**אלגוריתם למשחק XO מול המחשב (רמה בסיסית)**

**כללי משחק:**

שלב א – נקלוט מהלקוח מה התו שהוא רוצה

* אם בחר X המחשב יהיה O
* אם בחר O המחשב יהיה X

שלב ב – נקלוט מהלקוח האם הוא רוצה להתחיל במשחק או לא

* אם כן – השחקן יהיה ראשון והמחשב יהיה שני
* אם לא – המחשב יהיה ראשון והשחקן יהיה שני

**מספר האופציות של ניצחון:**

תהיה המטריצה ממימד n,

מספר האפשרויות לנצחון יהיה: n + n + 2

(מספר שורות + מספר עמודות + 2 אלכסונים).

**ניצור שני מערכים עבור השחקן ועבור המחשב – כל מערך יהיה בגודל n + n + 2**

* n תאים ראשונים יהיו עבור השורות במטריצה
* n תאים הבאים יהיו עבור העמודות במטריצה
* תא הבא יהיה עבור האלכסון הראשי
* תא הבא יהיה עבור האלכסון המשני

**עבור כל בחירת תא במטריצה (של שחקן**):

* נפנה למערך החד ממדי של השחקן
* נבדוק מה התא שהשחקן בחר במטריצה – ואז בהתאם נקדם באחד את תוכן התאים המתאימים לבחירה, לפי האלגוריתם הבא:

התא בהכרח גם בשורה מסוימת, וגם בעמודה מסוימת:

* + כדי לסמן במערך החד ממדי את השורה שנבחרה – נפנה לתא לפי אינדקס השורה במטריצה
  + כדי לסמן במערך החד ממדי את העמודה שנבחרה – נפנה לתא לפי אינדקס העמודה במטריצה +n

נבדוק האם הוא גם באלכסון הראשי:

(אופן הבדיקה יהיה האם אינדקס שורה שווה לאינדקס עמודה)

* + אם התא נמצא באלכסון הראשי כדי לסמנו במערך החד ממדי– נפנה לתא n+n

נבדוק האם הוא גם באלכסון המשני:

(אופן הבדיקה יהיה האם אינדקס שורה + אינדקס עמודה = ממד המטריצה - 1)

* + אם התא נמצא באלכסון המשני כדי לסמנו במערך החד ממדי– נפנה לתא 1+n+n
* בכל פעם שנקדם את תוכן התא במערך החד ממדי – נבדוק האם הוא שווה לn – אם כן סימן שהשחקן ניצח
* אם השחקן לא ניצח – נבדוק האם הגענו למצב של תיקו על ידי :

סכימת תוכן n התאים הראשונים של המערך החד ממדי הראשון + סכימת תוכן n התאים הראשונים של המערך החד ממדי השני

אם הסכום יהיה שווה לn\*n סימן שיש תיקו

**עבור כל בחירת תא במטריצה (של המחשב**):

* נבחר באיזה תא להציב, על פי האלגוריתם הבא:
  + נבדוק האם המחשב יכול לנצח בתור הנוכחי – כלומר נעבור על כל המערך החד ממדי של המחשב, ונבדוק האם יש תא המכיל n-1

אם כן – נבחר את התא הריק על מנת לנצח בתור הנוכחי

* + נבדוק האם השחקן יכול לנצח בתור הבא – כלומר נעבור על כל המערך החד ממדי של השחקן, ונבדוק האם יש תא המכיל n-1

אם כן – נבחר את התא הריק על מנת לחסום את האופציה של השחקן, ולא לתת לו לנצח

* + נבדוק האם יש אופציה שניתנת לניצחון עתידי –
    - כלומר נעבור על כל המערך החד ממדי של השחקן, ונבדוק האם יש תא המכיל 0
    - אם יש כמה תאים במערך השחקן המכילים אפס – נבדוק באיזה תא מקביל של המחשב אותו התא יהיה גדול מאפס

אם קיים כזה תא – נבחר אלמנט במטריצה המוכל באותו תא במערך החד ממדי

* + - אם כל התאים שהם אפס אצל השחקן – הינם אפס גם אצל המחשב, נשתמש באלגוריתם הבא:

**בממד שהוא זוגי**

עדיפות ראשונה:

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

יכול להיות שורה / עמודה / אלכסון

**בממד שהוא אי זוגי**

**עדיפות ראשונה:**

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

יכול להיות שורה / עמודה / אלכסון ראשי / אלכסון משני

**עדיפות שניה:**

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

יכול להיות שורה / עמודה / אלכסון

* נבדוק מה התא שהמחשב בחר במטריצה – ואז בהתאם נקדם באחד את תוכן התאים המתאימים לבחירה, לפי האלגוריתם הבא:

התא בהכרח גם בשורה מסוימת, וגם בעמודה מסוימת:

* + כדי לסמן במערך החד ממדי את השורה שנבחרה – נפנה לתא לפי אינדקס השורה במטריצה
  + כדי לסמן במערך החד ממדי את העמודה שנבחרה – נפנה לתא לפי אינדקס העמודה במטריצה +n

נבדוק האם הוא גם באלכסון הראשי:

(אופן הבדיקה יהיה האם אינדקס שורה שווה לאינדקס עמודה)

* + אם התא נמצא באלכסון הראשי כדי לסמנו במערך החד ממדי– נפנה לתא n+n

נבדוק האם הוא גם באלכסון המשני:

(אופן הבדיקה יהיה האם אינדקס שורה + אינדקס עמודה = ממד המטריצה - 1)

* + אם התא נמצא באלכסון המשני כדי לסמנו במערך החד ממדי– נפנה לתא 1+n+n
* בכל פעם שנקדם את תוכן התא במערך החד ממדי – נבדוק האם הוא שווה לn – אם כן סימן שהמחשב ניצח
* אם המחשב לא ניצח – נבדוק האם הגענו למצב של תיקו על ידי :

סכימת תוכן n התאים הראשונים של המערך החד ממדי הראשון + סכימת תוכן n התאים הראשונים של המערך החד ממדי השני

אם הסכום יהיה שווה לn\*n סימן שיש תיקו

**לינק להעמקה:**

<https://livebook.manning.com/#!/book/deep-learning-and-the-game-of-go/chapter-4/v-7/49>