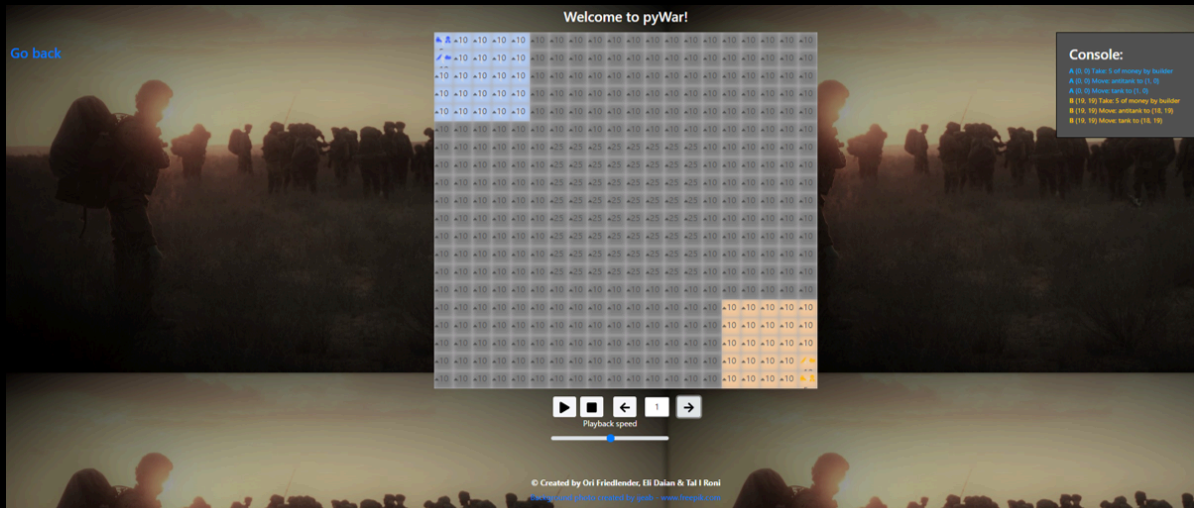
No logs for cobrastan

derkaders tan

No logs for derkaderstan

צפייה בהרצה עצמה

בשביל לצפות בהרצה עצמה לחצו על כפתור המשולש הכחול (play) ויפתח לכם הדף הבא:



במסך זה תוכלו לצפות בהרצה עצמה, בכלים זזים ובכסף על המשבצות (עד רמה מסוימת) ובפקודות הניתנות לכלים.

- באמצע אתם יכולים לראות את הלוח עצמו, בכלים, בשטח של כל מדינה ובכסף שיש על המשבצות.
- מתחת ללוח יש כפתורים:

- כפתור play כדי להריץ את המשחק
- כפתור stop כדי לעצור על תור מסוים
- כפתורים ימינה ושמאלה כדי להעביר לתור הבא והקודם בהתאמה
- בין הכפתורים ימינה ושמאלה יש את ספר התור הנוכחי
- מתחת לכפתורים יש סקאלת מהירות של תור (ימין - מהר, שמאל - לאט)
- משמאל ללוח יש כפתור שיחזיר אתכם לצפייה במידע על ההרצה
- מימין ללוח יש את ה-Console שבו ניתן לראות את כל הפקודות שכל מדינה שלחה בתור הזה

מחיקת הרצה

כאשר אתם רוצים למחוק הרצה, לחצו על כפתור הפח האדום שליד ההרצה. לאחר מכן אתם אמורים לקבל את ההודעה הבאה:

Run named run1 deleted successfully

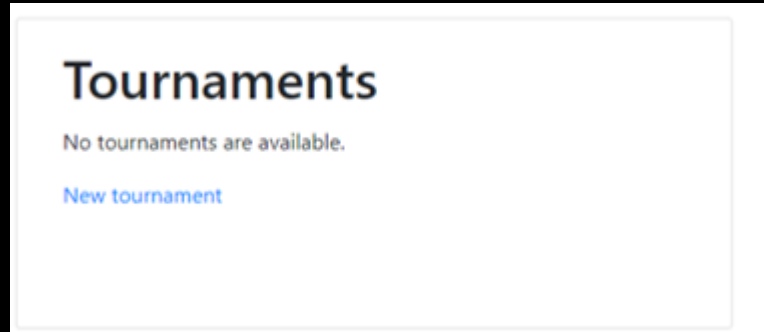
שרת ההרצה - Tournaments

מידע כללי

בחלק זה של האתר אתם יכולים להריץ טורנירים ולצפות בהם.

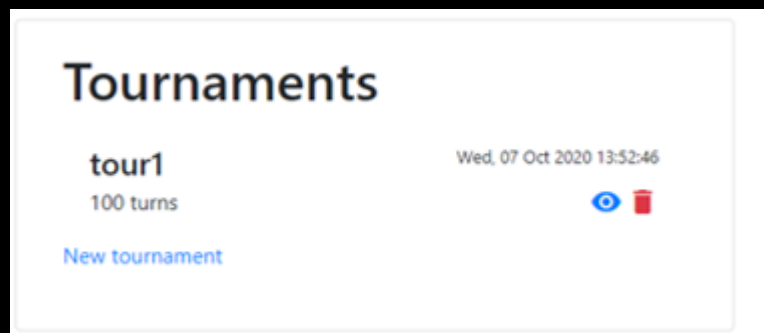
כשאתם טורנירים

כאשר אין לכם טורנירים לצפות בהם החלק של Tournaments יראה כך:



יהיה כתוב "No tournaments are available" ובשביל ליצור טורניר חדש לחצו על "New tournament".

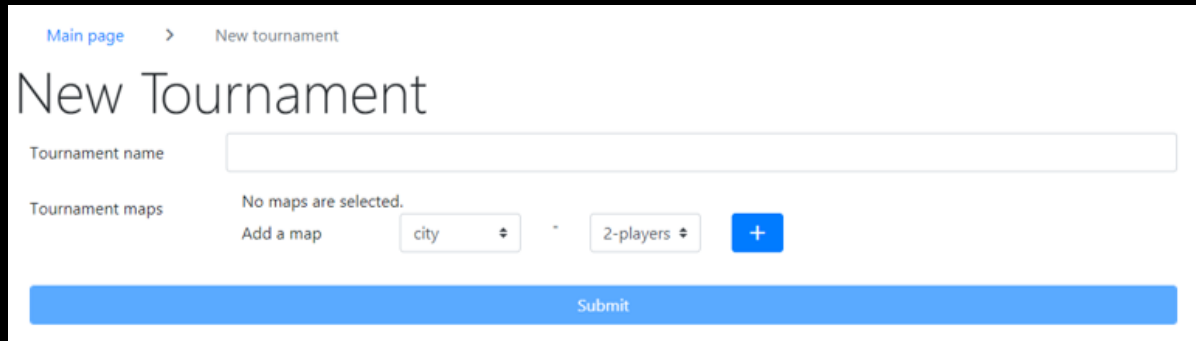
כשיש טורנירים



כאשר יש טורנירים, אתם תראו אותם בחלק זה. עבור כל טורניר אתם תראו את השם שלו, את הזמן בו הוא התרחש, את כמות התורות המקסימלית שלו ותוכלו לצפות במידע עליו ולמחוק אותו. שימו לב שגם כאשר יש טורנירים, תוכלו ללחוץ על "New tournament" בשביל ליצור טורניר חדש.

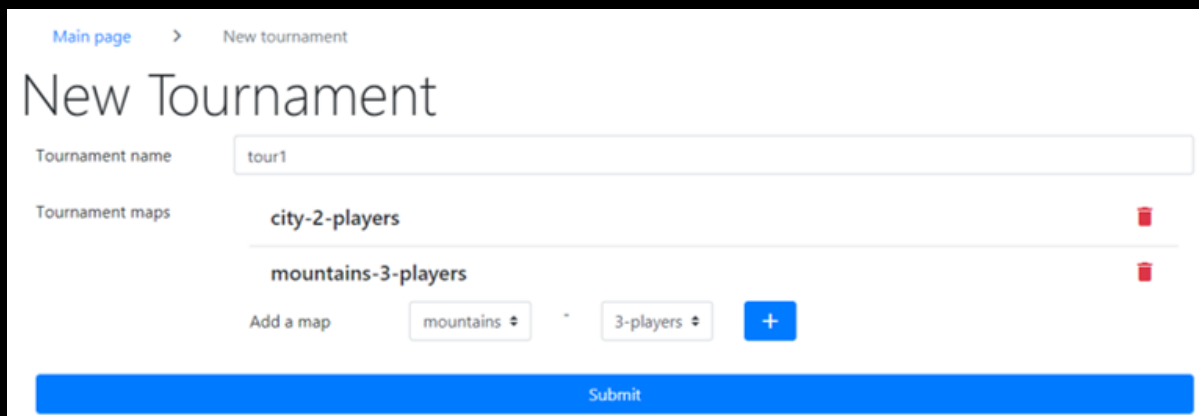
יצירת טורניר חדש

כדי ליצור טורניר חדש לחצו על "New tournament" ואז יפתח לכם העמוד הבא:



The screenshot shows the 'New Tournament' page. At the top, there are two links: 'Main page' and 'New tournament'. The title 'New Tournament' is prominently displayed. Below it, there is a text input field for 'Tournament name'. Underneath, the 'Tournament maps' section shows 'No maps are selected.' and an 'Add a map' button. To the right of this button are two dropdown menus: one set to 'city' and another to '2-players', followed by a blue '+' button. At the bottom of the form is a wide blue 'Submit' button.

בו תכתבו את שם הטורניר (לדוגמה: tour1) ותבחרו את המפות שישתתפו בטורניר. מוסיפים מפות על ידי בחירת המקום (לדוגמה: city) וכמות שחקנים (לדוגמה: 2) ולחיצה על הכפתור הכחול +. לאחר שהוספתם מפות אתם יכולים להוריד אותן על ידי לחיצה על כפתור הפח האדום שליד כל מפה:



This screenshot shows the 'New Tournament' page after some input. The 'Tournament name' field now contains 'tour1'. In the 'Tournament maps' section, two maps have been added: 'city-2-players' and 'mountains-3-players'. Each map entry has a red trash icon to its right. Below these entries, the 'Add a map' section shows dropdown menus for 'mountains' and '3-players', along with the blue '+' button. The blue 'Submit' button remains at the bottom.

לאחר שסיימתם לבחור את שם הטורניר ואת המפות לחצו על הכפתור הכחול "Submit" ויפתח לכם העמוד הבא:

[Main page](#) > [New tournament](#)

New Tournament

Tournament name

tour1

Maps

city-2-players

mountains-3-players

absurdistan

Advanced

berzerkistan

Empty

cobrastan

Simple

Max game turns

100

Max game time

120

Start

עמוד זה דומה לעמוד של יצירת הרצה חדשה מלבד שכאן מופיעות כמה מפות.
את העמוד הזה מלאו כמו [ביצירת הרצה חדשה](#) ולחצו על הכפתור הכחול
"Start" להתחלת הטורניר.
שהטורניר יגמר אתם תעברו לעמוד הראשי ותקבלו את ההודעה:

Run completed successfully

צפייה במידע על טורניר

בשביל לצפות במידע על טורניר לחצו על כפתור העין הכחול שמופיע ליד
הטורניר ויפתח לכם העמוד הבא:

[Main page](#) > [Tournament tour1 details](#)

Tournament tour1

Basic Information

Tournament time: Thu, 08 Oct 2020 17:17:46

Max turns: 100

Maps

city-2-players

Winner: [absurdistan](#)

mountains-3-players

Winner: [absurdistan](#)

Countries

Code Advanced as [absurdistan](#)

Code Empty as [berzerkistan](#)

Code Simple as [cobrastan](#)

Actions

Delete tournament

למעלה יופיע בגדול שם הטורניר.

בחלק Basic Information יופיעו הזמן בו התרחש הטורניר וכמות התורות המקסימלית.

בחלק Countries יופיע באיזה גרסת קוד או בוט השתמשה כל מדינה. בחלק Maps תוכלו לראות את המפות של הטורניר ומי היה המנצח בכל מפה. בחלק Actions תוכלו למחוק את הטורניר. ליד כל מפה בחלק Maps יש שני כפתורים; כפתור כחול של עין כדי [לצפות במידע של ההרצה](#) של המפה הזאת וכפתור של חץ כחול (play) כדי [לצפות בהרצה עצמה](#).

מחיקת טורניר

בשביל למחוק את הטורניר, לחצו על כפתור הפח האדום ליד הטורניר. לאחר מכן אתם אומרים לקבל את ההודעה הבאה:

Tournament named tour1 deleted successfully

לוגים וטיפול בשגיאות

שגיאות נפוצות

כאשר אתם הולכים לכתוב את הבוטים אתם תיתקלו בשלושה סוגים עיקריים של שגיאות:

1. שגיאות סינטקס
2. הקוד שלכם יקרוס באמצע ריצה
3. הקוד שלכם עשה מהלך לא חוקי

תוצאות השגיאות

1. במידה והייתה לכם שגיאת סינטקס, המדינה מפסידה אוטומטית בסיבוב.
2. אם הקוד קרס במהלך ריצה, המדינה לא תבצע פעולות באותו התור.
3. כאשר מדינה מנסה לבצע שתי פעולות עם אותו כלי בתור אחד או מנסה לבצע פעולה בלתי חוקית (כמו תזוזה מחוץ לטווח) ההתנהגות אינה מוגדרת (ככל הנראה, הכלי לא יבצע אף פעולה).

הדפסה לקבצי הלוג

במהלך ריצה, תוכלו להדפיס מחרוזות לקובץ הלוג באמצעות הפקודה `log()`. שימו לב שכרגע ניתן לרשום ללוג רק דרך ה-API הטקטי ולכן אנחנו מאוד ממליצים לממש במהרה את הפונקציה `log()` ב-API האסטרטגי.