**Intro to AI HW2 – Dry:**

**סטודנטים:**

עמית ענבר - 315836569

לינוי גנטי- 208536284

1. האסטרטגיה של שחקן זה היא להימנע מבורות בעומק 2 (2 מהלכים מראש)

ולהיצמד לקירות (-1 במפה) .

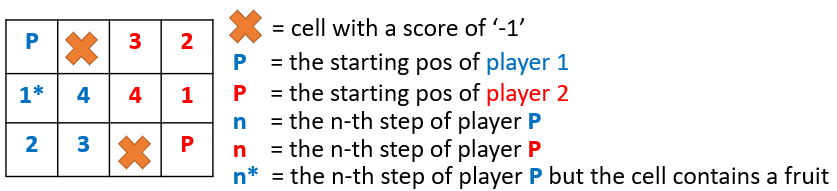
יתרונות:

1. ממלא את המפה בצורה מיטבית אם היה משחק לבד (הכי הרבה תאים שניתן להגיע אליהם ולהמשיך מהם היו נצבעים).
2. נמנע מלעבור למשבצת אם אין ממנה מהלכים אפשריים (נמנע מתאים **שמסווג** כבורות) כל עוד יש לשחקן אפשרות נוספת (ייתכן והמהלך היחיד שנותר הוא להיכנס לבור)

חסרונות:

1. "ראייה" של 2 מהלכים קדימה סה"כ והיא אינה מתחשבת במהלכים אפשריים של

ה-"אויב".

1. גם עם "ראייה" של 2 מהלכים קדימה ניתן להיכנס לבורות בעומק של יותר ממש מ-1:
2. דוגמא:

הסבר:

1. יתרונות:

* דירוג גבוה יותר של מסלולים שעוברים דרך פירות ובכך משפר את הסיכוי לקבל ניקוד גבוה.
* שימוש במידע נוסף מלבד חיפוש עיוור (היוריסטיקה מיודעת).

חסרונות:

* ייתכן ולא קיים מסלול בין המיקום הנוכחי לפרי (יש חסימה בכל מסלול אפשרי לפרי).
* יכולים להיות קירות (-1) החוסמים את כל המסלולים מהמיקום הנוכחי לפרי שאורכם הוא מרחק מנהטן מינימלי מהמיקום הנוכחי לפרי כך שיכול להיווצר מצב בו אנחנו רק "מתרחקים" במסלול הקיים מהמיקום הנוכחי לפרי:



* ההיוריסטיקה יכולה "להתבדות" אם קיים מסלול עם מרחק מנהטן אך הפרי נעלם לפני.
* יכול להיווצר מצב בו "נחתוך" את השטח שהשחקן שלנו יכול להגיע אליו ובכך "להיחסם" מוקדם יותר.
* אם הפרי נמצא בתוך בור אולי לא משתלם לקחת אותו.

1. ג
2. כן – בכל שלב בעץ המינימקס בו בוררים צעד של אחד האויבים נלקח המינימום מהבנים שלו (רקורסיבית עד שמגיעים לשלב בעץ בו אנחנו בוררים את התור שלנו – שם נקח מקסימום)
3. חסרונות:
   1. מקדם הסיעוף גדול בצורה משמעותית בשיטה זו ולכן גם זמן החישוב לכל צעד בודד. – לא פרקטי.
   2. מתייחסים לכל האויבים באופן שווה ו-"מפחדים" מכולם – בעצם אסטרטגיה פחדנית ואף פרנואידית(?) שיכולה לגרום לתוצאה שרחוקה מלהיות אופטימלית.
4. טרמינולוגיה: עבור  שחקנים: , .

תהיי  פונקציית דירוג (היוריסטיקה) לשחקן עבור כל אחד מצעדיו האפשריים.

לכל שחקן  וצעד אפשרי  נחשב את הדירוג: .

ונקח את התוחלת של של כל הצעדים האפשריים ונמקם אותם בטבלת דירוג הממוינת

לפי  אותה (את הטבלה) נעדכן בכל צעד שנבצע.

אקדיש לכל האויבים שלי שכבה אחת בעץ המינימקס בכל איטרציה באופן הבא:

אמשקל את כל צמתי הבנים (מתוכם אבחר את המינימלי) כך:



(אדגים עם טבלת דירוגים ספציפית לדוגמא)

טבלה:

|  |  |  |
| --- | --- | --- |
| average score | player | rank |
| 789 | 2 | 1 |
| 456 | 4 | 2 |
| 123 | 3 | 3 |

כעת, לכל צעד אפשרי של האויב אסכום את הניקוד המשוקלל של כל האויבים:



בדוגמא שלנו, בתור הבא של (מטעמי נוחות ומינימליות אניח כי יש לכל היותר 2 פעולות אפשריות):

בהינתן:



מתקיים:



ואמקם בעץ כבן של  עבור האופרטור  ומתוך הבנים האלו אבחר את המינימלי.

וכך גם אעשה עבור שאר השחקנים רק שעבורם, עבור שחקן :



בעצם, הרעיון באלגוריתם שלנו הוא לדרג את האויבים שלנו ולמשקל אותם לפי טיב היסטואציה שלהם ובכך לגרום לסוכן שלנו "לפחד" מההתקדמות של הסוכנים "החזקים" יותר מאשר מההתקדמות של הסוכנים "החלשים" תוך מיקסום (מקסימום) הניקוד העצמי.

1. תשובה:
2. **כן**, גיזום הענפים ב-alpha-beta יחסוך ביקורים במצבים שפסלנו ולכן יהיה מהיר יותר מ-minimax.
3. **לא**, alpha-beta פוסל רק צעדים שיחמירו את מצב הסוכן שלנו ולכן יתקדם בדיוק כמו minimax רק מהר יותר (?) – אולי התשובה היא כן אם עומק החיפוש יכול לגרום לנו לפסול "דרך" שלא הייתה נפסלת אם היה מדובר ב-alpha-beta בלי עומק סופי.התשובה היא לא כי מדברים על עומק חיפוש זהה בשני האלגוריתמים ולכן תוצאת הגיזום זהה לתוצאת המינימקס.
4. תשובה:
5. **לא** , בגיזום הענפים ב-alpha-beta עם סידור ילדים אנו מסתמכים על היורסטיקה כדי לסדר את הילדים ולכן נחלק למקרים:

במקרה בו ריצת היורסיטקה מהירה אנו משארים כי גיזום alpha-beta עם סידור ילדים ירוץ יותר מהר מאשר בגיזום -alpha-beta .אך במקרה בו ריצת היוריסטיקה מאוד ארוכה ריצת בשיטת גיזום -alpha-beta תהיה מהירה יותר מאשר בשיטת גיזום alpha-beta עם סידור ילדים.

1. **כן**, שניהם יתקדמו בדיוק כמו minimax רק בזמני ריצה שונים אחד מהשני.
2. תשובה:

Anytime contact הינה שיטה בה בעצם משנים את הגדרת הבעיה. במקום להחזיר את הצעד הטוב ביותר אנו רוצים להחזיר את הצעד הטוב ביותר תוך K שניות.

העמקה הדרגתית זו שיטה המקובלת להתמודדות עם הזמן לכן נגביל את העומק אליו האלגוריתם יכול להגיע ונגדיל את העומק לאט לאט עד שיגמר הזמן.

Anytime contact של האלגורתימם Minimax בעצם משתמש בפונקציית הערכה היורסטיקה ומפתחת צמתים באופן הדרגתים לפי שיטת העמקה ההדרגתית ומחזיר את הערך מקסימלי עד כה שהצליח להגיע אליו.

1. תשובה: הבעיה בהעמקה הדרגתית המוצגת בהרצאה הינה בעיית האיטרציה האחרונה.בממוצע האלגוריתם יופסק באמצע האיטרציה האחרונה והצעד שיוחזר הוא התוצאה של האיטרציה הקודמת. האיטרציה האחרונה צורכת הרבה משאבים וחבל לא להשתמש במידע שהגילינו שם לכן הפתרון שהוצא בהרצאה היה שבכל איטרציה נשמור את הערך המינימקס של כל אחד מהבנים ברמה העליונה ובכך ננצל את המידע שהשגנו.