אוניברסיטת בן גוריון

הפקולטה למדעי המחשב

מיני פרוייקט - נושאים במשחקי חשיבה – חידאתו

*מטרת ונושא הפרוייקט: להבין כיצד ניתן להשתמש בכלים מקורסים קודמים לפתרון המשחק חידאתו. במידת האפשר נתבונן גם במופעים אקראיים של הבעיה וננסה להבין את התנהגותם הטיפוסית.*

מרצה הקורס: פרופ' דניאל ברנד

מגישים: טל יצחק (204260533)

חיים סבן (308000371)

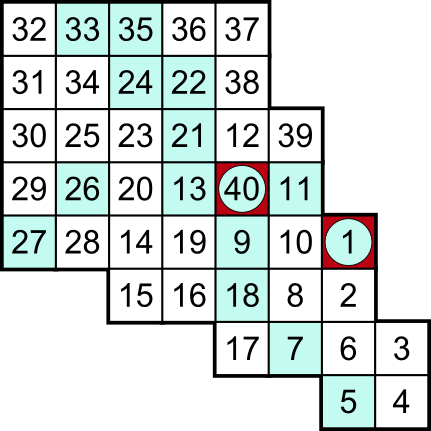
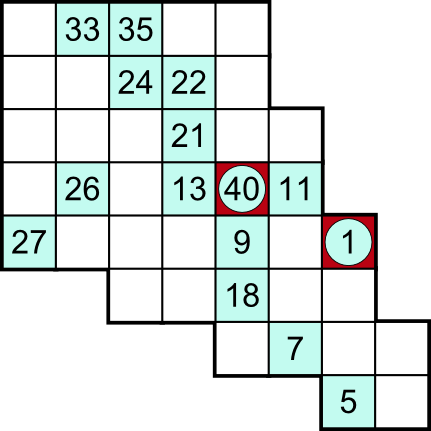
כלל הקוד בפרויקט נכתב בJava, וזמין ב-GitHub:

[github.com/talitz/Topics-in-Logic-Puzzle-Mini-Project-On-Hidato](https://github.com/talitz/Combinatorial-Optimization-MaxSAT)

סמסטר ב', תשע"ז, 2017

חידאתו – מבוא וכללי המשחק

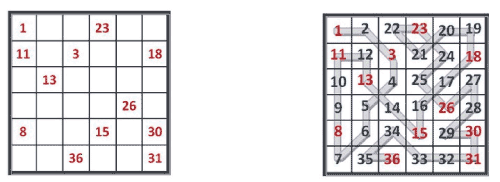
חידאתו היא חידה בה יש למלא על לוח משובץ את המספרים החסרים. על לוח החידה מסומנים בעיגול שני [מספרים](https://he.wikipedia.org/wiki/%D7%9E%D7%A1%D7%A4%D7%A8_%D7%98%D7%91%D7%A2%D7%99), הקטן ביותר והגדול ביותר שיופיעו עליו. כמו כן מופיעים על הלוח מספרים נוספים, שהם תחילתו של הפתרון, ומבטיחים שלחידה פתרון יחיד. על פותר החידה לשבץ מספרים נוספים על הלוח, כך שתיווצר שרשרת של מספרים עוקבים. שני מספרים הם מספרים סמוכים בשרשרת כאשר הם נמצאים על הלוח זה לצד זה, אנכית, אופקית או באלכסון.



בהינתן מופע כלשהו של משחק החידאתו, ננסה לפתור את המשחק בעזרת רדוקציה לבעיות מוכרות במדעי המחשב.

רדוקציה למסלול המילטון עם אילוצים על הקודקודים

נשים לב שפיתרון של בעיית החידאתו מגדיר מסלול, שמתחיל בקודקוד עם הערך ההתחלתי בחידה (בדוגמא: 1), עובר דרך הריבועים עם הערכים 2,3,4, וכן הלאה עד הערך הגדול ביותר 36.



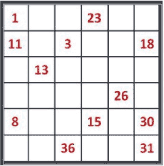
אם כן, בהגדרת הרדוקציה נרצה להשתמש בקופסה השחורה כאלגוריתם שבהינתן גרף, מחזיר את מסלול המילטון בגרף **עם אילוצים על הקודקודים** אם קיים, ואם לא מחזיר "לא קיים".

נגדיר מספר מושגים שיעזרו לנו בהצגת האלגוריתם:

קלט לבעיה: מערך דו מימדי , כאשר כל תא במערך מייצג מספר נתון של בעיית החידאתו, או 1- כמקום ריק שעל השחקן למלא.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| null | null | 23 | null | null | 1 |
| 18 | null | null | 3 | null | 11 |
| null | null | null | null | 13 | null |
| null | 26 | null | null | null | null |
| 30 | null | 15 | null | null | 8 |
| 31 | null | null | 36 | null | null |

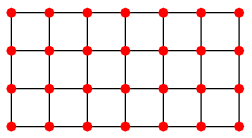
דוגמא:



רשימת "הערכים החסרים" :M לבעיה הנתונה היא כל הערכים אותם השחקן צריך להשלים בעצמו, למשל בדוגמא הנתונה:

גרף סריג (grid/lattice graph): גרף דו-מימדי , אשר ניתן להצגה במערכת צירים אוקלידית , ויוצר אריח (סריג).

דוגמא: סריג 4 על 7:



נגדיר מסלול המילטון **עם אילוצים על הקודקודים**:

בהינתן גרף , כאשר לכל קודקוד קיים ערך (מספר טבעי המייצג את ערכו של הקודקוד, או null אם אין לקודקוד ערך) נגדיר מסלול המילטון עם אילוצים באופן הבא:

1. מסלול המילטון.
2. מקיים כי עבור המסלול , אם הקודקוד ה- במסלול הוא בעל ערך , אזי .

דוגמא:

עם הערכים:

*נקבל כי:*

1. הוא מסלול המילטון, אבל לא מסלול המילטון עם אילוצים על הקודקודים, משום שהקודקוד הוא הקודקוד השני במסלול ובעל ערך השווה ל- .
2. הוא מסלול המילטון עם אילוצים, משום שמתקיים:
   1. *הקודקוד הוא הראשון במסלול ועם ערך השווה ל-.*
   2. *הקודקוד הוא השני במסלול וללא ערך (null), ולכן אין תנאי עבורו.*
   3. *הקודקוד הוא הקודקוד השלישי במסלול ועם ערך השווה ל-.*
   4. *הקודקוד הוא הקודקוד הרביעי במסלול ועם ערך השווה ל-.*

*לכן, מסלול זה הוא מסלול המילטון עם אילוצים.*

אלגוריתם לפיתרון בעיית החידאתו בעזרת רדוקציה למסלול המילטון עם אילוצים על הקודקודים:

בהינתן קלט לבעיה, מערך דו מימדי :

1. בנה את רשימת הערכים החסרים בחידאתו.
2. בנה מהבעיה גרף סריג מכוון .
3. הסר את כל הקשתות הנכנסות לקודקוד שערכו .
4. הסר\_צלעות\_לא\_רלוונטיות\_בין\_קודקודים\_עם\_ערכים\_עוקבים().
5. כל עוד לא ריקה:

4.1. בחר מינימלי.

4.2. מצא את הקודקוד כך ש- .

4.3. בחר j כך ש(, שבץ .

4.5. כל עוד לא קיים מסלול המילטון עם אילוצים בגרף :

4.5.1. אם סיימנו לעבור על כל השכנים של , החזר "אין פיתרון לחידאתו" וסיים.

4.5.2. בחר כך ש(, שבץ

.

4.6. סמן את הקודקוד כחלק ממסלול המילטון.

4.7. הסר\_צלעות\_לא\_רלוונטיות\_בין\_קודקודים\_עם\_ערכים\_עוקבים().

5. החזר את .

הסר\_צלעות\_לא\_רלוונטיות\_בין\_קודקודים\_עם\_ערכים\_עוקבים()

עבור על הגרף ולכל 2 קודקודים עם ערכים עוקבים (אנכית, אופקית או באלכסון):

4.1. הסר את כל הצלעות הנכנסות ל- שהם לא .

4.2. הסר את (,).

4.3. אם היא חלק ממסלול המילטון:

4.3.1. הסר את כל הצלעות היוצאות מ- שהם לא .