

חוקי המשחק בקצרה:

**מטרת המשחק**

* מטרת המשחק היא להגיע ל 1000 נקודות לפני הקבוצה השניה
* האורך המירבי של משחק הוא 1000 תורות
* אם עברו 1000 תורות, הקבוצה מנצחת היא הקבוצה שיש לה יותר נקודות

**מה במשחק?**

* לכל שחקן יש ספינות פיראטים בצבע של אותה קבוצה
* על המפה יש איים. ספינה יכולה לכבוש אי. אי יכול להיות נטרלי, או כבוש על ידי אחד השחקנים

**הפיראטים:**

אין אפשרות להשיג עוד ספינות. לכל שחקן מספר קבוע של ספינות אך ספינה עלולה לשקוע ואם כן תהיה  מחוץ למשחק למשך 40 תורות. ספינה כזו נקראת ספינה אבודה. כשספינה חוזרת למשחק היא תחזור לנקודה בה התחילה בתחילת המשחק.

**תזוזה:**

המשחק מתנהל בתורות, כאשר בכל תור שני השחקנים יכולים להזיז כל אחת מהספינות שלהם משבצת אחת בדיוק. ספינה יכול לזוז בכל תור לאחד מארבעת הכיוונים צפונה/דרומה/מזרח/מערב. בכל משבצת יכולה להיות אך ורק ספינה אחת. לא ניתן לנוע אל מחוץ לגבולות המפה.

**צבירת נקודות**

* צבירת נקודות נעשת על ידי שליטה באיים.
* מספר הנקודות הנצברות בכל תור גדל ככל ששולטים ביותר איים באופן הבא:
  + שליטה באי אחד - 1 נקודות בכל תור
  + שליטה בשני איים - 2 נקודות בכל תור
  + שליטה בשלושה איים - 4 נקודות בכל תור
  + שליטה בארבעה איים - 8 נקודות בכל תור
  + שליטה בחמישה איים - 16 נקודות בכל תור

**כיבוש איים**

* הדרך לכבוש אי היא לעמוד על המשבצת של האי עם ספינה באופן רצוף למשך 20 תורות.
* לאחר שהספינה סיימה לכבוש את אי, האי שייך לקבוצה (הספינה יכולה לעזוב את האי) ומקנה לקבוצתה נקודות, עד שהקבוצה המתחרה תחזיר אותה לבעלות ניטרלית
* במידה והאי כבר כבוש על ידי הקבוצה השניה, יש להחזיר קודם את האי לבעלות ניטרלית לפני שהוא עובר לבעלותו של השחקן הכובש. כדי להחזיר לשייכות נטרלית ספינת אויב צריכה לעגון באי גם כן למשך 20 תורות
* לכן למעשה כדי לכבוש אי בבעלות הקבוצה השניה צריך לעגון בו למשך 40 תורות בסה״כ
* על אי יכולה לעגון רק ספינה אחת בתור נתון

**איזור מבטחים**

הספינות של כל שחקן מתחילות באיזור בטוח הנקרא zone אליו השחקן השני לא יכול להיכנס. יפורט על כך בהמשך.

**מתי ספינות שוקעות?**

כשספינה מתקרבת לספינה של הקבוצה השניה היא תנסה לגרום לספינה של הקבוצה היריבה לשקוע. בכדי להתגונן מפני זה, על הספינות החברות להסתובב כקבוצה. כשיש ברשותכם קבוצה של ספינות חברות, הן מגינות אחת על השניה מפני ספינות עוינות. להלן סט מצבים והתוצאה של כל אחת מהן:

|  |  |
| --- | --- |
| אחת על אחת  https://lh3.googleusercontent.com/EGY_7kolsTLahW6U1ileZDHtAxxeHXUwTLR4itbQfxj_7RhKfTwnFFY9zwfMa0lSNjDBEXIHkSKFaGIEodHJTubFEwiM__FQo4ZFW-D4USRAIe8qYT9K0jouzwZBDRoSmQ | שתיים על אחת  https://docs.google.com/drawings/d/s-2JIjgGGVaK2u1VE3sBB6A/image?w=297&h=119&rev=10&ac=1 |
| שתיים על שתיים  https://lh6.googleusercontent.com/SmkgklhXqGo29JRPFt2aMtszVYfjeUoeVBcKJjcTFNM3boF8Csd2OLdOzopOv6x9PrDFAaPVoMb_sNPJYWiwZWf2cWW-AVcEDHQJmPEIhs-IgXE1k8u0lFAKnFngBemJ3g | שלוש על שתיים  https://lh4.googleusercontent.com/aDGIWVqRw11nSiz4JERqkf8XFi9IlEUTA7UrWDzIBoV89F4ZlftVd604X_n9ZHn2I8IOqr43iZFEzoyYRInEE74sD0IhCERdblS8_IlSLHqBa9rwUgYvsnbghGihcnR5zQ |

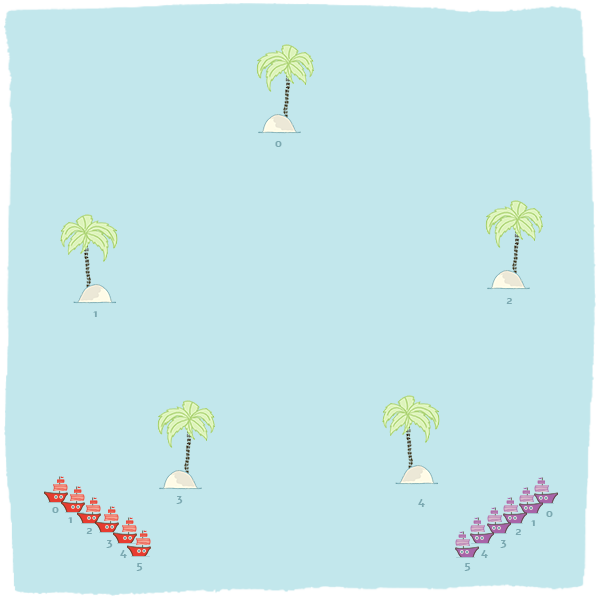
על מנגנון ההתקפה יפורט בהמשך.

כמו כן, כאשר שתי ספינות מקבוצות שונות מנסות לעלות על אותה משבצת הן מתנגשות ושוקעות (כמו במציאות). אבל על מנת למנוע מספינות של אותה קבוצה להתנגש ישנו מנגנון הגנה אוטומטי שמונע זאת (כלומר זה נעשה בשבילכם וקורה לבד אז אל תדאגו).

**תחילת המשחק:**

* כל האיים נטרלים
* כל ספינות הפיראטים מופיעות על המפה בנקודות ההתחלה שלהן

**מפת המשחק:**

****

* לכל אי ולכל ספינה מכל קבוצה יש מזהה קבוע (הנקרא id [מסוג int]), שהוא מספר שלא משתנה לכל אורך המשחק
* מיקומים על המפה נעשים באמצעות התייחסות לשורה ולעמודה. השורה הכי עליונה היא השורה ה-0 והעמודה הכי שמאלית היא העמודה ה-0. כלומר, הקוארדינטה (0, 0) היא בפינה השמאלית העליונה.

איך כותבים קוד?

ניווט:



[מדריך לכתיבת Bot ב-Python](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.gyp7lbsyo3jw)

[הסימולטור](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.6yh9h049108j)

[API למימוש Bot](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.j3vw3l519uci)

[API בסיסי](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.12itkdd0ze3o)

[API - איך זזים?](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.vrtisoiofkt0)

[ביצוע פעולות](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.bb2mz2hbiw7k)

[חוקי תקיפה מפורטים](https://docs.google.com/document/d/1sugWxVw5cUSOK3T1gUjRrrqLX3Jg_iq64e2ARAl0ung/preview#heading=h.msktdaeyplj)

מדריך לכתיבת Bot ב-Python

כאן נתאר את הצעדים הראשונים בכתיבת Bot.  בהמשך המסמך יוסבר איך אתם מריצים אותו.

כדי לייצר את ה-Bot הראשון שלכם עקבו אחרי הצעדים הבאים:

1. ראשית הורידו למחשב את הstarterkit מעמוד הבית של הקבוצה שלכם באתר האינטרנט של המשחק
2. פתחו את קובץ הzip למחשב שלכם
3. צרו קובץ פייתון חדש בשם MyBot.py (או שם לבחירתכם) בתיקייה bots שנמצאת בstarterkit
4. כתבו ושמרו בקובץ זה את התוכן הבא:

def do\_turn(game):

   pass

כעת ה-Bot שלכם לא עושה כלום מכיוון שהמימוש של do\_turn הוא ריק. הקוד שתכתבו בפונקציה זו תיקרא בכל תור במשחק.

שימו לב ש- do\_turn מקבלת פרמטר game, זהו האובייקט שמכיל את קריאות ה-API, עליהן נפרט בהמשך.

כעת נציג אסטרטגיה בסיסית ע"י האלגוריתם הבא:

* *אם כל האיים בשליטתי*
  + *אל תעשה כלום*
* *אחרת*
  + *סמן את האי הראשון מתוך האיים שלא שלי ב-island*
  + *עבור כל ספינת פיראטים ששייכת לקבוצה שלי:*
    - *שלח את הספינה לכבוש את island*

כעת נדגים איך ניתן לממש את האלגוריתם

לשם כך, נרצה לבדוק אם יש אי שלא בשליטתינו. הפונקציה הבאה מחזירה לנו רשימה עם כל האיים שלא בשליטתי:

game.not\_my\_islands()

כדי לבדוק מהו מספר האיים שלא בשליטתינו, נוכל לבדוק את אורך הרשימה:

len(game.not\_my\_islands())

ועל מנת לסיים את הפונקציה do\_turn, נשתמש בפקודה הבאה:

return

נרצה לסיים את התור שלנו אם אורך הרשימה הוא 0 (אין איים באינם תחת שליטתי). כלומר:

if len(game.not\_my\_islands()) == 0:

   return

עד כה פונקצית הdo\_turn שלנו נראית כך (ודאו כמובן שההזחה מתאימה):

|  |
| --- |
| def do\_turn(game):     if len(game.not\_my\_islands()) == 0:         return |

בשלב הבא, נרצה למצוא את האי הראשון מתוך האיים שלא בשליטתנו ולשמור אותו במשתנה בשם island. לכן נשתמש באותה הפונקציה כאשר אנחנו לוקחים את האי הראשון ברשימה:

island = game.not\_my\_islands()[0]

על מנת שנוכל לדעת איזה מספר אי בחרנו נרצה להדפיס למסך הודעה בכל תור שמדפיסה מספר זה. הפונקציה שמדפיסה למסך נקראת:

game.debug();

והיא מקבלת מחרוזת להדפיס. לכן במקרה שלנו נרשום:

game.debug("going to island " + str(island.id))

כאשר island.id נותן לנו את מספר האי והפונקציה ()str  הופכת אותו למחרוזת. כך כשנריץ את הקוד שלנו, נוכל לראות בכל תור איזה אי בחרנו.

כעת נרצה לשלוח את כל הפיראטים שלנו לאי island. כדי לקבל רשימה של כל הפיראטים שלנו שכרגע על המפה:

game.my\_pirates()

נשמור את הרשימה למשתנה בשם pirates:

pirates = game.my\_pirates()

עד כה הקוד שלכם אמור להיראות כך:

|  |
| --- |
| def do\_turn(game):     if len(game.not\_my\_islands()) == 0:         return     island = game.not\_my\_islands()[0]     game.debug("going to island " + str(island.id))     pirates = game.my\_pirates() |

נרצה להורות לכל ספינה בתוך pirates (היכונו ללולאת for) לנוע לכיוון האי שבחרנו בisland.

על מנת למצוא את כיוון התנועה אל island נשתמש ב:

game.get\_directions(pirate, island)

שמחזירה לנו **רשימה** של כיוונים שאם pirate תלך באחד מהם, היא תוכל להגיע ל-island בדרך הקצרה ביותר.  
נבחר את הכיוון הראשון ברשימה ונכניס אותו למשתנה בשם direction כלומר:

direction = game.get\_directions(pirate, island)[0]

ואז כדי לבצע את התזוזה נשתמש בפונקציה בשם set\_sail (=להפליג):

game.set\_sail(pirate, direction)

סה"כ, כדי לשלוח את כל הפיראטים בpirates אל האי island:

pirates = game.my\_pirates()

for pirate in pirates:

   direction = game.get\_directions(pirate, island)[0]

   game.set\_sail(pirate, direction)

סה"כ do\_turn שלכם תיראה עכשיו כך:

|  |
| --- |
| def do\_turn(game):     if len(game.not\_my\_islands()) == 0:         return     island = game.not\_my\_islands()[0]     game.debug("going to island " + str(island.id))     pirates = game.my\_pirates()     for pirate in pirates:         direction = game.get\_directions(pirate, island)[0]         game.set\_sail(pirate, direction) |

זהו, כתבתם את ה-Bot הראשון שלכם! מוכנים לראות אותו בפעולה?

הסימולטור

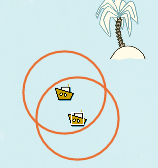
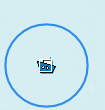
1. היכנסו לתיקייה starter\_kit והריצו (לחצו פעמיים) קובץ בשם open\_cmd. זה יפתח לכם חלון של command line בתיקייה הנוכחית.

[**שימו לב** שהstarter\_kit יושב במקום שאין בנתיב עברית. מקום טוב לדוגמא הינו \:C או \:D]

1. הריצו בה את הפוקדה הבא: run.bat bots\MyBot.py bots\demoBot1.pyc (אם בחרתם שם קובץ שונה אז שימו אותו במקום MyBot.py)
2. לאחר כמה שניות במהלכם תראו שהמשחק מתקדם ב-command line, ייפתח לכם tab בדפדפן ובו הסימולטור של המשחק אשר מציג את המשחק בצורה גרפית. כאן תוכלו לצפות בקוד שתכנתתם משחק.
3. בסימולטור האישי שנפתח, השחקן הראשון שהקלדתם יהיה תמיד בצד שמאל של המפה והשני יהיה תמיד בצד ימין של המפה. כמו כן, הצבעים והלוגו לא יהיו כמו שבחרתם באתר. את הלוגו והצבעים שלכם נראה רק בהמשך.

על המסך יש כפתורים שאיתם תוכלו לנווט את הצפייה במשחק:  


ועוד שני כפתורים שימושיים הם:  


כדי לראות את הרדיוס של ספינה (בתחום זה היא יכולה להטביע ספינות אחרות), ואת כל הרדיוסים שבהם היא נמצאת, עמדו עם העכבר על הספינה:  


API למימוש Bot

להלן הפונקציות שנותנות לכם לקבל מידע בזמן המשחק מתוך הקוד שלכם.

שימו לב כי כל הערכים החוזרים מהפונקציות הינם by-value ולא by-reference. כלומר, הנתונים עדכניים רק לתור בהם קראתם לפונקציה. רק ה- id (מזהה) של הספינות והאיים אינו משתנה במהלך המשחק.

ראשית נתאר את הטיפוסים ספינת-פיראט ואי:

אובייקט מסוג אי מכיל את המידע הבא:

|  |  |
| --- | --- |
|  | Class Island: |
| המספר הסידורי של האי | id               (int) |
| קורדינאטות (עמודה,שורה) של האי | location         (tuple) |
| השחקן אליו שייך האי, הערך יכול להיות , game.NEUTRAL, game.ME או game.ENEMY | owner            (int) |
| מי מנסה לכבוש עכשיו את האי. הערך יכול להיות None (אף אחד), game.ME או game.ENEMY | team\_capturing   (int) |
| מספר התורות ברצף שהאי כבר נתקף | capture\_duration (int) |

אובייקט מסוג ספינה מכיל את המידע הבא:

|  |  |
| --- | --- |
|  | Class Pirate: |
| המספר הסידורי של הספינה | id              (int) |
| קורדינאטות (עמודה,שורה) של המיקום הנוכחי של הספינה | location        (tuple) |
| השחקן אליו שייך הספינה, הערך יכול להיות game.ME או game.ENEMY | owner           (int) |
| האם הספינה במשחק עכשיו (אחרת היא שקעה ותחזור למשחק בעוד מספר תורות) | is\_lost         (boolean) |
| המיקום ההתחלתי של הספינה | initial\_loc     (tuple) |
| אם הספינה שקעה, כמה תורות עד שהיא חוזרת למשחק. אחרת None. | turns\_to\_revive (int) |

**API בסיסי**

פונקציות לאחזור איים (כל הפונקציות המתוארות בטבלה מחזירות רשימה של אובייקטים מסוג island למעט הפונקציה האחרונה שמחזירה אי בודד):

|  |  |
| --- | --- |
| שם הפונקציה | תיאור |
| game.islands() | מחזירה את כל האיים במשחק |
| game.my\_islands() | מחזירה את כל האיים שנכבשו ע"י השחקן |
| game.not\_my\_islands() | מחזירה את כל האיים שאינם כבושים ע"י השחקן |
| game.enemy\_islands() | מחזירה את כל האיים שנכבשו ע"י האויב |
| game.neutral\_islands() | מחזירה את כל האיים שלא נכבשו על ידי אף קבוצה |
| game.get\_island(id) | מקבלת מזהה יחודי (מספר מ0 עד 4) ומחזירה את האי המתאים במשחק. |

פונקציות לאחזור פיראטים (כל הפונקציות המתוארות בטבלה מחזירות רשימה של אובייקטים מסוג pirate למעט הפונקציות האחרונות שמחזירות פיראט בודד):

|  |  |
| --- | --- |
| שם הפונקציה | תיאור |
| game.my\_pirates() | מחזירה רשימה של כל הפיראטים שלי שמופיעות עכשיו על המפה |
| game.my\_lost\_pirates() | מחזירה רשימה של כל הפיראטים האבודים של השחקן |
| game.enemy\_pirates() | מחזירה רשימה של כל הפיראטים של האויב שעל המסך |
| game.get\_pirate\_on(island) | מקבל אי או מיקום ומחזירה את הפיראט שעל האי או במיקום זה. |
| game.get\_my\_pirate(id) | מקבלת מזהה יחודי (מספר מ0 עד 5) ומחזירה את הפיראט המתאים של השחקן |

**API - איך זזים?**

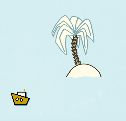
על מנת לנוע במשחק ישנן שתי פונקציות מפתח והם get\_directions ו - set\_sail.

כאן הם יתוארו בקצת יותר פירוט ועם דוגמאות על מנת שתלמדו לשלוט בתנועה שלכם בצורה טובה.

game.get\_directions(loc1,loc2)

הפונקציה מקבלת שני פרמטרים. הם יכולים להיות מסוג pirate/island או tuple של (עמודה,שורה).

הפונקציה מחזירה **רשימה** של כיוונים (גם אם יש רק כיוון אחד).  
כיוון הוא אחד מהמחרוזות : "n" (צפון), "s" (דרום), "e" (מזרח), "w" (מערב) או "-" (להשאר במקום))  שבהם יש ללכת כדי להגיע מ-loc1 ל-loc2 במסלול המהיר ביותר.

בדוגמא הבאה, הספינה הכתומה מנסה להגיע לאי הקרוב. הנה מצב בו הפונקציה get\_directions תחזיר רשימה שבה כיוון אחד בלבד  והוא 'e':  
  
ולהלן דוגמא שבה הפונקציה תחזיר רשימה של שני כיוונים 'n','e':  


הפונקציה שפוקדת על ספינה לנוע הינה:

game.set\_sail(pirate,direction)

מקבלת אובייקט מסוג pirate ואת כיוון התזוזה. כיוון התזוזה יכול להיות n,s,e או w (או '-'. ראה get\_directions לעיל).

הפונקציה תבצע תזוזה של הספינה בכיוון שהתבקש.

עוד פונקציה שימושית שתוכלו להיעזר בה היא:

game.destination(obj,direction)

הפונקציה מקבלת אובייקט מסוג pirate  או tuple של (עמודה,שורה) כפרמטר ראשון וכיוון   
(אחד מתוך 'n','w','s','e') כפרמטר שני,  ומחזירה את המיקום החדש של הספינה, אם תלך בכיוון שניתן.

**ביצוע פעולות**

|  |  |
| --- | --- |
| שם הפונקציה | תיאור |
| game.get\_directions(pirate, island) | מקבלת אובייקט פיראט ואובייקט אי ומחזירה רשימה של כיוונים בהם הפיראט צריך ללכת על מנת להגיע לאי המצוין |
| game.set\_sail(pirate, direction) | מקבלת אובייקט פיראט ומיקום (מחרוזת של 'n','s','w' או 'e') ושולחת פקודה להזיז את הפיראט בכיוון המצוין |
| game.distance(pirate, island) | מקבלת שני פרמטרים שיכולים להיות איים או פיראטים (או מיקום) ומחזירה את המרחק (בתורות) ביניהם. |
| **game.debug(string)** | מחליף את print. מדפיס עבורכם הודעות מהbot שלכם (כלי עזר לפיתוח) |
| game.destination(pirate, direction) | מקבלת אובייקט פיראט וכיוון ומחשבת את המיקום החדש של הפיראט לו ילך בכיוון זה |
| game.is\_capturing(pirate) | מקבלת אובייקט פיראט ומחזירה True אם הפיראט עכשיו כובש אי אחרת False. |
| game.get\_turn() | מחזירה את מספר התור הנוכחי (מספר בין 1 ל1000) |

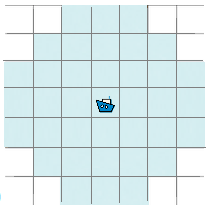
לטבלה מסודרת של כל הAPI לחצו כאן: [Python API](https://docs.google.com/document/d/1gWUXp3gpeeEUW1LPj4dZyWoqWnkbHDt0bnaFcWzAML0/edit?usp=sharing)

(אם הלינק לא עובד תעתיקו את :https://docs.google.com/document/d/1gWUXp3gpeeEUW1LPj4dZyWoqWnkbHDt0bnaFcWzAML0/edit?usp=sharing)

**חוקי תקיפה מפורטים**

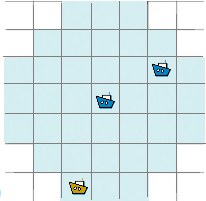
סעיף זה מסביר בפירוט את חוקי התקיפה והינו סעיף מתקדם. מומלץ לקרוא סעיף זה רק בשלב מתקדם יותר, לאחר שכבר הצלחתם לכתוב בוט משלכם.

כאשר ספינות מקבוצות שונות נמצאות באותו איזור הן יכולות לגרום לספינות יריבות לשקוע או להגן אחת על השניה בהתאם לחוקיות הבאה. לכל ספינה מוגדרת הרדיוס הבא:

(הספינה הכחולה במרכז והמשבצות הצבועות מוגדרות כתחום הרדיוס)

**כאשר ברדיוס של ספינה יש *יותר* ספינות של האויב מאשר ספינות חברות** (לא סופרים את הספינה עצמה) אז היא תישקע ותחזור רק לאחר 20 תורות למיקומה ההתחלתי.

להלן דוגמא:



במקרה זה הכחולה במרכז לא תישקע כיוון שיש לה כתומה (אויב) אחת ברדיוס אך גם כחולה (חברה) אחת ברדיוס.

במקרה הבא הספינה הכחולה תישקע כיוון שיש לה אויב אחד ברדיוס ואין לה חברות (אם לספינה הצהובה גם כן אין חברות ברדיוס שלה היא תישקע גם כן)

