תיק פרויקט - עידן משיח

**הקדמה ותודות:**

תחילה הייני רוצה להודות למוריי אלה וגיא שעזרו לנו במהלך בניית הפרויקט. זהו היה פרויקט שנמשך כמעט שנה שלמה ונדרשנו להשקיע בו שעות רבות. אצלי בחירת המשחק לקחה זמן רב עד שמצאתי את המשחק אותו אני מגיש לבגרות ולולא עזרתם של גיא ואלה ספק אם הייתי מוצא אותו. בנוסף לעזרה על בחירת המשחק הם גיא ואלה עזרו לנו בכתיבת קודים נכונים ותיקון בעיות. אלה אף הביאה את בניה אשר למדו פרולוג על מנת לעזור לנו בפרויקט.

בנוסף לתודות למורים הנני רוצה להגיד תודה גם לחבריי לכיתה אשר יש להם חלק לא קטן בפרויקט שלי.

**תוכן עניינים:**

פרק ראשון עמודים : 2-3

פרק שני עמודים: 4-8

פרק שלישי עמודים: 9-10

פרק רביעי עמודים: 11-14

פרק חמישי עמודים: 15-20

פרק שישי עמודים: 21-24

פרק שביעי עמודים: 25-26

פרק שמיני עמודים: 27

פרק תשיעי עמודים: 28

קבצי המשחק: 29-114

**פרק 1 - הצעת עבודה**

**נושא העבודה:** משחק לוח בשילוב בינה מלאכותית

**שם מגיש העבודה:** עידן משיח

**שם המורה המנחה:** אלה לב

**שנת הגשת העבודה:** 2015, תשע"ה

**מומחה אנושי:** חקרתי לבדי את המשחק, חוקיו ואסטרטגיות אפשריות והפכתי למומחה האנושי.

**קהל היעד:** המערכת מיועדת לכל אדם המעוניין לשחק במשחק Snort אשר צורת הלוח בו היא שונה (משושה) ולכן גם כמה מהחוקים, אפשרי לשחק בכל גיל.

**מבוא:**

המשחק מתאים לשני שחקנים: השחקנים יכולים להיות אנושיים, אחד מול השני או משחק של שחקן אנושי מול המחשב ברמות קושי שונות.

אין תזוזה על הלוח אלא הצבה של הכלים.

**מטרות:** המערכת הממוחשבת תהיה בעלת בינה שתוכל לנצח מומחה אנושי. יתבצעו מהלכים אסטרטגים ומחושבים.המערכת המומחית תהיה חלק ממשחק שישמש הנאה למשתמשים בו.

**שאלות עיקריות עליהן המערכת עונה:** המערכת המומחית תאפשר לשחקן לשחק במשחק נגד שחקן אחר, נגד שחקן מחשב. תהיינה אפשרויות של יציאה , עצה.

**אילוצים ודרישות:** 🖫 המערכת הממוחשבת תציע למשתמש מגוון אפשרויות (כמו בתפריטים השונים) ולא תגיע להחלטות בעצמה, חוץ ממהלכי השחקן הממוחשב.

🖫 שפת הכתיבה: פרולוג, השתמשתי בתוכנת Prolog Amzi.

🖫 לוח זמנים :

בחירת המשחק עד למצב עבודה תקין של "אנושי מול אנושי" - ספטמבר 08' עד ינואר 09'

רמת בינה 0 גרפיקה בסיסית, כולל תפריטים - ינואר 09' עד פברואר 09'

המשך עבודה כולל שיפור גרפיקה ותכנות רמות בינה 1,2,3 - פברואר 09' עד אפריל 09'

הוספת רמת בינה 4 בשילוב MiniMax וגראפיקה סופית - אפריל 09' עד מאי 09' (הגשה סופית)

🖫 ידידותיות למשתמש: במערכת מגוון תפריטים נוחים וידידותיים בשפה פשוטה. במידת הצורך, מתקבלות הודעות שגיאה ומתבצעות בדיקות תקינות למשתמש האנושי. המשחק מעוצב בצורה גרפית יפה ונעימה לעין.

🖫 המערכת מסוגלת לתת עצה/המלצה : המהלך הכי טוב שניתן לבצע באותו מצב ספציפי.

**פרק 2 - תיאור תחום הידע**

**שיטות לשאיבת הידע**

לאחר שבחרתי את המשחק Snort, שיחקתי בו מספר רב של פעמים. המטרה הייתה להבין מספר דברים חשובים:

🖫 מטרת המשחק

🖫 מהם המהלכים האפשריים שניתן לבצע בכל תור

🖫 מהו ניצחון במשחק

🖫 מהו המהלך הכי טוב שניתן לעשות בכל שלב במשחק, ולמה

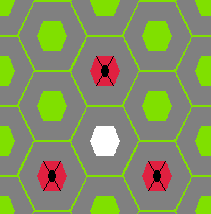
🖫 רעיונות לשדרוג המשחק הקיים - גרפיקה ושיפורים אפשריים נוספים, אך בלי לפגוע בחוקי המשחק, אופן ביצוע המהלכים וכו'.

**סיכום תחום הידע**

מהלך חוקי במשחק:

מהלך חוקי במשחק Snort הוא מהלך אפשרי שניתן לעשות בכל שלבי המשחק.

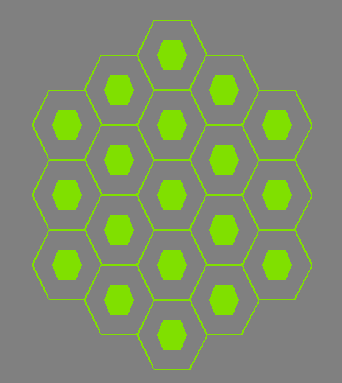
כל שחקן אשר מציבים בלוח יוצר משבצות אסורות אשר נמצאות בכיוונים 8 , 1 , 3 (אם 5 היא המשבצת בה הצבנו) אשר השחקן השני אינו יכול להציב בהם.



זוהי דוגמא בה ניתן לראות שחקן לבן ואת המשבצות שהוא משפיע עליהם וגורם להם להפוך לבלתי חוקיות עבור השחור.

**מצבים שונים של הלוח:**

**לוח התחלתי לדוגמה (משושה של שלוש):**

****

כל הלוחות הם בצורת משושה אשר אין בו כלים כלל.

ישנם 3 גדלים שונים ללוח המשחק:

לוח משושה בגודל 3

לוח משושה בגודל 4

לוח משושה בגודל 5

**ב. מטרת המשחק**

ניצחון במשחק מתקיים כאשר לא נותרים כלים אשר השחקן היריב יכול להציב על הלוח, כלומר המטרה היא להשתלט על כמה שיותר משבצות על הלוח.

**ג. מהלכים אפשריים**

המהלך האפשרי (ללא הכפתורים) היחיד המותר במשחק הינו הצבה של שחקן שלך על הלוח.

מהלכים אפשריים במשחק:

ניתן לבצע מהלכים לצורך הגעה לניצחון במשחק.

בנוסף, יש עוד שלל מהלכים אפשריים שניתן לבצע במהלך המשחק, כמו לחיצה על כפתורים שונים.

❖ שמירה –

save

ניתן ללחוץ על כפתור השמירה בכל שלב במהלך המשחק , אשר יגרום לשמירת המצב הנוכחי.

❖ עצה -

advice

כפתור העצה יאפשר למשתמש לקבל המלצה "חכמה" - המהלך הכי טוב שניתן לעשות על אותו הלוח כדי להגיע לניצחון.

בעת לחיצה על כפתור העצה, יוצג על הלוח המהלך המומלץ.

לאחר מכן יהיה ניתן לבחור האם לקבל את עצת המחשב או לבצע מהלך על פי בחירתו של השחקן.

❖ הוראות המשחק –

ora

ניתן ללחוץ על כפתור הוראות המשחק בתפריט הראשי.

לחיצה על כפתור זה יציג בפני המשתמש את הוראות המשחק. בסיום קריאת ההוראות יהיה ניתן ללחוץ על כפתור הenter וכך הוראות המשחק ייעלמו מהמסך.

❖ יציאה –

exit

בעזרת כפתור היציאה ניתן לצאת מהמשחק.

❖ חזור –

back

על לחיצת הכפתור חזור המצב יוחזר למצב הקודם שלו לפני ביצוע המהלך האחרון.

❖ משחק חדש –

NewGame

על לחיצת הכפתור משחק חדש המצב יוחזר למצב ההתחלתי שנקבע על ידי המשתמש.

❖ טען משחק –

reload

על לחיצת הכפתור טען משחק המצב שנשמר על ידי כפתור השמירה יטען למצב הנוכחי.

**פרק 3 - מבנה המערכת המומחית**

מערכת מומחית מורכבת מארבעה חלקים עיקריים :

1. בסיס הידע - KNOWLEDGE BASE

2. מערכת היסק - INFERENCE ENGINE

3. מערכת הסבר – EXPLANATORY INTERFACE

4. ממשק למשתמש – USER INTERFACE

**היררכית ההשתייכות של תוכניות מחשב מתחום הבינה המלאכותית:**

תכניות בינה מלאכותית

מערכות מבוססות ידע

מערכות מומחות

משלבות התנהגות אינטילגנטית בעזרת שיטות היוריסטיות

בסיס הידע ניפרד משאר המערכת

מוסיפות ידע מומחי כדי לפתור בעיות מן העולם האמיתי

### הסבר למושגים בסיסים במערכות מומחות :

**בסיס ידע –** בסיס הידע מתייחס לעולם התוכן המומחי של המערכת.

הוא מתאר את המרכיבים של תחום הידע, ואת הקשרים הקיימים בין המרכיבים השונים.   
בדרך כלל מתואר בסיס הידע באמצעות נתוני יסוד – עובדות, וכללים המתארים את הקשרים ביניהם ,קרי, חוקים. (במקרה של פרולוג)

העובדות מתארות את תחום הידע והחוקים את היחסים בין פרטים או מצבים בתחום הידע.

**מערכת היסק** – מערכת ההיסק מורכבת ממנגנון המסוגל להסיק מסקנות בהסתמך על בסיס הידע ומנגנון ההיסק קובע כיצד לנתח את הידע המתקבל.

המערכת מתייחסת אל בסיס הידע כאל אוסף של טענות הכוללות עובדות וחוקים בתחום הידע.  
המנגנון מקשר בין בסיס הידע כדי להגיע למסקנה.

תיאור סכמטי של המערכת המומחית וההסבר

בסיס הידע KNOWLEDGE BASE

משתמש - USER

ממשק למשתמש USER INTERFACE

מעטפת (SHELL)

מערכת הסבר INTERFACE EXPLANATORY

מערכת היסק INFERENCE ENGINE

**מדוע המערכת שלי הינה מערכת מומחית?**

המערכת המומחית שבניתי, מכילה את כל המאפיינים של המערכת המומחית הנזכרים לעיל, החוקים והעובדות אותם בניתי מהווים את בסיס הידע הדרוש להרצת הפרויקט לפי הגדרותיו, מסך המשחק, הכפתורים ומסך ההסבר משמשים כממשק למשתמש, מערכות ההיסק וההסבר אשר משולבים במהלכי המחשב ובהמלצת המהלך לשחקן מסבירה מדוע התוכנית פעלה בדרך שפעלה. מערכות אלו מהוות "מטא אינטרפטר" שהוא מרכיב מפתח במערכות מומחות.  
המערכת שבניתי מכילה את כל מרכיבי המערכת הדרושים והמצויינים לעיל, ולכן היא מערכת מומחית.

**פרק 4**

**4.1 – בחירת שיטה לייצוג מידע**

מצב(\_תור\_מי,\_לוח)

אצלי המצב מורכב משני פרמטרים , \_תור\_מי אשר מציין לי התור של מי עכשיו , כלומר מי מבצע עכשיו את המהלך , ו - \_לוח שזהו רשימת כלים שבה כל כלי נמצא על הלוח.

**4.2 – תרשים לולאת המשחק**

בחירת אפשרויות משחק

הדפסת לוח

האם קיימים מהלכים אפשריים

כן לא

השחקן שתורו בוחר היכן לשים כלי

המחשב בודק תקינות המקום המבוקש

תקין לא תקין

מתבצע המהלך – הכלי מוצב במקומו הודעה מתאימה

התור מוחלף

סוף משחק?

כן לא

מי ניצח – הודעה מתאימה

סגור חלון

הלולאה בפרולוג עובדת על ידי הכשל , השפה מבצעת את כל השלבים על ובסוף כאשר היא בודקת אם סוף המשחק (בתנאי שאין סוף משחק ) היא נכשלת וחוזרת חזרה למעלה ל - 'חזור' וככה היא ממשיכה עד שהיא מצליחה ואז היא סוגרת את המשחק.

**4.3**

משחק:-

%% 1. האתחל משחק בעצם הוא זה שמנקה את כל מה שהיה בזיכרון לפני זה ויוצר מצב %% חדש , בנוסף האתחל משחק גם פותח לי חלון ובו הוא בונה את הלוח והכפתורים , %% ואז מגיעים למצב שאפשר לשחק

אתחל\_משחק,

%% הלולאת חזור היא זו שממשיכה את המשחק עד שהסוף משחק מצליח

חזור,

%% 2. המצב הנוכחי לוקח את המצב הנוכחי והופך אותו לנגיש לשאר החוקים בתור %%... משתנה \_מצב

מצב\_נוכחי(\_מצב),

%% 3. הקבל מהלך מחכה לקבל מהלך מהפונקציה הדינמית (הפונקציה הדינמית בעצם %% ממירה את הלחיצות על הלוח למהלכים) וברגע שהוא מקבל אותו הוא הוא בודק %% האם הוא חוקי , אם הוא חוקי הוא מעביר אותו הלאה ואם לא הוא דורש מהלך חדש

קבל\_מהלך(\_מצב,\_מהלך),

%% 4. הבצע מהלך לוקח את המהלך מהקבל מהלך ומבצע אותו , כלומר משנה את המצב %% למצב לאחר ביצוע המהלך

בצע\_מהלך(\_מצב, \_מהלך, \_מצב\_חדש),

%% 5. שמירת המצב הקודם נועדה לכפתור חזור והעדכן מצב נועד על מנת להפוך את %% המצב לאחר ביצוע המהלך למצב הנוכחי שעליו אנו עובדים

שמירת\_מצב\_קודם(\_מצב),

עדכן\_מצב(\_מצב\_חדש),

%% 6. ההדפס מצב והעדכן גרפי נועדו להציג למשתמש את המצב לאחר ביצוע המהלך

הדפס\_מצב(\_מצב\_חדש),

עדכן\_גרפי(\_מצב\_חדש),

%% 7. בדוק סוף משחק נועד לבדוק אם המשחק יסתיים בין אם זה על ידי כך שמישהו %% מנצח או על ידי כך שנלחץ כפתור היציאה , אם המשחק מסתיים הלולאת משחק %% עוברת לסור משחק אבל אם זה לא סוף המשחק הבדוק סוף משחק נכשל ואז %%.. הלולאת משחק חוזרת ל'חזור'

בדוק\_סוף\_משחק(\_מצב\_חדש, \_מנצח),!,

%% 8. הסגור משחק סוגר את המשחק ומציג הודעה בהתאם

סגור\_משחק(\_מנצח).

**4.4 – שלבי פיתוח המשחק**

טקסטואלי לשני שחקנים אנושיים

גרפי לשני שחקנים אנושיים

גרפי לשחקן אנושי וגרפי לשחקן ממוחשב בעל רמת בינות

הוספת מינמקס לבינות

**4.5 – רמות הבינה השונות**

בינה 0 : זוהי לא בדיוק בינה , המחשב מבצע מהלכים אקראיים

בינה 1 : זוהי בינה שבה המחשב מבצע מהלכים אקראיים אך אם הוא רואה סוף משחק הוא מבצע אותו

בינה 2 : זוהי בינה שעובדת על מספר המקומות שתופס השחקן על הלוח , ככל שהשחקן יתפוס יותר מקום כך הציון שלו יהיה יותר גבוה

בינה 3 : זוהי הבינה הכי חכמה שהיא לא מינימקס , בעצם הבינה לוקחת המהלך שיגרום לכך שיהיה כמה שפחות מהלכים ליריב (אצלי מהלך זה בעצם תפיסת משבצת).

בינה מינימקס : זוהי בעצם הבינה הכי חכמה , הבינה לוקחת את בינה 3 ומחפשת איתה לעומק את המהלך הכי טוב .

**4.6 – התוספת שלי בפרוייקט**

הדבר הראשון שהוספתי בעצמי היה האלגוריתם ללוח משושה שיצרתי בעצמי שזוהי סדרה של כמה חוקים היוצרים משבצות חוקיות ולא חוקיות ככה שיווצר לוח משושה עם הכיוונים שרציתי.

בנוסף יצרתי את מערך החוקים של איך מציבים את הכלי (איזה משבצות לא חוקיות הוא יוצר) ואיך הוא נראה על הלוח.

בנוסף לחוקים אני יצרתי את התמונות לבד של הכלים והלוח.

**פרק חמישי – תיאור מנגנון ההיסק.**

הסבר כללי:

מערכת ההיסק היא מנגנון המסוגל להסיק מסקנות ע"פ הידע הקיים בבסיס הידע. תפקידה, בדומה למומחה אנושי לנתח את הידע הקיים ולהגיע בעזרתו למסקנות הנכונות.

על מערכת ההיסק להשתמש בחוקים ובעובדות הקשורות ליעדה על מנת להגיע למסקנה. מערכת ההיסק בדומה למערכת ההסבר אינה תלויה בבסיס הידע ולכן שייכת למעטפת הכללית של המערכת המומחית. ישנם שינויים קלים אשר יכולים להתאים מערכות היסק שונות בצורה טובה יותר למומחיות מסוימת, אך באופן כללי, המערכת אמורה לפעול על כל בסיס ידע.

מהי מערכת ההיסק ומטא אינטרפטר?

מערכת ההיסק היא למעשה מטא אינטרפטר,דהיינו תכנית העוברת ועוקבת אחר תוכנית אחרת, מנגנון מפרש. המיוחד במנגנון זה הוא עצם כתיבת המפרש באותה שפה בה כתובה התכנית עצמה. המערכת פועלת ע"פ שיטת **Backward Chaining**, כאשר בשיטה זו ההיסק מתחיל מן המטרה עצמה ומסתיים בעובדות הקיימות בבסיס הידע. מנגנון ההיסק משתמש בחוקים על מנת למצוא עובדות המוכיחות את הצלחת המטרה.

המטרה:

מטרת מנגנון ההיסק היא להוכיח יחסים בין חוקים ועובדות על ידי חיקוי הפעולות של התכנית בפרולוג, ולבסוף להציגן למשתמש.

מרכיבי מערכת ההיסק:

המערכת מבוססת על החוק **הוכחה/2** המשתמש בחוק **פסוק/2.**

הסבר לחוק פסוק/2:

המתאר **פסוק( \_ראש\_החוק, \_גוף\_החוק )** מקבל ראש חוק ומחזיר את גופו.

לדוגמא אם קיים החוק:

**ניסיון(\_ניסוי,\_ניסוי2):-**

**\_ניסוי<5,**

**\_ניסוי2>40.**

עבור הפעלת החוק  **פסוק(ניסיון, \_ביצוע\_ניסוי)** נקבל

\_ביצוע\_ניסוי = ( \_ניסוי<5, \_ניסוי2>40).

הסבר החוק **הוכחה/2**

**הוכחה מטרה, הסבר( \_מטרה, \_גוף, \_הסבר\_גוף ) ):-**

**פסוק( \_מטרה, \_גוף ),**

**המשך\_הוכחה( \_גוף, \_הסבר\_גוף ).**

החוק **הוכחה/2** יקבל את המטרה לה אנו רוצים לקבל הסבר, יפרק אותה **לראש וגוף** בעזרת המתאר **פסוק/2**, לאחר מכן ימשיך את החוק, את ההוכחה על הגוף ויחזיר הסבר בהתאם, הסבר זה יכנס לתוך רשימת ההסבר שתתורגם לשפה טבעית דרך מנגנון ההסבר.

טיפול בעובדה:

**המשך\_ההוכחה( true, [] ):-!**

במקרה שהוחזר True סימן שההוכחה הגיעה לעובדה.

טיפול במטרה חסומה:

**המשך\_הוכחה(\_מטרה\_בודדת,[]):-**

**מטרה\_חסומה(\_מטרה\_בודדת),!,**

**\_מטרה בודדת.**

חוק זה מטפל במקרה שישנו חלק בתוכנית **שאין אנו רוצים להציג את ההסבר שלו**.

החוק יחזיק רשימה ריקה. שרשרת ההיסק תסתיים אבל המטרה תתבצע.

דוגמא:

מטרה\_חסומה(!).

מטרה\_חסומה(אקראי(\_)).

טיפול בתנאי לוגי,בביטויים חשבוניים ובמתארי רשימות:

**הוכחה( \_מטרה, הסבר(\_מטרה, \_מסוג, \_ההסבר ) ):-**

פסוק\_מסוג(\_מטרה, \_מסוג, \_הסבר),!,

\_מטרה.

חוק זה מטופל במתארי רשימות, ביטויים חשבוניים וביטויים לוגיים אשר מוגדרים עם הסבר מראש ואשר אין צורך לכלול את הסברם ברשימת ההסבר. סימן ה- ! מונע מן החוק למצוא את המטרה בדרך אחרת.

המתאר **פסוק\_מסוג/3** יכיל את כל הביטויים הלוגיים החשבוניים ומתארי הרשימה והסברם.

דוגמאות:

**טיפול בביטוי חשבוני.**

**פסוק\_מסוג(\_תוצאה הוא \_חישוב, חשבוני, ['מחישוב מקבלים ש',\_תוצאה1,' הוא ',\_חישוב1]):-**

**\_תוצאה הוא \_חישוב,**

זו התוספת שלי להוכחה.

**תרגם\_מספר\_לערך\_תווי(\_תוצאה,\_תוצאה1),**

**תרגם\_ביטוי\_לערך\_תווי(\_חישוב,\_חישוב1).**

ביתר פירוט:

**תרגם\_ביטוי\_לערך\_תווי(\_חישוב,\_חישוב1):-**

מספר(\_חישוב),

תרגם\_מספר\_לערך\_תווי(\_חישוב,\_חישוב1).

**תרגם\_ביטוי\_לערך\_תווי(\_חישוב,\_חישוב1):-**

פנקטור(\_חישוב,\_פעולה,2),

תרגם\_פעולה\_לערך\_תווי(\_פעולה,\_פעולה1),

ארגומנט(1,\_חישוב,\_חלק\_ראשון),

תרגם\_ביטוי\_לערך\_תווי(\_חלק\_ראשון,\_חלק\_ראשון1),

ארגומנט(2,\_חישוב,\_חלק\_שני),

תרגם\_ביטוי\_לערך\_תווי(\_חלק\_שני,\_חלק\_שני1),

שרשר\_מחרוזות(['( ',\_חלק\_ראשון1, \_פעולה1 ,\_חלק\_שני1,' )'],\_חישוב1),!.

תרגם\_פעולה\_לערך\_תווי(+,' + ').

תרגם\_פעולה\_לערך\_תווי(-,' – ').

תרגם\_פעולה\_לערך\_תווי(\*,' \* ').

תרגם\_פעולה\_לערך\_תווי(/,' / ').

תרגם\_פעולה\_לערך\_תווי(\_,' # ').

**טיפול בביטויים לוגיים.**

פסוק\_מסוג(\_א = \_ב,בהשוואה,[\_א,'שווה ל',\_ב]).

פסוק\_מסוג(\_א \= \_ב,בהשוואה,[\_א,'שונה מ',\_ב]).

פסוק\_מסוג(\_א =< \_ב,בהשוואה,[\_א,'קטן או שווה ל',\_ב]).

פסוק\_מסוג(\_א < \_ב,בהשוואה,[\_א,'קטן מ',\_ב]).

פסוק\_מסוג(\_א > \_ב,בהשוואה,[\_א,'גדול מ',\_ב]).

פסוק\_מסוג(\_א >= \_ב,בהשוואה,[\_א,'גדול או שווה ל',\_ב]).

פסוק\_מסוג(\_א =:= \_ב,בהשוואה,[\_א,'ו',\_ב,'בעלי אותו ערך']).

**טיפול במתארי רשימה**

פסוק\_מסוג(חבר(\_איבר,\_רשימה),מתארי\_רשימה, ['האיבר',\_איבר,'חבר ברשימה' \_רשימה]).

פסוק\_מסוג(שרשר(\_א,\_ב,\_רשימה),מתארי\_רשימה, ['משרשור עולה ש',\_א,'ו',\_ב,'בונים את',\_רשימה])

פסוק\_מסוג(ראשון\_ברשימה(\_א,\_רשימה),מתארי\_רשימה,['האיבר',\_א,'נמצא במקום הראשון ברשימה' \_רשימה]).

פסוק\_מסוג(מספר\_איברים(\_מ,\_רשימה),מתארי\_רשימה, [\_מ,'הוא מספר האיברים ברשימה',\_רשימה]).

פסוק\_מסוג(מצא\_כל(\_מ,\_תנאי,\_רשימה),מתארי\_רשימה,['מהפעלת מצא\_כל מתקבלת הרשימה',\_רשימה]).

**טיפול בטענת שלילה:**

פסוק\_מסוג(לא (\_מטרה),שלילה,['אי אפשר להסיק ש',\_מטרה]).

טענת השלילה תטופל ביחד עם החוק האחראי לטיפול במתאר פסוק\_מסוג/3 כאשר ההסבר יהיה "לא ניתן להסיק ש" ולאחר מכן הסבר הוכחת המטרה.

**חוק המטפל במתארי מערכת:**

פסוק\_מסוג(\_מטרה,מערכת,[\_מטרה,'היא פועלת מערכת']):-

\_מטרה\=(\_,\_),

לא (פסוק(\_מטרה,\_)).

כאשר פסוק אינו מצליח לפרק חוק לראש וגוף אזי החוק הינו פעלת מערכת. לדוגמא "קרא", "כתוב", "פתח\_חלון" וכו'

(גם !).

**טיפול בגוף חוק מורכב:**

המשך\_הוכחה((\_מטרה\_מבודדת,\_שאר),\_הסבר\_שאר):-

מטרה\_חסומה(\_מטרה\_מבודדת),!,

\_מטרה\_מבודדת,

המשך\_הוכחה(\_שאר,\_הסבר\_שאר).

המשך\_הוכחה((\_מטרה\_מבודדת,\_שאר),[\_הסבר\_מטרה|\_הסבר\_שאר]):-

הוכחה(\_מטרה\_מבודדת,\_הסבר\_מטרה),

המשך\_הוכחה(\_שאר,\_הסבר\_שאר).

המשך\_הוכחה(\_מטרה\_מבודדת,[\_הסבר\_מטרה]):-

הוכחה(\_מטרה\_מבודדת,\_הסבר\_מטרה).

טיפול בגוף חוק מורכב מקבל **רשימה עם סוגריים עגולים** כאשר **הפסיק ברשימה עם הסוגריים העגולים משמש כהפרד (ברשימה רגילה)**, כלומר מוציא מטרה אחת מבין כל המטרות, לאחר מכן בודק אם נמצאת בעבודה **מטרה\_חסומה/1** במקרה שכן לא מוחזר הסבר ושרשרת ההיסק תמשיך על שאר המטרות, במקרה שהמטרה אינה נמצאת באסורים, תמשך ההוכחה עליה דרך **המשך\_הוכחה/2** וישורשר ההסבר שלה בהתאם.

**פרק שישי – תיאור מנגנון ההסבר.**

**כתוב\_יפה(\_מטרה):-**

מילול(\_מטרה,\_מילולי),!,

כתוב\_רשימה(\_מילולי).  
  
חוק זה אחראי לתרגם את המטרה בצורה טבעית, כאשר התרגום יהיה באמצעות מילול/2.

דוגמא **למילול/2:**  
**מילול(אחזר\_תור(\_,\_תור),['כעת תורו של השחקן ה',\_צבע]):-**

תרגם\_מספר\_לצבע(\_תור,\_צבע).

לאחר הפעלת חוק זה יוחזר תורו של איזה שחקן (צבע השחקן הנוכחי).

**%הצג\_הוכחה( הסבר( \_מטרה, \_גוף\_או\_סוג, \_הסבר )).**

**הצג\_הוכחה(הסבר(\_עובדה,true,[]),\_תשובה):-**

כתוב\_יפה(\_עובדה),

כתוב\_בחלון(' נתון כעובדה'),

שורה\_בחלון,

כתוב\_בחלון('חזור/סוף'),שורה\_בחלון,

קרא\_בחלון(\_תשובה),

!.

חוק זה מטפל במקרה של עובדה, תחילה תיבדק העובדה אם קיים מילול עבורה ולאחר מכן יודפס אוטומטית בחלון "נתון כעובדה".

**הצג\_הוכחה(הסבר(\_פסוק\_מסוג,\_סוג,\_הסבר),\_תשובה):-**

פסוק\_מסוג(\_פסוק\_מסוג,\_סוג,\_הסבר),

כתוב\_רשימה\_של\_הוכחה (\_הסבר),

שורה\_בחלון,

כתוב\_בחלון('חזור/סוף'),שורה\_בחלון,

קרא\_בחלון(\_תשובה),!.

במידה וזהו פסוק הסבר מסוג מסוים יודפס ההסבר לאותו סוג.

**הצג\_הוכחה(הסבר(\_מטרה,\_גוף,\_הסבר),\_תשובה\_מוחזרת):-**  
 חזור,

כתוב\_יפה(\_מטרה),

כתוב\_בחלון(' כי:'),שורה\_בחלון,

כתוב\_תת\_מטרות(\_הסבר,1),

כתוב\_בחלון('מספר/חזור/סוף'),שורה\_בחלון,

קרא\_בחלון(\_תשובה),

המשך\_הצג\_הוכחה(\_תשובה,\_הסבר,\_תשובה\_מוחזרת),!.

במקרה של הסבר מורכב יודפס ראש החוק ואחריו הגוף בצורה מסודרת, החוק כתוב\_תת\_מטרות אחראי למיספור וסידור החוקים הנמצאים בגוף.

כתוב\_תת\_מטרות([],\_):-שורה\_בחלון.

כתוב\_תת\_מטרות([הסבר(\_תת\_מטרה,\_,\_)|\_שאר],\_מספר):-  
 כתוב\_בחלון(\_מספר),

כתוב\_בחלון(') '),

כתוב\_יפה(\_תת\_מטרה),

שורה\_בחלון,

\_מספר\_פלוס הוא \_מספר + 1,

כתוב\_תת\_מטרות(\_שאר,\_מספר\_פלוס).

חוק זה אחראי לכתיבת כל החוקים של גוף הוכחה בצורה מסודרת וממוספרת.

כתוב\_רשימה\_של\_הוכחה([]).

כתוב\_רשימה\_של\_הוכחה([סמן\_מיוחד(שורה)|\_שאר]):-

שורה\_בחלון,

כתוב\_רשימה\_של\_הוכחה(\_שאר).

כתוב\_רשימה\_של\_הוכחה([סמן\_מיוחד(רווח(\_רווח))|\_שאר]):-

רווח\_בחלון(\_רווח),

כתוב\_רשימה\_של\_הוכחה(\_שאר).

כתוב\_רשימה\_של\_הוכחה([\_מילה|\_שאר]):-\_מילה \= סמן\_מיוחד(\_),

לא מספר(\_מילה),

כתוב\_בחלון(\_מילה),

כתוב\_רשימה\_של\_הוכחה(\_שאר).

כתוב\_רשימה\_של\_הוכחה([\_מילה|\_שאר]):-\_מילה \= סמן\_מיוחד(\_),

תרגם\_מספר\_לערך\_תווי(\_מילה,\_מילה\_תווי),

כתוב\_בחלון(\_מילה\_תווי),

כתוב\_רשימה\_של\_הוכחה(\_שאר).

חוק הכותב רשימה של מילול בחלון על ידי פירוק כל מילה והדפסתה.

המשך\_הצג\_הוכחה(ח,\_,ח).

המשך\_הצג\_הוכחה(ס,\_,ס).

המשך\_הצג\_הוכחה(\_מספר,\_הסבר,\_תשובה):-

מיקום\_איבר(\_מספר,\_תת\_הסבר,\_הסבר),

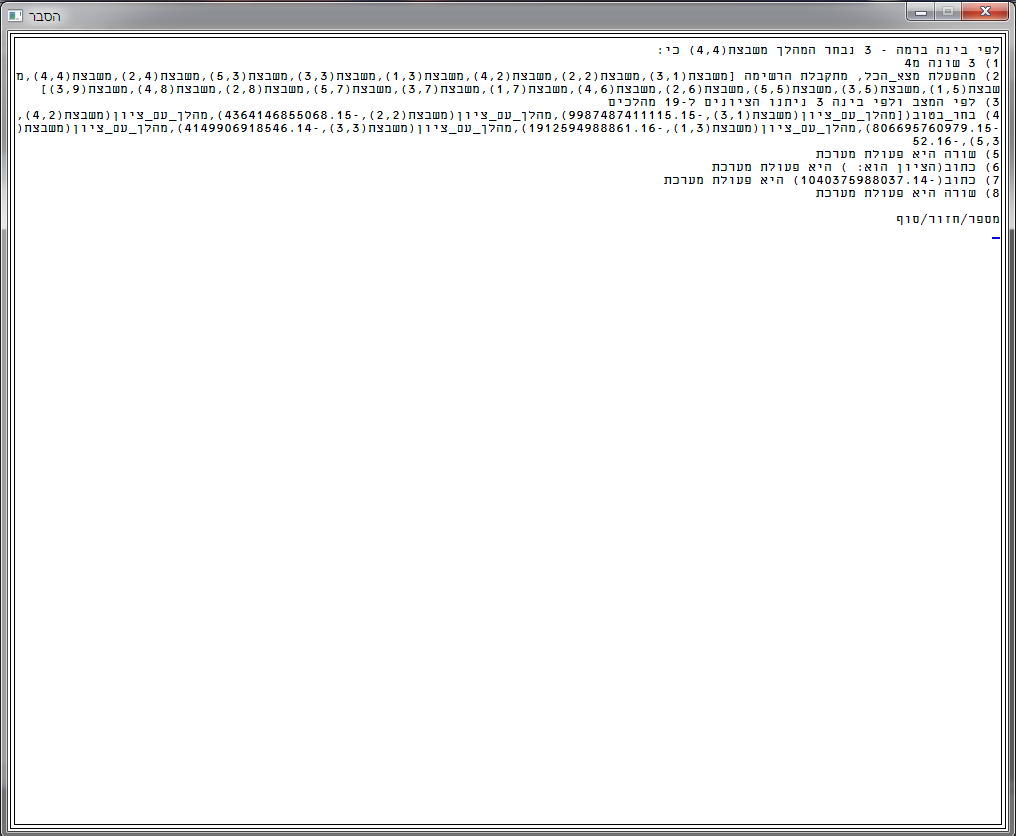
הצג\_הוכחה(\_תת\_הסבר,\_תשובה),

\_תשובה = 'ס'.  
חוק האחראי להמשך הדפסת התפריט בהתאם לבחירתו של המשתמש.

**דוגמא למנגנון ההסבר בפרוייקט שלי:**

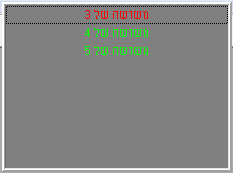
זוהי דוגמא לחלון ההוכחה/הסבר במשחק שלי:

בדוגמא זו רואים איך המשחק הגיע למתן ציון למהלך.



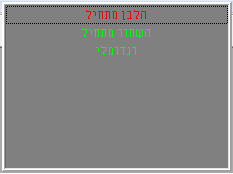
**פרק 7 – תיאור ממשק פלט/קלט**

מסך ראשון:



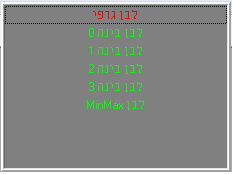
במסך זה יהיה על המשתמש לבחור את גודל הלוח בו הם רוצים לשחק.

מסך שני:



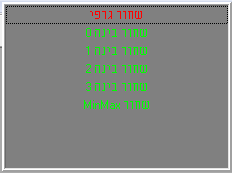
במסך זה יהיה על המשתמש לבחור מי מתחיל , לבן , שחור או רנדומלי.

מסך שלישי:



במסך זה יהיה על המשתמש לבחור האם הכלי הלבן יהיה שחקן אנושי או מחשב בעל רמות בינה שונות.

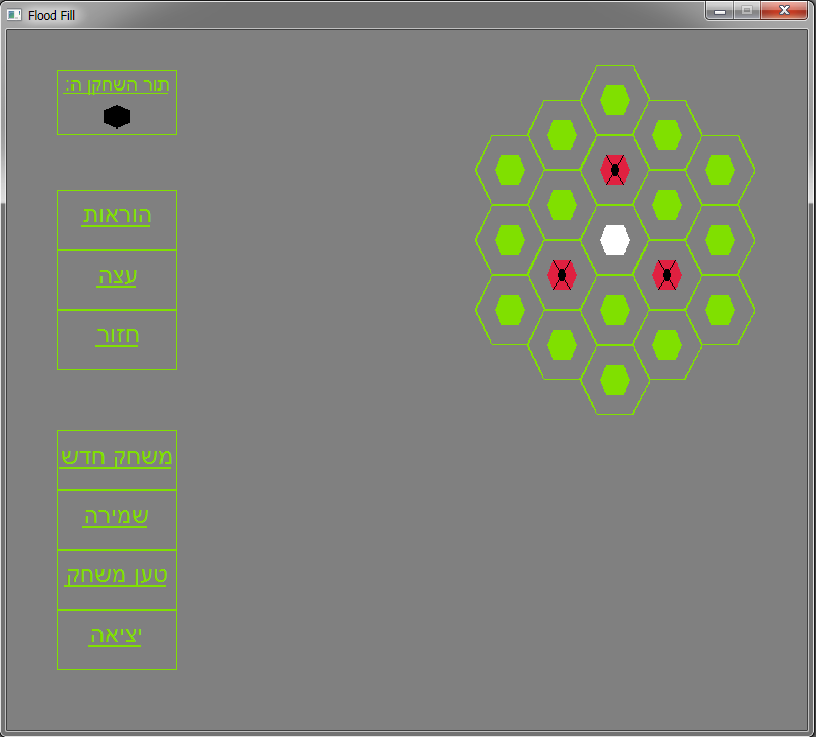
מסך רביעי:



במסך זה יהיה על המשתמש לבחור האם הכלי השחור יהיה שחקן אנושי או מחשב בעל רמות בינה שונות.

מסך המשחק:

פלט המחזיר את תור איזה שחקן



כפתור קלט המציג את הוראות המשחק

כפתור קלט הנותן עצה והסבר לה

כפתור קלט החוזר למצב הקודם

כפתור קלט היוצר משחק חדש

כפתור קלט השומר את המשחק

כפתור קלט הטוען את המשחק השמור

כפתור קלט הגרום ליציאה

מסך הפלט/קלט , לחיצה על הלוח תגרום לקליטת מיקום השחקן ולהצגתו על הלוח

**פרק 8 – סיכום – הרחבות ושיפורים אפשריים**

**8.1 – הרחבות שבוצעו במערכת**

כל מהלך הלמידה שלי התבצע לאורך כל השנה , בכל פעם אשר נדרשתי להכין חלק אחר מהפרויקט למדתי את הידע הדרוש לי להכין אותו. כמובן שרוב הידע התבסס על ידע קודם הנלמד לפני זה בכיתה י''א אך לפעמים היה הצורך ללמוד גם חומר חדש.

הידע שאני יצרתי בעצמי היה פיתוח אלגוריתם המאפשר את בניית הלוח משושה , זהו אלגוריתם שישבתי עליו זמן רב עד שלבסוף פתרתי אותו ללא שום עזרה חיצונית.

**8.2 – הצעות לשיפורים בעתיד**

לפי דעתי סביבת העבודה היא קצת בעייתית ויש לה המון באגים , אם בעתיד תהיה סביבת עבודה טובה יותר זה יקל על תהליך העבודה.

**8.3 – רפלקציה**

לפי דעתי המשחק הוא משחק מאוד חכם שיכול להביס כל אחד. היה נחמד מאוד לבצע את הפרויקט גם אם מידי פעם היה לחץ רב או המון בעיות שאנו לא ידענו כל כך איך להתעסק איתם , בסופו של דבר כאשר הצלחתי לפתור משהו תמיד זה היה מלווה בתחושה טובה.

**פרק 9 – נספחים**

**9.1 – הוראות הפעלה**

על מנת להתחיל את המשחק יש לפתוח את תוכנת ה – wideA ,לאחר מכן ללחוץ על כפתור ה – Listener וללחוץ על start . ברגע זה נפתח חלון הפקודות , כעת יש לטעון את קבצי המשחק וזאת על ידי לחיצה על כפתור ה – Listener שוב וללחוץ על Consult , כעת יפתח חלון אשר בו צריך לבחור את הקובץ אותו אתם רוצים לטעון , אצלנו זה יהיה קובץ ה – main בפרויקט. לאחר טעינת הקובץ תופיע הודעה ' הקבצים נטענו בהצלחה '. כעת צריך לרשום 'משחק.' בחלון , מה שיריץ את לולאת המשחק ויפתח לנו את התפריט הראשון. לאחר הבחירות בתפריטים לפי רצונם של המשתמשים המשחק יעלה ויהיה אפשר להתחיל לשחק בו.

**9.2 – ביבליוגרפיה - אתרי אינטרנט וספרים**

התוכנה של זיליון בה מצאתי ולמדתי את המשחק.

**קבצי המשחק:**

**0-Main:**

%% קובץ ראשי %%

%% ========== %%

%% %%

%% כותב: עידן משיח %%

%% ====================== %%

%% החוק טוען לזיכרון את כל הקבצים הדרושים להתחלת המשחק.

:-סע,

טען\_והוסף('list\_Proj.pro'),

טען\_והוסף('1-Game\_Loop.pro'),

טען\_והוסף('2-Data\_Retrieve.pro'),

טען\_והוסף('3-Init\_Game.pro'),

טען\_והוסף('4-Get\_Move.pro'),

טען\_והוסף('5-Excecute\_Move.pro'),

טען\_והוסף('6-Graphics.pro'),

טען\_והוסף('7-End\_Game.pro'),

טען\_והוסף('8-DynamicFunction'),

טען\_והוסף('9-MinMax'),

טען\_והוסף('10-exp'),

הודעה('הקבצים נטענו בהצלחה').

**1-Game\_Loop:**

%% לולאת משחק

%% ----------

%%

%% כותב: עידן משיח.

%% החוק הראשי:

%% ===========

משחק:-

אתחל\_משחק,

חזור,

מצב\_נוכחי(\_מצב),

קבל\_מהלך(\_מצב,\_מהלך),

בצע\_מהלך(\_מצב, \_מהלך, \_מצב\_חדש),

שמירת\_מצב\_קודם(\_מצב),

עדכן\_מצב(\_מצב\_חדש),

הדפס\_מצב(\_מצב\_חדש),

עדכן\_גרפי(\_מצב\_חדש),

בדוק\_סוף\_משחק(\_מצב\_חדש, \_מנצח),!,

סגור\_משחק(\_מנצח).

**2-Data\_Retrieve:**

%% איחזור מידע

%% -----------

%% כותב : עידן משיח.

%% אחזר\_פרמטר\_כלשהו(\_מצב, \_פרמטר\_מוחזר).

%%=======================================

אחזר\_תור\_מי(מצב(\_תור\_מי, \_), \_תור\_מי).

אחזר\_לוח(מצב(\_, \_לוח), \_לוח).

אחזר\_כלי\_מלוח(\_לוח,\_כלי):-

חבר(\_כלי, \_לוח).

אחזר\_צבע\_של\_כלי(כלי(\_צבע,\_), \_צבע).

אחזר\_משבצת\_מכלי(כלי(\_, \_משבצת), \_משבצת).

אחזר\_שורה\_ממשבצת(משבצת(\_שורה, \_), \_שורה).

אחזר\_טור\_ממשבצת(משבצת(\_, \_טור), \_טור).

אחזר\_תיאור\_רכיב(תיאור\_רכיב(\_תיאור, \_רכיב)):-

תיאור\_רכיב(\_תיאור, \_רכיב).

אחזר\_תיאור\_מתיאור\_רכיב( תיאור\_רכיב(\_תיאור, \_), \_תיאור).

אחזר\_רכיב\_מתיאור\_רכיב( תיאור\_רכיב(\_, \_רכיב) , \_רכיב).

**3-Init\_Game:**

%% אתחל משחק

%% ---------

%% כותב : עידן משיח.

%% -----------------

אתחל\_משחק:-

הסרה\_קודמת,

קבע\_גודל\_לוח,

קבע\_מי\_מתחיל,

קבע\_סוג\_בינה\_לבן,

קבע\_סוג\_בינה\_שחור,

%% פתח\_חלון\_בשם(\_איקס\_עליון\_ימין, \_וואי\_עליון\_ימין , \_איקס\_תחתון\_שמאל, \_וואי\_תחתון\_שמאל, \_צבע\_טקסט, \_סוג\_מסגרת, \_שם\_חלון),

פתח\_חלון\_בשם(100, 100, 900, 800, 16, 162, 'Flood Fill'),

מצב\_התחלתי(\_מצב),

עדכן\_מצב(\_מצב),

הדפס\_מצב(\_מצב),

הוסף(מצב\_שמור(\_מצב)),

הוסף(מצב\_קודם(\_מצב)),

הצג\_מצב(\_מצב),

1=1.

הסרה\_קודמת:- הסר\_כל(גודל\_לוח(\_, \_)),

הסר\_כל(כלי(\_, \_)),

הסר\_כל(סוג\_שחקן(\_, \_)),

הסר\_כל(מצב\_נוכחי(\_)),

הסר\_כל(נלחץ\_יציאה),

הסר\_כל(תיאור\_רכיב(\_, \_)),

הסר\_כל(נקלט\_מהלך(\_)).

%%===========================תפריט================================%%

%% תפריט( שיעור\_X\_פינה\_ימנית\_עליונה , שיעור\_Y\_פינה\_ימנית\_עליונה,

%% שיעור\_X\_פינה\_שמאלית\_תחתונה , שיעור\_Y\_פינה\_שמאלית\_תחתונה,

%% צבע\_הרקע\_של\_התפריט\_והכתב , צבע\_הרקע\_של\_פס\_הבחירה\_והכתב\_על\_רקע\_הבחירה,

%% רשימת\_האפשרויות\_מהן\_יש\_לבחור, \_הבחירה ).

%%===============================================================================

קבע\_גודל\_לוח:-

תפריט(570, 30, 800, 200, 138, 139, ['משושה של 3', 'משושה של 4', 'משושה של 5'], \_בחירה),

הסר\_כל(גודל\_לוח(\_,\_)),

הוסף\_סוג\_לוח(\_בחירה).

קבע\_מי\_מתחיל:-

תפריט(570, 30, 800, 200, 138, 139, ['הלבן מתחיל', 'השחור מתחיל', 'רנדומלי'], \_בחירה),

הסר\_כל(מצב\_התחלתי( מצב(\_,\_))),

הוסף\_תור\_מי(\_בחירה).

קבע\_סוג\_בינה\_לבן:-

תפריט(570, 30, 800, 200, 138, 139, ['לבן גרפי','לבן בינה 0', 'לבן בינה 1', 'לבן בינה 2','לבן בינה 3','MinMax לבן'], \_בחירה),

הסר\_כל(סוג\_שחקן(\_,\_)),

הוסף\_סוג\_בינה\_לבן(\_בחירה).

קבע\_סוג\_בינה\_שחור:-

תפריט(570, 30, 800, 200, 138, 139, ['שחור גרפי','שחור בינה 0', 'שחור בינה 1', 'שחור בינה 2','שחור בינה 3','MinMax שחור'], \_בחירה),

הוסף\_סוג\_בינה\_שחור(\_בחירה).

הוסף\_סוג\_לוח('משושה של 3'):-

הוסף(גודל\_לוח( 9, 5)),

1 = 1.

הוסף\_סוג\_לוח('משושה של 4'):-

הוסף(גודל\_לוח( 13, 7)),

1 = 1.

הוסף\_סוג\_לוח('משושה של 5'):-

הוסף(גודל\_לוח( 17, 9)),

1 = 1.

הוסף\_תור\_מי('הלבן מתחיל'):-

הוסף(מצב\_התחלתי( מצב( לבן ,[] ))),

1 = 1.

הוסף\_תור\_מי('השחור מתחיל'):-

הוסף(מצב\_התחלתי( מצב( שחור ,[] ))),

1 = 1.

הוסף\_תור\_מי('רנדומלי'):-

הגרל\_מספר(2,\_מספר),

מספר\_לצבע(\_מספר,\_צבע),

הוסף(מצב\_התחלתי( מצב( \_צבע ,[]))),

1 = 1.

הוסף\_סוג\_בינה\_לבן('לבן גרפי'):-

הוסף(סוג\_שחקן(לבן, אנושי(גרפי))),

1=1.

הוסף\_סוג\_בינה\_לבן('לבן בינה 0'):-

הוסף(סוג\_שחקן(לבן , מחשב(0))),

1=1.

הוסף\_סוג\_בינה\_לבן('לבן בינה 1'):-

הוסף(סוג\_שחקן(לבן , מחשב(1))),

1=1.

הוסף\_סוג\_בינה\_לבן('לבן בינה 2'):-

הוסף(סוג\_שחקן(לבן , מחשב(2))),

1=1.

הוסף\_סוג\_בינה\_לבן('לבן בינה 3'):-

הוסף(סוג\_שחקן(לבן , מחשב(3))),

1=1.

הוסף\_סוג\_בינה\_לבן('MinMax לבן'):-

הוסף(סוג\_שחקן(לבן , מחשב(4))),

1=1.

הוסף\_סוג\_בינה\_שחור('שחור גרפי'):-

הוסף(סוג\_שחקן(שחור, אנושי(גרפי))),

1=1.

הוסף\_סוג\_בינה\_שחור('שחור בינה 0'):-

הוסף(סוג\_שחקן(שחור , מחשב(0))),

1=1.

הוסף\_סוג\_בינה\_שחור('שחור בינה 1'):-

הוסף(סוג\_שחקן(שחור , מחשב(1))),

1=1.

הוסף\_סוג\_בינה\_שחור('שחור בינה 2'):-

הוסף(סוג\_שחקן(שחור , מחשב(2))),

1=1.

הוסף\_סוג\_בינה\_שחור('שחור בינה 3'):-

הוסף(סוג\_שחקן(שחור , מחשב(3))),

1=1.

הוסף\_סוג\_בינה\_שחור('MinMax שחור'):-

הוסף(סוג\_שחקן(שחור , מחשב(4))),

1=1.

הגרל\_מספר( \_גבול\_עליון, \_מספר):- אקראי(\_מספר\_בין\_0\_ל\_1),

\_מספר\_ממשי הוא \_מספר\_בין\_0\_ל\_1 \* \_גבול\_עליון + 1,

קטום(\_מספר\_ממשי, \_מספר).

מספר\_לצבע(1,שחור).

מספר\_לצבע(2,לבן).

%% גודל\_לוח(\_שורות, \_טורים)

%%==========================

%%גודל\_לוח( 9 , 5). %% משושה של שלוש

%%גודל\_לוח( 13, 7). %% משושה של ארבע

%%גודל\_לוח( 17, 9). %% משושה של חמש

%%מצב\_התחלתי( מצב( \_תור מי,\_רשימת\_כלים)).

%%=======================================

%% עדכן\_מצב(\_מצב).

%% ===============

%% חוק זה טוען לזכרון את העובדה: מצב\_נוכחי(\_מצב).

עדכן\_מצב(\_מצב):-

הסר\_כל(מצב\_נוכחי(\_)),

הוסף(מצב\_נוכחי(\_מצב)).

**4-Get\_Move:**

%% קבל מהלך

%% --------

%%

%% כותב : עידן משיח.

צבע(שחור).

צבע(לבן).

%% סוג\_שחקן(\_שם, \_סוג\_וממשק).

%%===========================

%% סוג\_שחקן(לבן, אנושי(טקסט)).

%% סוג\_שחקן(שחור, אנושי(טקסט)).

%% סוג\_שחקן(לבן, אנושי(גרפי)).

%% סוג\_שחקן(שחור, אנושי(גרפי)).

%% סוג\_שחקן(לבן , מחשב(0)).

%% סוג\_שחקן(שחור, מחשב(0)).

%% סוג\_שחקן(לבן , מחשב(1)).

%% סוג\_שחקן(שחור, מחשב(1)).

%% סוג\_שחקן(לבן , מחשב(2)).

%% סוג\_שחקן(שחור, מחשב(2)).

%% סוג\_שחקן(לבן , מחשב(3)).

%% סוג\_שחקן(שחור, מחשב(3)).

%% סוג\_שחקן(לבן , מחשב(4)).

%% סוג\_שחקן(שחור, מחשב(4)).

%%======================================================

%%=============== החוק הראשי בקובץ זה: =================

%%======================================================

קבל\_מהלך(\_מצב, \_מהלך):-

קבל\_מהלך\_מממשק\_שחקן(\_מצב, \_מהלך),

מהלך\_אפשרי(\_מצב, \_מהלך),

1 = 1, !.

%%======================================================

%%==================== לשחקן אנושי =====================

%%======================================================

קבל\_מהלך\_מממשק\_שחקן(\_מצב, \_מהלך):-

אחזר\_תור\_מי(\_מצב, \_שחקן),

סוג\_שחקן( \_שחקן , אנושי(\_ממשק)),

קבל\_מהלך\_מממשק( \_ממשק, \_מצב, \_מהלך ),!.

%%=====================================================

%======================================================

קבל\_מהלך\_מממשק(גרפי, \_מצב, \_מהלך ):-

אחזר\_תור\_מי(\_מצב, \_שחקן),

סוג\_שחקן( \_שחקן , אנושי(גרפי)),

הסר\_כל(נקלט\_מהלך(\_)),

חזור,

המתן(1),

נקלט\_מהלך(\_מהלך),!, %% התוכנית "תקועה" בלולאה עד אשר יופיע 'נקלט\_מהלך(\_מהלך)'

%% 'נקלט\_מהלך(\_מהלך)' - יופיע מהפונקציה הדינמית, כאשר היא "תרגיש" לחיצה.

%% \_מהלך יכול להיות מהלך משחק רגיל או

%% לחיצה על כפתור המלצה או לחיצה על כפתור יציאה.

מהלך\_אפשרי(\_מצב, \_מהלך).

%%=====================================================

%%=====================================================

%%=====================================================

קבל\_מהלך\_מממשק(טקסט, \_מצב, \_מהלך ):-

\_מהלך = מהלך(משבצת(\_שורה,\_טור)),

\_מצב = מצב(\_צבע, \_לוח),

שורה,

כתוב(' הקש מיקום יעד לכלי'),

שורה,

כתוב('שורה : '),

שורה,

קרא(\_שורה),

כתוב('טור: '),

שורה,

קרא(\_טור),

שורה, שורה.

%% מהלך ייחשב למהלך אפשרי אם נלחצה משבצת חוקית או נלחץ אחד הכפתורים המותרים.

%% ==========================================================================

%% מהלכים אפשריים...

%% ==================

מהלך\_אפשרי(\_מצב, יציאה).

מהלך\_אפשרי(\_מצב, המלצה).

מהלך\_אפשרי(\_מצב, משחק\_חדש).

מהלך\_אפשרי(\_מצב, הוראות).

מהלך\_אפשרי(\_מצב, שמירה).

מהלך\_אפשרי(\_מצב, חזור).

מהלך\_אפשרי(\_מצב, טען\_משחק).

%% מהלך ייחשב למהלך אפשרי אם נבחרה משבצת חוקית

%% ============================================

% מהלך\_אפשרי(מצב(\_צבע,\_לוח), \_מהלך)

%% ==================================

%% כאשר הוא מקבל מהלך מהטקסט

מהלך\_אפשרי(\_מצב, \_מהלך):-

\_מהלך = מהלך(\_משבצת),

מהלך\_חוקי(\_מצב, \_מהלך), !.

%% כאשר הוא מקבל מהלך מהגרפי

מהלך\_אפשרי(\_מצב, \_מהלך):-

\_מהלך = \_משבצת,

\_מהלך\_חדש = מהלך(\_משבצת),

מהלך\_חוקי(\_מצב, \_מהלך\_חדש).

% מהלך ייחשב למהלך חוקי אם המשבצת שנבחרה נמצאת על הלוח, אם משבצת במשבצת המוצא יש כלי מצבע החשקן ואם במשבצת המטרה אין אף כלי.

%% =========================================================================================================================

מהלך\_חוקי(\_מצב, \_מהלך):-

\_מצב = מצב( \_צבע, \_לוח ),

\_מהלך = מהלך(\_משבצת),

אחזר\_משבצת\_חוקית(\_משבצת),

לא חבר( כלי(\_ , \_משבצת), \_לוח ),

שחקן\_נגדי(\_צבע,\_צבע\_יריב),

לא כלי\_אויב\_צמוד(\_משבצת,\_צבע\_יריב, \_לוח).

%%=================================================================================================

כלי\_אויב\_צמוד(\_משבצת,\_צבע\_יריב, \_לוח):-

תוספת\_רגילה(\_כיוון,\_,\_),

משבצת\_שכנה\_בכיוון(\_משבצת, \_משבצת\_בבדיקה, \_כיוון),

חבר(כלי(\_צבע\_יריב , \_משבצת\_בבדיקה), \_לוח ).

%% בודק מהו הכיוון היחסי בין משבצת המוצא למשבצת המטרה. הכיוונים האפשריים הם: [1,2,3,4,6,7,8,9].

%% ============================================================================================

משבצת\_שכנה\_בכיוון(\_ממשבצת, \_משבצת\_שכנה, \_כיוון):-

לא משתנה(\_ממשבצת),

\_ממשבצת = משבצת(\_משורה, \_מעמודה),

\_משבצת\_שכנה = משבצת(\_שורה\_חדשה, \_עמודה\_חדשה),

תוספת\_רגילה(\_כיוון, \_תוספת\_לטור, \_תוספת\_לשורה),

\_שורה\_חדשה הוא \_משורה + \_תוספת\_לשורה,

\_עמודה\_חדשה הוא \_מעמודה + \_תוספת\_לטור.

משבצת\_שכנה\_בכיוון(\_ממשבצת, \_משבצת\_שכנה, \_כיוון):-

לא משתנה(\_משבצת\_שכנה),

\_ממשבצת = משבצת(\_משורה, \_מעמודה),

\_משבצת\_שכנה = משבצת(\_שורה\_חדשה, \_עמודה\_חדשה),

תוספת\_רגילה(\_כיוון, \_תוספת\_לטור, \_תוספת\_לשורה),

\_משורה הוא \_שורה\_חדשה - \_תוספת\_לשורה,

\_מעמודה הוא \_עמודה\_חדשה - \_תוספת\_לטור.

/\*

הכיוון מוגדר לפי המקלדת.

9 8 7

6 5 4

3 2 1

מסתכלים על תא מספר 5 ועל האופן בו מודפס הלוח שלנו.

הלוח ממוספר מימין לשמאל ומלמעלה כלפי מטה:

1 2 3 4 .....

1

2

3

.

.

לכן התוספות מתבצעות בצורה הבאה:

\*/

%================================================

%% תוספת(\_כיוון, \_תוספת\_לטור, \_תוספת\_לשורה).

%% =========================================

תוספת\_הפוכה(8, 0, -2).

תוספת\_רגילה(9, -1, -1).

תוספת\_הפוכה(3, -1, 1).

תוספת\_רגילה(2, 0, 2).

תוספת\_הפוכה(1, 1, 1).

תוספת\_רגילה(7, 1, -1).

%%%=============================================================

%% משבצת נחשבת לחוקית אם היא בגבולות הלוח.

%% ========================================

משבצת\_חוקית(משבצת(\_שורה, \_טור)):-

גודל\_לוח(\_שורות, \_טורים),

\_טורים\_בבדיקה הוא ((\_טורים+1)//2),

חלק\_ושארית(\_טורים\_בבדיקה, 2, \_,1),

בין\_מספרים(1, \_טור , \_טורים),

בין\_מספרים(1, \_שורה, \_שורות),

\_סכום הוא \_טור + \_שורה,

חלק\_ושארית(\_סכום, 2, \_, 0),

משבצת\_למשושה(משבצת(\_שורה,\_טור)), !.

משבצת\_חוקית(משבצת(\_שורה, \_טור)):-

גודל\_לוח(\_שורות, \_טורים),

\_טורים\_בבדיקה הוא ((\_טורים+1)//2),

חלק\_ושארית(\_טורים\_בבדיקה, 2, \_,0),

בין\_מספרים(1, \_טור , \_טורים),

בין\_מספרים(1, \_שורה, \_שורות),

\_סכום הוא \_טור + \_שורה,

חלק\_ושארית(\_סכום, 2, \_, 1),

משבצת\_למשושה(משבצת(\_שורה,\_טור)), !.

אחזר\_משבצת\_חוקית(משבצת(\_שורה, \_עמודה)):-

גודל\_לוח(\_שורות, \_עמודות),

בין\_מספרים(1 , \_שורה, \_שורות),

בין\_מספרים(1 , \_עמודה, \_עמודות),

משבצת\_חוקית(משבצת(\_שורה,\_עמודה)).

משבצת\_למשושה(משבצת(\_שורה,\_טור)):-

גודל\_לוח(\_שורות, \_טורים),

\_שורה\_בעבודה הוא \_שורה,

\_שורות\_בעבודה הוא ((\_טורים+1)//2),

\_שורה\_בעבודה =< \_שורות\_בעבודה,

\_הוספה הוא (\_שורה\_בעבודה-1),

\_טור =< ((\_טורים //2)+1)+\_הוספה,

\_טור >= ((\_טורים //2)+1)-\_הוספה.

משבצת\_למשושה(משבצת(\_שורה,\_טור)):-

גודל\_לוח(\_שורות, \_טורים),

\_שורה\_בעבודה הוא \_שורה,

\_שורה\_בבדיקה הוא (\_שורות-\_שורה),

\_שורות\_בעבודה הוא ((\_טורים+1)//2),

\_שורה\_בעבודה >= \_שורות\_בעבודה,

\_שורה\_בבדיקה >= \_שורות\_בעבודה,

\_הוספה הוא (\_שורות\_בעבודה-1),

\_טור =< ((\_טורים //2)+1)+\_הוספה,

\_טור >= ((\_טורים //2)+1)-\_הוספה.

משבצת\_למשושה(משבצת(\_שורה,\_טור)):-

גודל\_לוח(\_שורות, \_טורים),

\_שורה\_בעבודה הוא ((\_שורות-\_שורה)+1),

\_שורות\_בעבודה הוא ((\_טורים+1)//2),

\_שורה\_בעבודה =< \_שורות\_בעבודה,

\_הוספה הוא (\_שורה\_בעבודה-1),

\_טור =< ((\_טורים //2)+1)+\_הוספה,

\_טור >= ((\_טורים //2)+1)-\_הוספה.

%% אם המהלך לא אפשרי אזי יש הודעה.

%% ===============================

מהלך\_אפשרי(\_מצב, \_מהלך):-

לא מהלך\_חוקי(\_מצב, \_מהלך),

הודעה(' המהלך לא אפשרי, בחר משבצת חדשה'),

הכשל,!.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% מהלך של שחקן מחשב.

%% ===================

קבל\_מהלך\_מממשק\_שחקן(\_מצב, \_מהלך):-

אחזר\_תור\_מי(\_מצב, \_שחקן),

סוג\_שחקן( \_שחקן , מחשב(\_בינה)), %%<<<<=================================== שחקן ממוחשב!!!

מהלך\_לפי\_בינה(\_מצב, \_בינה, \_מהלך),

כתוב('המחשב בחר במהלך הבא: \_מהלך = '),

כתוב(\_מהלך),

שורה,

1 = 1.

%%===============================================================================

%%================================== ב י נ ה ==============================

%%===============================================================================

מהלך\_לפי\_בינה(\_מצב, 4, \_מהלך):-

מינימקס(\_מצב, 4, \_מהלך, \_ציון),

!.

מהלך\_לפי\_בינה(\_מצב, \_בינה, \_מהלך):-

\_בינה \= 4,

מצא\_הכל(\_משבצת , מהלך\_חוקי(\_מצב, מהלך(\_משבצת)), \_מהלכים), %% לשים לב שיהיה משתנה אחר ב-'מצא\_הכל/3'

ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_מהלכים, \_מהלכים\_עם\_ציון),

בחר\_בטוב(\_מהלכים\_עם\_ציון, \_מהלך, \_ציון),

שורה,

כתוב('הציון הוא: '),

כתוב(\_ציון), שורה.

%% ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_מהלכים, \_מהלכים\_עם\_ציון)

%% ===========================================================

ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, [], []).

ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, [\_מהלך|\_שאר\_מהלכים], [מהלך\_עם\_ציון(\_מהלך, \_ציון)|\_שאר\_עם\_ציון]):-

חשב\_ציון\_למהלך(\_מצב, \_בינה, \_מהלך, \_ציון),

ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_שאר\_מהלכים, \_שאר\_עם\_ציון).

%% בחר\_בטוב(\_מהלכים\_עם\_ציון, \_מהלך, \_ציון)

%%=========================================

%% חוק עצירה - אם יש מהלך אחד אז הוא המהלך הנבחר.

%% ==============================================

בחר\_בטוב([מהלך\_עם\_ציון(\_מהלך, \_ציון)], \_מהלך, \_ציון).

%% הרקורסיה - אם יש רשימת מהלכים, אז בוחרים את המהלך הטוב ביותר מהזנב של רשימת המהלכים ומשווים אותו למהלך הראשון. הכי טוב הוא הטוב מבין השניים.

%% ============================================================================================================================================

בחר\_בטוב([מהלך\_עם\_ציון(\_מהלך1, \_ציון1)|\_שאר], \_מהלך\_טוב, \_ציון\_טוב):-

בחר\_בטוב(\_שאר, \_מהלך2, \_ציון2),

טוב\_מבין\_שניים(\_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך\_טוב,\_ציון\_טוב).

%% בחירה של מהלך טוב ביותר בין שני מהלכים. החלטה שלי (!!!) המהלך הטוב יותר הוא זה שיש לו את הציון הגבוה יותר.

%% ============================================================================================================

%% טוב\_מבין\_שניים(\_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך\_טוב,\_ציון\_טוב).

%% =================================================================

טוב\_מבין\_שניים(\_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך1,\_ציון1):- \_ציון1 > \_ציון2.

טוב\_מבין\_שניים(\_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך2,\_ציון2):- \_ציון1 =< \_ציון2.

%%===============================================================================

%% בינה = 0 >>>> מהלך אקראי.

%% ==========================

%% חשב\_ציון\_למהלך(\_מצב, \_בינה, \_מהלך, \_ציון)

%% =========================================

חשב\_ציון\_למהלך(\_מצב, 0, \_מהלך, \_ציון):-

בצע\_מהלך(\_מצב, \_מהלך, \_), %% בשביל ליצור רשימת מושפעים ( אחרת ישאר H ).

אקראי(\_ציון).

חשב\_ציון\_למהלך(מצב(\_תור,\_לוח), \_בינה, \_מהלך, \_ציון):-

\_בינה > 0,

אקראי(\_מספר),

בצע\_מהלך(מצב(\_תור,\_לוח), \_מהלך, \_מצב\_חדש),

ציון\_מצב\_לשחקן\_לפי\_בינה(\_בינה, \_תור, \_מהלך, \_מצב\_חדש, \_ציון\_זמני),

\_ציון הוא \_ציון\_זמני + \_מספר.

%%===============================================================================

%%===============================================================================

%%===============================================================================

%% ציון\_מצב\_לשחקן\_לפי\_בינה(\_בינה, \_שחקן, \_מהלך, \_מצב\_חדש, \_ציון\_זמני)

%% ==================================================================

ציון\_מצב\_לשחקן\_לפי\_בינה(1, \_, \_, \_מצב, 10000):-

בדוק\_סוף\_משחק(\_מצב, \_),!.

%% במידה והמהלך שהתבצע עכשיו (הצעד האחרון) איננו הצעד המנצח, אז הציון הזמני שלו 0 ובהמשך מתווסף לו ערך אקראי.

%%============================================================================================================

ציון\_מצב\_לשחקן\_לפי\_בינה(1, \_, \_, \_, 0).

%%===============================================================================

%% מהלך של ניצחון

%% --------------

ציון\_מצב\_לשחקן\_לפי\_בינה(2, \_ , \_, \_מצב, 10000):-

בדוק\_סוף\_משחק(\_מצב, \_),!.

%% מהלך של תפיסת מקום

%% ------------------

ציון\_מצב\_לשחקן\_לפי\_בינה(2, \_שחקן , \_מהלך,\_מצב, \_ציון):-

\_שחקן = שחור,

מונה\_מקומות(\_מצב,\_מונה\_לבן,\_מונה\_שחור),

\_ציון = (\_מונה\_שחור\*\_מונה\_שחור) - (\_מונה\_לבן\*\_מונה\_לבן).

ציון\_מצב\_לשחקן\_לפי\_בינה(2, \_שחקן , \_מהלך,\_מצב, \_ציון):-

\_שחקן = לבן,

מונה\_מקומות(\_מצב,\_מונה\_לבן,\_מונה\_שחור),

\_ציון = (\_מונה\_לבן\*\_מונה\_לבן) - (\_מונה\_שחור\*\_מונה\_שחור).

מונה\_מקומות(\_מצב,\_מונה\_לבן,\_מונה\_שחור):-

הוסף(משבצות\_לבן(0)),

הוסף(משבצות\_שחור(0)),

גודל\_לוח(\_שורות, \_עמודות),

בין\_מספרים(1 , \_שורה, \_שורות),

בין\_מספרים(1 , \_עמודה, \_עמודות),

אחזר\_שורה\_ממשבצת( \_משבצת, \_שורה ),

אחזר\_טור\_ממשבצת( \_משבצת, \_עמודה ),

מקום\_לצבע(שחור,\_מצב,\_משבצת,\_מונה\_שחור\_זמני\_ראשון),

מקום\_לצבע(לבן,\_מצב,\_משבצת,\_מונה\_לבן\_זמני\_ראשון),

צבע\_מייצר\_לא\_חוקי\_לנגדי(שחור,\_מצב,\_משבצת,\_מונה\_שחור\_זמני\_שני),

צבע\_מייצר\_לא\_חוקי\_לנגדי(לבן,\_מצב,\_משבצת,\_מונה\_לבן\_זמני\_שני),

משבצות\_לבן(\_מונה\_לבן\_לולאה),

\_מונה\_לבן\_חדש הוא \_מונה\_לבן\_לולאה + \_מונה\_לבן\_זמני\_ראשון + \_מונה\_לבן\_זמני\_שני,

הסר\_כל(משבצות\_לבן(\_)),

הוסף(משבצות\_לבן(\_מונה\_לבן\_חדש)),

משבצות\_שחור(\_מונה\_שחור\_לולאה),

\_מונה\_שחור\_חדש הוא \_מונה\_שחור\_לולאה + \_מונה\_שחור\_זמני\_ראשון + \_מונה\_שחור\_זמני\_שני,

הסר\_כל(משבצות\_שחור(\_)),

הוסף(משבצות\_שחור(\_מונה\_שחור\_חדש)),

\_עמודה = \_עמודות,

\_שורה = \_שורות,!,

משבצות\_לבן(\_מונה\_לבן),

הסר\_כל(משבצות\_לבן(\_)),

משבצות\_שחור(\_מונה\_שחור),

הסר\_כל(משבצות\_שחור(\_)),

1 = 1.

מקום\_לצבע(\_צבע,\_מצב,\_משבצת,1):-

\_מצב = מצב(\_,\_לוח),

חבר( כלי(\_צבע , \_משבצת), \_לוח ),

!.

מקום\_לצבע(\_צבע,\_מצב,\_משבצת,0).

צבע\_מייצר\_לא\_חוקי\_לנגדי(\_צבע,\_מצב,\_משבצת,1):-

משבצת\_חוקית(\_משבצת),

\_מצב = מצב(\_,\_לוח),

משבצת\_שכנה\_בכיוון(\_משבצת\_בבדיקה, \_משבצת, \_),

חבר(כלי(\_צבע , \_משבצת\_בבדיקה), \_לוח ),

!.

צבע\_מייצר\_לא\_חוקי\_לנגדי(\_צבע,\_מצב,\_משבצת,0).

%%===============================================================================

%% מהלך של ניצחון

%% --------------

ציון\_מצב\_לשחקן\_לפי\_בינה(3, \_ , \_, \_מצב, 10000):-

בדוק\_סוף\_משחק(\_מצב, \_),!.

%% המהלך הכי פחות טוב ליריב הוא הכי טוב לי

%% -------------------------------------------------

ציון\_מצב\_לשחקן\_לפי\_בינה(3, \_שחקן, \_מהלך, \_מצב, \_ציון):-

מצא\_הכל(\_מהלך\_חוקי,מהלך\_חוקי(\_מצב,\_מהלך\_חוקי),\_מהלכים\_אפשריים),

מספר\_איברים(\_מספר,\_מהלכים\_אפשריים),

\_ציון הוא ( \_מספר \* -1),!.

**5-Excecute\_Move:**

%% בצע מהלך

%% --------

%%

%% כותב : עידן משיח.

%% -----------------

%% בצע\_מהלך(\_מצב, \_מהלך, \_מצב\_חדש)

%% ===============================

בצע\_מהלך( \_מצב, יציאה, \_מצב):-

הוסף(נלחץ\_יציאה). %% לזכרון מתווספת העובדה: 'נלחץ\_יציאה'

%% ===================================================================================

בצע\_מהלך( \_מצב , המלצה , \_מצב\_חדש):-

מצב\_נוכחי(\_מצב),

הוכחה(מהלך\_לפי\_בינה(\_מצב, 3 , \_מהלך), \_הסבר),!,

כתוב(\_מהלך),

שורה,

הצג\_מהלך\_מומלץ(\_מהלך),!,

הודעה('זה המהלך המומלץ'),

הסבר\_אם\_צריך(\_הסבר),

בצע\_מהלך\_אם\_צריך(\_מצב, \_מהלך, \_מצב\_חדש),

!.

הסבר\_אם\_צריך(\_הסבר):-

הודעה('?האם תרצה לקבל את הסבר',4,[כן,לא],כן),!,

פתח\_חלון\_בשם(600, 50 , 1600, 850, 240, 102, 'הסבר'),

נקה,

הצג\_הוכחה(\_הסבר,\_),

%%%=======================

כתוב\_בחלון('רשום כל דבר בשביל לסיים'),

קרא\_בחלון(\_עוד),

!,

%%% =======================

סגור\_חלון\_בשם('הסבר'),

חלון\_פעיל\_בשם('Flood Fill').

הסבר\_אם\_צריך(\_).

בצע\_מהלך\_אם\_צריך( \_מצב, \_מהלך, \_מצב\_חדש):-

\_מצב = מצב(\_צבע,\_לוח),

הודעה('?האם תרצה לקבל את ההמלצה',4,[כן,לא],כן),!,

בצע\_מהלך(\_מצב,\_מהלך,\_מצב\_חדש),

הסר\_מהלך\_מומלץ(\_מהלך),

1=1.

בצע\_מהלך\_אם\_צריך( \_מצב, \_מהלך, \_מצב):-

הסר\_מהלך\_מומלץ(\_מהלך).

הצג\_מהלך\_מומלץ(\_משבצת):-

תיאור\_רכיב(\_משבצת, \_רכיב),

שם\_רכיב( \_רכיב, 'images\\ADV.bmp').

הסר\_מהלך\_מומלץ(\_משבצת):-

תיאור\_רכיב(\_משבצת, \_רכיב),

שם\_רכיב( \_רכיב, 'images\\oval\_empty.bmp').

%% ===================================================================================

בצע\_מהלך( \_, חזור , \_מצב):-

מצב\_קודם(\_מצב).

שמירת\_מצב\_קודם(\_מצב):-

הסר\_כל(מצב\_קודם(\_)),

הוסף(מצב\_קודם(\_מצב)).

%% ===================================================================================

בצע\_מהלך( \_מצב, משחק\_חדש, \_מצב\_חדש ):-

מצב\_שמור(\_מצב\_חדש),

!.

%% ===================================================================================

בצע\_מהלך( \_מצב,הוראות, \_מצב ):-

פתח\_חלון\_בשם(1000, 200 , 1600, 400, 240, 102, 'הוראות'),

נקה,

תמונה('Images\\orar.bmp' , 0, 0, 600, 200 , \_),

קרא\_בחלון(\_עוד),

!,

סגור\_חלון\_בשם('הוראות'),

חלון\_פעיל\_בשם('Flood Fill'),

!.

%% ===================================================================================

בצע\_מהלך( \_מצב,שמירה, \_מצב ):-

פתח\_כ('GameSave.pro'),

גודל\_לוח(\_שורות, \_טורים),

כתוב('גודל\_לוח('),

כתוב(\_שורות),

כתוב(', '),

כתוב(\_טורים),

כתוב(').'),

שורה, שורה, שורה,

סוג\_שחקן(שחור, \_סוג\_שחקן\_1),

כתוב('סוג\_שחקן(שחור, '),

כתוב(\_סוג\_שחקן\_1),

כתוב(').'),

שורה, שורה, שורה,

סוג\_שחקן(לבן, \_סוג\_שחקן\_2),

כתוב('סוג\_שחקן(לבן, '),

כתוב(\_סוג\_שחקן\_2),

כתוב(').'),

שורה, שורה, שורה,

%% שמירת המצב.

כתוב('מצב\_שמור('),

כתוב(\_מצב),

כתוב(').'),

סגור\_כ,

הודעה('השמירה התבצעה בהצלחה'), !.

%% ===================================================================================

בצע\_מהלך( \_, טען\_משחק, \_מצב ):-

הסרה\_קודמת,

פתח\_ק('GameSave.pro'),

קרא(גודל\_לוח(\_שורות, \_טורים)),

הוסף(גודל\_לוח(\_שורות, \_טורים)),

קרא(סוג\_שחקן(שחור, \_סוג\_שחקן\_1)),

הוסף(סוג\_שחקן(שחור, \_סוג\_שחקן\_1)),

קרא(סוג\_שחקן(לבן, \_סוג\_שחקן\_2)),

הוסף(סוג\_שחקן(לבן, \_סוג\_שחקן\_2)),

קרא(מצב\_שמור(\_מצב)),

סגור\_ק('GameSave.pro'),

עדכן\_מצב(\_מצב),

הצג\_מצב(\_מצב),

הודעה('.הטעינה הסתיימה בהצלחה'),

!.

%% ===================================================================================

%% בצע\_מהלך(\_מצב, \_מהלך, \_מצב\_חדש)

%% ===============================

%% בצע\_מהלך(\_מצב, %% המצב שהיה קודם

%% \_מהלך, %% \_מהלך = מהלך( \_סוג, \_ממשבצת, \_למשבצת)

%% \_מצב\_חדש ):- %% המצב החדש שנוצר בעקבות המהלך, כולל החלפת תור.

%%===============================================================================

%% שחקן\_נגדי( \_צבע, \_צבע\_נגדי)

%%==============================

שחקן\_נגדי(לבן , שחור).

שחקן\_נגדי(שחור, לבן).

%% ==============================================================================

בצע\_מהלך( מצב(\_צבע, \_לוח),

מהלך(\_משבצת),

מצב(\_צבע\_נגדי, [ כלי(\_צבע, \_משבצת) | \_לוח])):-

שחקן\_נגדי( \_צבע, \_צבע\_נגדי),!.

בצע\_מהלך( מצב(\_צבע, \_לוח),

\_משבצת,

מצב(\_צבע\_נגדי, [ כלי(\_צבע, \_משבצת) | \_לוח])):-

שחקן\_נגדי( \_צבע, \_צבע\_נגדי),!.

**6-Graphics:**

%% גרפיקה

%% ------

%%

%% כותב : עידן משיח

%% ----------------

%% הדפס\_מצב(\_מצב). - חוק זה מדפיס את הלוח.

%% ==========================================

הדפס\_מצב(מצב(\_תור,\_לוח)):-

הדפס\_לוח(\_לוח),

הדפס\_תור\_מי(\_תור).

%% החוק הבא מדפיס הודעה לגבי תור מי לשחק כרגע.

%% ===========================================

הדפס\_תור\_מי(\_תור):-

שורה,

שורה,

כתוב('כעת תורו של השחקן ה'),

כתוב(\_תור),

שורה.

%% החוק הבא מדפיס באופן טקסטואלי את לוח המשחק עם כל הכלים הנמצאים עליו.

%% ====================================================================

הדפס\_לוח(\_לוח):-

גודל\_לוח(\_שורות,\_טורים),

שורה,שורה,

כתוב(' '),

בין\_מספרים(1,\_טור1,\_טורים),

כתוב(' '),כתוב(\_טור1),כתוב\_רווח\_אם\_צריך(\_טור1),

\_טור1 = \_טורים,

שורה,

כתוב(' '),

בין\_מספרים(1,\_טור2,\_טורים),

כתוב(' ---'),

\_טור2 = \_טורים,

בין\_מספרים(1,\_שורה,\_שורות),

שורה,

כתוב(' '),כתוב(\_שורה),כתוב\_רווח\_אם\_צריך(\_שורה),

בין\_מספרים(1,\_טור,\_טורים),

הדפס\_משבצת( משבצת(\_שורה,\_טור), \_לוח),

\_טור = \_טורים,

כתוב('|'),שורה,

כתוב(' '),

בין\_מספרים(1,\_טור3,\_טורים),

כתוב(' ---'),

\_טור3 = \_טורים,

\_שורה = \_שורות,

שורה.

%% החוק הבא דואג לכך שכותרת הלוח והמספרים בצד יהיו עם רווח נוסף במידה וזה מספר חד ספרתי.

%% ======================================================================================

כתוב\_רווח\_אם\_צריך(\_מספר):-

\_מספר < 10,!,

כתוב(' ').

%% חוק הבא אינו עושה דבר, התוכנית מגיעה לחוק זה אם \_מספר גדול מ- 10 או אם \_מספר איננו מספר.

%% =========================================================================================

כתוב\_רווח\_אם\_צריך(\_מספר).

%% אם יש כלי על המשבצת שעומדים להדפיס אז יצויין צבע הכלי, אחרת עוברים לחוק הבא.

%% =============================================================================

הדפס\_משבצת(\_משבצת,\_לוח):-

כתוב('|'),

משבצת\_חוקית(\_משבצת),

חבר(כלי(\_צבע,\_משבצת),\_לוח), !,

ציור\_של\_צבע(\_צבע, \_ציור),

כתוב(' '),כתוב(\_ציור),כתוב(' ').

%% לשורה זו מגיעים אם המשבצת ריקה.

%% ===============================

הדפס\_משבצת(\_משבצת,\_לוח):-

משבצת\_חוקית(\_משבצת),

ציור\_של\_צבע(ריק, \_ציור),

כתוב(' '),כתוב(\_ציור),כתוב(' ').

הדפס\_משבצת(\_משבצת,\_לוח):-

לא (משבצת\_חוקית(\_משבצת)),

כתוב('XXX').

ציור\_של\_צבע(שחור , 'ש').

ציור\_של\_צבע(לבן , 'ל').

ציור\_של\_צבע(ריק , ' ').

%%===============================================================================

%%===============================================================================

%%===============================================================================

הצג\_מצב( \_מצב ):-

אחזר\_תור\_מי(\_מצב, \_שחקן),

אחזר\_לוח(\_מצב, \_לוח),

הצג\_לוח(\_לוח),

הצג\_תור\_מי(\_שחקן).

הצג\_תור\_מי(\_צבע):-

קובץ\_תור( \_צבע, \_קובץ ),

%% תמונה( \_קובץ, \_איקס, \_וואי , \_אורך , \_רוחב , \_רכיב),

תמונה( \_קובץ , 630, 40, 120 , 65 , \_רכיב ),

אחזר\_תיאור\_מתיאור\_רכיב( \_תיאור\_רכיב , תור),

אחזר\_רכיב\_מתיאור\_רכיב( \_תיאור\_רכיב , \_רכיב),

הוסף(\_תיאור\_רכיב),

1 = 1 .

קובץ\_תור( לבן , 'Images\\TW.BMP' ).

קובץ\_תור( שחור, 'Images\\TB.BMP' ).

%% החוק הבא מציג באופן גרפי את לוח המשחק עם כל הכלים הנמצאים עליו.

%% ================================================================

הצג\_לוח(\_לוח):-

תמונה('Images\\backround.bmp' , 0, 0, 1050, 1050 , \_),

גודל\_לוח(\_שורות, \_עמודות),

בין\_מספרים(1 , \_שורה, \_שורות),

בין\_מספרים(1 , \_עמודה, \_עמודות),

אחזר\_שורה\_ממשבצת( \_משבצת, \_שורה ),

אחזר\_טור\_ממשבצת( \_משבצת, \_עמודה ),

הצג\_משבצת( \_משבצת , \_לוח),

\_עמודה = \_עמודות,

\_שורה = \_שורות,

הוסף\_תמונות,

1 = 1.

%% החוק הבא מוסיף בלולאה, תמונות מיוחדות ללוח. תמונות מיוחדות הן התמונות שאינן הלוח עצמו.

%% =======================================================================================

הוסף\_תמונות:-

נתוני\_תמונה(\_קובץ, \_תיאור, \_איקס, \_וואי, \_אורך, \_רוחב),

תמונה(\_קובץ, \_איקס, \_וואי, \_אורך, \_רוחב, \_רכיב),

אחזר\_תיאור\_מתיאור\_רכיב( \_תיאור\_רכיב, \_תיאור),

אחזר\_רכיב\_מתיאור\_רכיב( \_תיאור\_רכיב , \_רכיב),

הוסף(\_תיאור\_רכיב),

הכשל.

הוסף\_תמונות.

%% נתוני\_תמונה(\_קובץ, \_תיאור, \_מיקום\_איקס, \_מיקום\_וואי, \_אורך, \_רוחב)

%% ==================================================================

נתוני\_תמונה('Images\\ora.bmp' , הוראות , 630, 160, 120, 60).

נתוני\_תמונה('Images\\Advice.bmp' , המלצה , 630, 220, 120, 60).

נתוני\_תמונה('Images\\back.bmp' , חזור , 630, 280, 120, 60).

נתוני\_תמונה('Images\\NewGame.bmp' , משחק\_חדש , 630, 400, 120, 60).

נתוני\_תמונה('Images\\save.bmp' , שמירה , 630, 460, 120, 60).

נתוני\_תמונה('Images\\reload.bmp' , טען\_משחק , 630, 520, 120, 60).

נתוני\_תמונה('Images\\exit.bmp' , יציאה , 630, 580, 120, 60).

%% גודל\_תמונה(\_אורך, \_רוחב) - כמה פיקסלים התמונה תתפוס.

%% ========================

גודל\_תמונה(70 , 70).

%% החוק הבא מקבל מספר ומוצא את התמונה המתאימה למספר זה. בדוגמא זו המספרים הם בין 1 - 10.

%% ======================================================================================

/\*

מספר | כתובת התמונה

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

1 | Images\pic1.bmp

2 | Images\pic2.bmp

3 | Images\pic3.bmp

4 | Images\pic3.bmp

.. | ...

.. | ...

10 | Images\pic10.bmp

\*/

תמונה\_למספר(\_מספר, \_קובץ):-

הפוך\_למחרוזת(\_מספר, \_ערך\_כמחרוזת),

שרשר\_מחרוזות(['Images\\', 'pic', \_ערך\_כמחרוזת, '.bmp'], \_קובץ).

%% החוק בהא מציג את התמונה המתאימה לכל משבצת בהתאם למה שנמצא במארז 'מצב/2'

%% ========================================================================

הצג\_משבצת(\_משבצת , \_לוח):-

מיקום\_תמונה\_למשבצת(\_משבצת, \_איקס, \_וואי, \_אורך, \_רוחב),

תמונה\_מתאימה(\_משבצת, \_לוח, \_קובץ),

תמונה(\_קובץ, \_איקס, \_וואי, \_אורך, \_רוחב, \_רכיב),

אחזר\_תיאור\_מתיאור\_רכיב( \_תיאור\_רכיב, \_משבצת),

אחזר\_רכיב\_מתיאור\_רכיב( \_תיאור\_רכיב , \_רכיב),

הוסף( \_תיאור\_רכיב ),

1 = 1 .

%% אם המשבצת איננה משבצת חוקית אז לא מציגים אותה ולא עושים כלום.

%%==============================================================

הצג\_משבצת(\_משבצת , \_לוח):-

לא משבצת\_חוקית(\_משבצת).

%% החוק מקבל קואורדינטות טקסטואליות של משבצת ומתרגם אותם לקואורדינטות גרפיות המתאימות למסך הגרפי.

%% ================================================================================================

מיקום\_תמונה\_למשבצת(\_משבצת, \_איקס, \_וואי, \_אורך, \_רוחב):-

אחזר\_שורה\_ממשבצת( \_משבצת, \_שורה ),

אחזר\_טור\_ממשבצת( \_משבצת, \_עמודה ),

גודל\_תמונה(\_אורך, \_רוחב),

\_עמודה\_חדשה הוא \_עמודה,

\_שורה\_חדשה הוא \_שורה,

\_איקס הוא \_אורך \* \_עמודה\_חדשה\*3//4, %% בששביל לצופף את המשבצות החוקיות ולהעלים את מי שלא חוקי

\_וואי הוא \_רוחב \* \_שורה\_חדשה//2 +0. %% לפי שרטוט וחישוב והאפס בשביל להציג שאפשר להוסיף

%% אם הכלי נמצא על הלוח אז יש לו תמונה.

%% ====================================

תמונה\_מתאימה(\_משבצת , \_כלים, \_קובץ):-

אחזר\_משבצת\_מכלי(\_כלי, \_משבצת),

אחזר\_צבע\_של\_כלי(\_כלי, \_צבע),

אחזר\_כלי\_מלוח(\_כלים, \_כלי),

תמונה\_לכלי(\_צבע, \_קובץ),!.

%% אם הכלי לא נמצא על הלוח , אבל המשבצת חוקית, אז תופיע משבצת ריקה.

%% ================

%% =================================================================

תמונה\_מתאימה(\_משבצת , \_כלים, 'Images\\FillBlack.bmp'):- %% צבע לבן מציב

אחזר\_משבצת\_מכלי(\_כלי, \_משבצת),

לא אחזר\_כלי\_מהלוח(\_כלים, \_כלי),

משבצת\_חוקית(\_משבצת),

אחזר\_לוח(\_מצב\_שחור, \_כלים),

אחזר\_תור\_מי(\_מצב\_שחור, שחור),

לא מהלך\_חוקי(\_מצב\_שחור, מהלך(\_משבצת)),

אחזר\_לוח(\_מצב\_לבן, \_כלים),

אחזר\_תור\_מי(\_מצב\_לבן, לבן),

מהלך\_חוקי(\_מצב\_לבן, מהלך(\_משבצת)).

תמונה\_מתאימה(\_משבצת , \_כלים, 'Images\\FillWhite.bmp'):- %% צבע שחור מציב

אחזר\_משבצת\_מכלי(\_כלי, \_משבצת),

לא אחזר\_כלי\_מהלוח(\_כלים, \_כלי),

משבצת\_חוקית(\_משבצת),

אחזר\_לוח(\_מצב\_שחור, \_כלים),

אחזר\_תור\_מי(\_מצב\_שחור, שחור),

מהלך\_חוקי(\_מצב\_שחור, מהלך(\_משבצת)),

אחזר\_לוח(\_מצב\_לבן, \_כלים),

אחזר\_תור\_מי(\_מצב\_לבן, לבן),

לא מהלך\_חוקי(\_מצב\_לבן, מהלך(\_משבצת)).

תמונה\_מתאימה(\_משבצת , \_כלים, 'Images\\FillBoth.bmp'):- %% אף צבע לא יכול להציב

אחזר\_משבצת\_מכלי(\_כלי, \_משבצת),

לא אחזר\_כלי\_מהלוח(\_כלים, \_כלי),

משבצת\_חוקית(\_משבצת),

אחזר\_לוח(\_מצב\_שחור, \_כלים),

אחזר\_תור\_מי(\_מצב\_שחור, שחור),

לא מהלך\_חוקי(\_מצב\_שחור, מהלך(\_משבצת)),

אחזר\_לוח(\_מצב\_לבן, \_כלים),

אחזר\_תור\_מי(\_מצב\_לבן, לבן),

לא מהלך\_חוקי(\_מצב\_לבן, מהלך(\_משבצת)).

תמונה\_מתאימה( \_משבצת , \_כלים , 'Images\\oval\_empty.bmp' ):- %% שני הצבעים יכולים להציב

אחזר\_משבצת\_מכלי(\_כלי, \_משבצת),

לא אחזר\_כלי\_מהלוח(\_כלים, \_כלי),

משבצת\_חוקית(\_משבצת),

אחזר\_לוח(\_מצב\_שחור, \_כלים),

אחזר\_תור\_מי(\_מצב\_שחור, שחור),

מהלך\_חוקי(\_מצב\_שחור, מהלך(\_משבצת)),

אחזר\_לוח(\_מצב\_לבן, \_כלים),

אחזר\_תור\_מי(\_מצב\_לבן, לבן),

מהלך\_חוקי(\_מצב\_לבן, מהלך(\_משבצת)).

%% לכל צבע יש קובץ שמתאים לו.

%% ==========================

תמונה\_לכלי(שחור, 'Images\\Black.bmp').

תמונה\_לכלי(לבן , 'Images\\white.bmp').

%%===============================================================================

%%===============================================================================

%%================================ עדכן\_גרפי( \_מצב ) ============================

%%===============================================================================

%%===============================================================================

%% עדכן\_גרפי( \_מצב ) - מקבל את המצב חדש

%% =======================================

%% עדכן\_גרפי( \_מצב\_חדש)

%%=======================

עדכן\_גרפי( \_מצב):-

אחזר\_תור\_מי(\_מצב, \_תור),

אחזר\_לוח(\_מצב, \_לוח),

עדכן\_תור(\_תור),

עדכן\_לוח(\_לוח).

%% מעדכן רק את המשבצות ולא את כותרות הטורים והשורות, רק המשבצות הן פוטנציאל לעדכון.

%%=================================================================================

עדכן\_לוח(\_לוח):-

גודל\_לוח(\_שורות, \_עמודות),

בין\_מספרים(1 , \_שורה, \_שורות),

בין\_מספרים(1 , \_עמודה, \_עמודות),

אחזר\_שורה\_ממשבצת(\_משבצת, \_שורה),

אחזר\_טור\_ממשבצת(\_משבצת, \_עמודה),

עדכן\_משבצת( \_משבצת, \_לוח),

\_עמודה = \_עמודות,

\_שורה = \_שורות.

עדכן\_לוח(\_).

%% מוצאים את התמונה שצריכה להיות במקום זה על הלוח ושמים אותה שם.

%%==============================================================

עדכן\_משבצת( \_משבצת, \_לוח):-

תמונה\_מתאימה(\_משבצת, \_לוח, \_קובץ),

אחזר\_תיאור\_מתיאור\_רכיב( \_תיאור\_רכיב, \_משבצת),

אחזר\_תיאור\_רכיב(\_תיאור\_רכיב),

אחזר\_רכיב\_מתיאור\_רכיב( \_תיאור\_רכיב , \_רכיב),

שם\_רכיב(\_רכיב,\_קובץ).

עדכן\_תור(\_שחקן):-

אחזר\_תיאור\_מתיאור\_רכיב( \_תיאור\_רכיב, תור),

אחזר\_תיאור\_רכיב(\_תיאור\_רכיב),

אחזר\_רכיב\_מתיאור\_רכיב( \_תיאור\_רכיב , \_רכיב),

קובץ\_תור(\_שחקן,\_קובץ),

שם\_רכיב(\_רכיב,\_קובץ).

**7-End\_Game:**

%% --------------------------------------------------------------------------------------

%% סוף משחק

%% --------

%%

%% בודק האם הגענו למצב של סיום משחק ומסיים תוך סגירת המשחק.

%%

%% --------------------------------------------------------------------------------------

%% כותב : עידן משיח.

%% --------------------------------------------------------------------------------------

%% המשחק מסתיים באחד משני תנאים:

%% 1. אם נלחץ כפתור היציאה

%%========================

בדוק\_סוף\_משחק(\_,\_):-

נלחץ\_יציאה.

%% 2. המשחק נגמר כאשר לשחקן שזהו תורו, אין מהלך חוקי שניתן לעשותו.

%% בדוק\_סוף\_משחק(\_מצב, \_מנצח).

%%=============================

בדוק\_סוף\_משחק(מצב(\_צבע, \_לוח ) , \_מנצח):-

לא מהלך\_חוקי(מצב(\_צבע, \_לוח), \_ ), %% אין מהלכים חוקיים לזה שסיים את הסיבוב הנוכחי

שחקן\_נגדי(\_צבע,\_מנצח).

%%================================================================================

%% כאשר נלחץ כפתור היציאה, לא צריך להחליף תור!!

%% =============================================

סגור\_משחק(\_):-

נלחץ\_יציאה,

מצב\_נוכחי(\_מצב),

אחזר\_לוח(\_מצב, \_לוח),

הדפס\_לוח(\_לוח),

הודעה\_מתאימה(\_),

סגור\_חלון\_בשם(\_).

%% כאשר לא נלחץ כפתור היציאה.

%% ==========================

סגור\_משחק(\_מנצח):-

לא נלחץ\_יציאה,

מצב\_נוכחי(\_מצב),

אחזר\_לוח(\_מצב, \_לוח),

הדפס\_לוח(\_לוח),

אחזר\_תור\_מי(\_מצב, \_שחקן),

שחקן\_נגדי(\_שחקן, \_מנצח),

הצג\_תור\_מי(\_מנצח),

הודעה\_מתאימה( \_מנצח ),

סגור\_חלון\_בשם(\_).

%% הודעה לגבי יציאה מהמשחק.

%% ========================

הודעה\_מתאימה(\_):-

נלחץ\_יציאה,!,

הודעה('החלטת לצאת מהמשחק, נתראה בפעם הבא ').

%% הודעה אודות המנצח.

%% ===================

הודעה\_מתאימה(\_מנצח):-

שרשר\_מחרוזות(['המשחק נגמר, המנצח: ', \_מנצח], \_הודעה),

הודעה(\_הודעה),!.

**8-DynamicFunction:**

%%=====================================================

%%================ הפונקציה הדינמית ==================

%%=====================================================

%% תופסת לחיצות העכבר ומתרגמת אותן לפעולות שונות.

%% ==============================================

פונקציה\_דינמית( \_חלון, \_רכיב, לחיצה\_בודדת):-

אחזר\_רכיב\_מתיאור\_רכיב( \_תיאור\_רכיב , \_רכיב),

אחזר\_תיאור\_רכיב(\_תיאור\_רכיב),

אחזר\_תיאור\_מתיאור\_רכיב( \_תיאור\_רכיב, \_תיאור), %% כפי שהתווסף לזכרון המחשב בשלב בניית הלוח.

פעולה\_מתאימה(\_תיאור).

%% =====================================

פעולה\_מתאימה(\_מהלך):-

הוסף(נקלט\_מהלך(\_מהלך)).

**9-MinMax:**

% מינימקס %

%% מהלכים\_אפשריים(\_מצב\_נוכחי ,\_רשימת\_מהלכים\_אפשריים).

%%===================================================

מהלכים\_אפשריים(\_מצב,\_מהלכים):-

מצא\_הכל(\_מהלך\_חוקי,מהלך\_חוקי(\_מצב,\_מהלך\_חוקי),\_מהלכים).

%% אחזר\_שחקן(\_מצב\_נוכחי,\_שחקן\_נוכחי).

%%===================================

אחזר\_שחקן(\_מצב,\_שחקן):-

אחזר\_תור\_מי(\_מצב,\_שחקן).

%% ציון\_מצב\_לשחקן\_לפי\_בינה(\_בינה\_לפיה\_מחשבים\_ציון, \_שחקן\_נוכחי, \_מצב\_נוכחי, \_ציון\_למצב).

%% =====================================================================================

ציון\_מצב\_לפי\_בינה(4, \_, \_מצב, \_ציון):-

ציון\_למצב(\_מצב,\_ציון).

%% מקדם\_ציון\_לשחקן(\_שחקן, \_מקדם).

%%===============================

מקדם\_ציון\_לשחקן(שחור , 1). %% שחור - שחקן "MAX"

מקדם\_ציון\_לשחקן(לבן , -1). %% לבן - שחקן "MIN"

%% החוק הבא הנו דוגמא!! כל אחד יחליט מהו ה-"ציון למצב" המתאים למשחק שלו רשום בקובץ: EXTRA\_RULES

%%==============================================================================================

%%

%% ציון למצב ( ללא תלות בשחקן!!! )

%%================================

ציון\_למצב(\_מצב,\_ציון):-

\_מצב = מצב(\_שחקן,\_לוח),

מקדם\_ציון\_לשחקן(\_שחקן,\_מקדם),

מצא\_הכל(\_מהלך\_חוקי,מהלך\_חוקי(\_מצב,\_מהלך\_חוקי),\_מהלכים\_אפשריים),

מספר\_איברים(\_מספר,\_מהלכים\_אפשריים),

\_ציון הוא \_מספר \* \_מקדם.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% אלגוריתם מינימקס %

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

מעקב.

%%=============================================================

%%=============================================================

עומק\_לחיפוש(3).

מינימקס(\_מצב, \_בינה, \_מהלך, \_ציון):-

עומק\_לחיפוש(\_עומק),

מהלכים\_אפשריים(\_מצב ,\_מהלכים),

בחר\_מהלך\_טוב(\_בינה , \_עומק, \_מצב, \_מהלכים, \_מהלך, \_ציון).

%%-------------------------------------------

בחר\_מהלך\_טוב(\_בינה,\_עומק, \_מצב, \_מהלכים\_אפשריים, \_מהלך\_בסט, \_ציון\_בסט):-

הוסף( מינימקס( \_עומק, \_מהלכים\_אפשריים) ),

הוסף( מהלך\_נבחר(\_עומק, \_, \_) ),

חזור,

לולאת\_מינימקס(\_עומק, \_מצב, \_בינה ),

מינימקס(\_עומק, []),!,

הסר\_כל( מינימקס( \_עומק, \_) ),

מהלך\_נבחר( \_עומק, \_מהלך\_בסט, \_ציון\_בסט ),

הסר\_כל(מהלך\_נבחר( \_עומק, \_, \_ )),

מעקב\_צומת(\_עומק, \_ציון\_בסט),

1=1.

לולאת\_מינימקס(\_עומק,\_מצב,\_בינה):-

אחזר\_שחקן(\_מצב,\_שחקן),

הסרה\_והוספה\_של\_מהלך\_נבחר\_ומהלכים\_אפשריים(\_עומק,\_מהלך,\_מהלך\_טוב,\_ציון\_טוב),

בצע\_מהלך(\_מצב, \_מהלך, \_מצב\_חדש),

ציון\_בצעדים( \_בינה, \_עומק, \_מצב\_חדש, \_ציון),

טוב\_מבין\_שניים\_לשחקן(\_שחקן, \_מהלך,\_ציון,\_מהלך\_טוב,\_ציון\_טוב,\_מהלך\_נבחר,\_ציון\_נבחר),

הוסף(מהלך\_נבחר(\_עומק,\_מהלך\_נבחר,\_ציון\_נבחר)),!.

הסרה\_והוספה\_של\_מהלך\_נבחר\_ומהלכים\_אפשריים(\_עומק,\_מהלך\_נוכחי,\_מהלך\_טוב,\_ציון\_טוב):-

מינימקס(\_עומק,[\_מהלך\_נוכחי|\_מהלכים\_יתר]),

הסר\_כל(מינימקס(\_עומק,\_)),

הוסף(מינימקס(\_עומק,\_מהלכים\_יתר)),

מהלך\_נבחר(\_עומק,\_מהלך\_טוב,\_ציון\_טוב),

הסר\_כל(מהלך\_נבחר(\_עומק,\_,\_)),!.

%%-------------------------------------------

%% הכניסה לעומק.

%%==============

%% במקרה שהגענו לסוף משחק (עלה ממשי בעץ המינימקס) יש להחזיר את הציון של המצב הנוכחי ע"פ הבינה.

%%=============================================================================================

ציון\_בצעדים( \_בינה, \_צעדים, \_מצב, \_ציון):-

\_צעדים > 1,

בדוק\_סוף\_משחק(\_מצב, \_ ),

אחזר\_שחקן(\_מצב,\_שחקן),

ציון\_מצב\_לפי\_בינה(\_בינה, \_שחקן, \_מצב, \_ציון),

מעקב\_עלה(\_צעדים,\_ציון).

%% במקרה שהגענו לעומק 1 (עלה מדומה בעץ המינימקס) יש להחזיר את הציון של המצב הנוכחי ע"פ הבינה.

%%===========================================================================================

ציון\_בצעדים( \_בינה, 1, \_מצב, \_ציון):-

אחזר\_שחקן(\_מצב,\_שחקן),

ציון\_מצב\_לפי\_בינה(\_בינה, \_שחקן, \_מצב, \_ציון),

מעקב\_עלה(\_ציון).

%% במקרה שלא הגענו לסוף משחק או עומק 1 יש למצוא את כל המהלכים האפשריים ולהחזיר את המהלך והציון הטובים ביותר

%% תוך כדי שימוש באלגוריתם מינימקס לאחר קידום בצעד אחד.

%%==========================================================================

ציון\_בצעדים( \_בינה, \_צעדים, \_מצב, \_ציון):-

\_צעדים > 1,

לא בדוק\_סוף\_משחק(\_מצב, \_ ),

\_צעדים\_פחות הוא \_צעדים - 1,

מהלכים\_אפשריים(\_מצב ,\_מהלכים),

בחר\_מהלך\_טוב( \_בינה, \_צעדים\_פחות, \_מצב, \_מהלכים, \_ ,\_ציון).

%%-------------------------------------------

%% טוב\_מבין\_שניים\_לשחקן(\_שחקן, \_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך\_טוב\_לשחקן,\_ציון\_טוב\_לשחקן).

%% =================================================================

טוב\_מבין\_שניים\_לשחקן(\_, \_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך1,\_ציון1):-

משתנה(\_מהלך2),!.

טוב\_מבין\_שניים\_לשחקן(\_, \_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך2,\_ציון2):-

משתנה(\_מהלך1),!.

טוב\_מבין\_שניים\_לשחקן(\_שחקן, \_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך1,\_ציון1):-

מקדם\_ציון\_לשחקן(\_שחקן, \_מקדם),

\_ציון\_מתוקן1 הוא \_ציון1 \* \_מקדם,

\_ציון\_מתוקן2 הוא \_ציון2 \* \_מקדם,

\_ציון\_מתוקן1 > \_ציון\_מתוקן2.

טוב\_מבין\_שניים\_לשחקן(\_שחקן, \_מהלך1,\_ציון1,\_מהלך2,\_ציון2,\_מהלך2,\_ציון2):-

מקדם\_ציון\_לשחקן(\_שחקן, \_מקדם),

\_ציון\_מתוקן1 הוא \_ציון1 \* \_מקדם,

\_ציון\_מתוקן2 הוא \_ציון2 \* \_מקדם,

\_ציון\_מתוקן1 =< \_ציון\_מתוקן2.

%%========================================================================================================

%%========================================================================================================

%% חוקי מעקב

%%==========

מעקב\_עלה(\_עומק,\_ציון):- לא מעקב.

מעקב\_עלה(\_ציון):- לא מעקב.

מעקב\_צומת(\_עומק,\_ציון):- לא מעקב.

מעקב\_עלה(\_עומק,\_ציון):-

מעקב,

\_רווחים הוא \_עומק \* 8,

רווח(\_רווחים),כתוב('#ציון: '),כתוב(\_ציון),שורה.

מעקב\_עלה(\_ציון):-

מעקב,

כתוב('-ציון: '),כתוב(\_ציון),שורה.

מעקב\_צומת(\_עומק,\_ציון):-

מעקב,

\_רווחים הוא \_עומק \* 8,

רווח(\_רווחים),כתוב('ציון נבחר: '),כתוב(\_ציון),שורה.

**10-exp:**

%%%==========================================================================

%%%==========================================================================

%%%============================ "הוכחה/2" של אייל ואלה לב =============

%%%==========================================================================

%%%==========================================================================

% הוכחה(\_מטרה,\_הסבר).

% ====================

%

% טיפול בפסוק מסוג ידוע, כגון: שלילה, ביטוי לוגי, ביטוי אריתמטי, מתארי רשימות. בקטגוריה זו יכללו רקורסיות וחוקים

%================================================================================================================

% שאין עניין להתעמק בהסבר שלהם. כל אחד יוסיף כאן חוקים בהתאם לתוכנית שלו.

%========================================================================

הוכחה(\_מטרה, הסבר(\_מטרה, \_מסוג, \_הסבר)):-

פסוק\_מסוג(\_מטרה, \_, \_),

!, % על מנת למנוע ניסיון לבדוק בדרך אחרת.

%%כתוב\_שלי(א(\_מטרה)),

\_מטרה, % מפעילים את ה-"\_מטרה" ומודעים שהיא מתקיימת.

%%כתוב\_שלי(ב),

פסוק\_מסוג(\_מטרה, \_מסוג, \_הסבר),

%%כתוב\_שלי(א),

1=1.

% טיפול בחוק רגיל. פירוק של חוק לראש וגוף באמצעות "פסוק/2". החוק "המשך\_הוכחה/2" מטפל בהוכחת הגוף.

%=================================================================================================

הוכחה(\_מטרה, הסבר(\_מטרה, \_גוף, \_הסבר\_גוף)):-

פסוק(\_מטרה, \_גוף),

המשך\_הוכחה(\_גוף, \_הסבר\_גוף).

%%=================================================================================

%%======================= ישנם 5 מיקרים ש- 'המשך\_הוכחה/2' מטפל בהם: ===============

%%=================================================================================

% טיפול בעובדה.

% ==============

המשך\_הוכחה(true, []):-!.

% טיפול במטרה בודדת, חסומה. למשל:---->>> א:-ב. 'ב' היא מטרה חסומה. במקרה כזה רשימת ההסבר ריקה.

% =============================================================================================

המשך\_הוכחה(\_מטרה\_בודדת, []):-

מטרה\_חסומה(\_מטרה\_בודדת),

!,

\_מטרה\_בודדת.

% טיפול במטרה בודדת, שאיננה חסומה. אבל שיש לה פסוג מסוג

% ================================================================================================================

%%% המשך\_הוכחה(\_מטרה\_בודדת, [הסבר(\_מטרה\_בודדת, \_פסוק\_מסוג, [\_הסבר\_מטרה])]):-

%%% פסוק\_מסוג(\_מטרה\_בודדת, \_פסוק\_מסוג, \_הסבר\_מטרה),

%%% !,

%%% \_מטרה\_בודדת.

%%%

% טיפול במטרה בודדת, שאיננה חסומה. למשל:---->>> א:-ב. במקרה כזה יש רשימת הסבר של המטרה. הפעלה רקורסיבית של הוכחה.

% ================================================================================================================

המשך\_הוכחה(\_מטרה\_בודדת, [\_הסבר\_מטרה]):-

הוכחה(\_מטרה\_בודדת, \_הסבר\_מטרה).

% טיפול בגוף חוק מורכב, כלומר התקבלה "רשימה עם סוגריים עגולות". למשל: ----->>>> א:-ב, ג, ד.

% \_מטרה\_בודדת הנה המטרה הראשונה (והיא חסומה) ואילו '\_שאר' זהו שאר הגוף. ----->>>> הגוף הוא (ב, ג, ד)

% ====================================================================================================

המשך\_הוכחה((\_מטרה\_בודדת, \_שאר),\_הסבר\_שאר):-

מטרה\_חסומה(\_מטרה\_בודדת), % כאשר "\_מטרה\_בודדת" הנה רכיב חסום.

!,

\_מטרה\_בודדת,

המשך\_הוכחה(\_שאר, \_הסבר\_שאר).

% טיפול בגוף חוק מורכב, כלומר התקבלה "רשימה עם סוגריים עגולות". למשל: ----->>>> א:-ב, ג, ד.

% \_מטרה\_בודדת הנה המטרה הראשונה, והיא לא חסומה, ואילו '\_שאר' זהו שאר הגוף. ----->>>> הגוף הוא (ב, ג, ד)

% ====================================================================================================

המשך\_הוכחה((\_מטרה\_בודדת, \_שאר), [\_הסבר\_מטרה|\_הסבר\_שאר]):-

הוכחה(\_מטרה\_בודדת, \_הסבר\_מטרה), % כאשר "\_מטרה\_בודדת" אינה חסומה.

המשך\_הוכחה(\_שאר, \_הסבר\_שאר).

% ====================================================================================================

% ============================================= פסוק\_מסוג/3 ========================================

% ====================================================================================================

% טיפול בשלילה.

% ==============

פסוק\_מסוג(לא(\_מטרה), שלילה, ['אי אפשר להסיק ש',\_מטרה]).

פסוק\_מסוג(לא (\_מטרה), שלילה, ['אי אפשר להסיק ש',\_מטרה]).

פסוק\_מסוג(לא \_מטרה, שלילה, ['אי אפשר להסיק ש',\_מטרה]).

% טיפול בביטוי חשבוני.

% =====================

פסוק\_מסוג(\_תוצאה הוא \_חישוב, חשבוני, ['מחישוב מקבלים ש',\_תוצאה\_כערך\_תווי,' הוא ',\_חישוב]):-

המר\_מספר\_לערך\_תווי(\_תוצאה,\_תוצאה\_כערך\_תווי).

המר\_מספר\_לערך\_תווי(\_תוצאה,\_מחרוזת):-

שלם(\_תוצאה),

תרגם\_למחרוזת(\_תוצאה,\_מחרוזת).

המר\_מספר\_לערך\_תווי(\_תוצאה,\_תוצאה\_כערך\_תווי):-

ממשי(\_תוצאה),

קטום(\_תוצאה,\_תוצאה1),

הפך\_מקטום(\_תוצאה,\_תוצאה2),

ערך\_מוחלט(\_תוצאה2, \_זנב\_חיובי),

תרגם\_למחרוזת(\_תוצאה1,\_מחרוזת\_אחדות),

תרגם\_למחרוזת(\_זנב\_חיובי,\_מחרוזת\_עשרוני),

שרשר\_מחרוזות([\_מחרוזת\_עשרוני,'.',\_מחרוזת\_אחדות],\_תוצאה\_כערך\_תווי).

תרגם\_למחרוזת(0,'0').

תרגם\_למחרוזת(\_מספר,\_מחרוזת):-

תרגם\_מספר\_למחרוזת(\_מספר,\_אסקי),

מחרוזת(\_מחרוזת,\_אסקי).

תרגם\_מספר\_למחרוזת(0,[]).

תרגם\_מספר\_למחרוזת(\_מספר,\_מחרוזת):-

\_מספר < 0,

\_חיובי הוא \_מספר \* -1,

תרגם\_מספר\_למחרוזת(\_חיובי,\_המספר),

שרשר(\_המספר,[45],\_מחרוזת).

תרגם\_מספר\_למחרוזת(\_מספר,\_מחרוזת):-

\_מספר > 0,

\_לתרגם הוא \_מספר mod 10,

\_אסקי הוא \_לתרגם + 48,

\_להמשך הוא \_מספר divu 10,

תרגם\_מספר\_למחרוזת(\_להמשך,\_שאר\_אסקי),

שרשר(\_שאר\_אסקי,[\_אסקי],\_מחרוזת).

הפך\_מקטום(\_תוצאה,\_תוצאה2):-

עומק\_אחרי\_נקודה(\_כמה),

קטום(\_תוצאה,\_פחות),

\_תוצאה\_בלי\_אחדות הוא \_תוצאה - \_פחות,

פעל\_עבור\_הפך\_מקטום(\_תוצאה\_בלי\_אחדות,\_תוצאה2,\_כמה).

עומק\_אחרי\_נקודה(5).

פעל\_עבור\_הפך\_מקטום(\_תוצאה,\_תוצאה1, 0):-

קטום(\_תוצאה,\_תוצאה1).

% \_תוצאה1 > 0,

% 1.0 הוא \_תוצאה1 / \_תוצאה.

פעל\_עבור\_הפך\_מקטום(\_תוצאה,\_תוצאה1,\_כמה):-

\_כמה > 0,

\_כמה\_חדש הוא \_כמה - 1,

\_תוצאה\_הבא הוא \_תוצאה \* 10,

פעל\_עבור\_הפך\_מקטום(\_תוצאה\_הבא,\_תוצאה1,\_כמה\_חדש).

% ביטוי\_חשבוני(\_תוצאה := \_תרגיל):-\_תוצאה := \_תרגיל.

% טיפול בביטויים לוגיים.

% =======================

פסוק\_מסוג(\_א = \_ב, השוואה, [\_א,' שווה ל',\_ב]).

פסוק\_מסוג(\_א \= \_ב, השוואה, [\_א,' שונה מ',\_ב]).

פסוק\_מסוג(\_א =< \_ב, השוואה, [\_א,' קטון או שווה ל',\_ב]).

פסוק\_מסוג(\_א < \_ב, השוואה, [\_א,' קטן מ',\_ב]).

פסוק\_מסוג(\_א > \_ב, השוואה, [\_א,' גדול מ',\_ב]).

פסוק\_מסוג(\_א >= \_ב, השוואה, [\_א,' גדול או שווה',\_ב]).

פסוק\_מסוג(\_א =:= \_ב, השוואה, [\_א,' ו',\_ב,' בעלי אותו ערך']).

% ============== טיפול במתארי רשימה. חובה לרשום כאן את כל מתארי הרשימה גם מ-my\_list!!======

% טיפול במתארי רשימה.

% ===================

פסוק\_מסוג( חבר(\_איבר,\_רשימה), מתארי\_רשימה, ['האיבר ',\_איבר,' חבר ברשימה ',\_רשימה]).

פסוק\_מסוג( שרשר(\_א,\_ב,\_רשימה), מתארי\_רשימה, ['משרשור עולה ש',\_א,' ו',\_ב,' בונים את ',\_רשימה]).

פסוק\_מסוג( מיקום\_איבר(\_מ,\_א,\_רשימה), מתארי\_רשימה, ['האיבר ',\_א,' נמצא במקום מספר ',\_מ,' ברשימה ',\_רשימה]).

פסוק\_מסוג( ראשון\_ברשימה(\_א,\_רשימה), מתארי\_רשימה, ['האיבר ',\_א,' נמצא במקום ראשון ברשימה ',\_רשימה]).

פסוק\_מסוג( מספר\_איברים(\_מ,\_רשימה), מתארי\_רשימה, [\_מ ,' הוא מספר האיברים ברשימה ',\_רשימה]).

פסוק\_מסוג( סכום\_רשימה(\_ס,\_רשימה), מתארי\_רשימה, [\_ס ,' הוא סכום האיברים ברשימה ',\_רשימה]).

%פסוק\_מסוג( מצא\_כל(\_מ,\_תנאי,\_רשימה), מתארי\_רשימה, ['מהפעלת מצא\_כל, מתקבלת הרשימה ',\_רשימה]).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

פסוק\_מסוג( מצא\_הכל(\_,מהלך\_וציון(\_מצב, \_, \_),\_רשימה), רשימה\_של(מהלך\_וציון(\_מצב), \_רשימה), ['ישנם ', \_מספר, ' מהלכים']):-משתנה(\_רשימה),!.

פסוק\_מסוג( מצא\_הכל(\_,מהלך\_וציון(\_מצב, \_, \_),\_רשימה), רשימה\_של(מהלך\_וציון(\_מצב), \_רשימה), ['ישנם ', \_מספר, ' מהלכים']):-לא משתנה(\_רשימה), מספר\_איברים(\_מספר, \_רשימה),!.

פסוק\_מסוג( ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_מהלכים, \_מהלכים\_עם\_ציון), רשימה\_של(מהלך\_וציון(\_מצב, \_בינה), \_מהלכים\_עם\_ציון), ['ישנם ', \_מספר, ' מהלכים']):-משתנה(\_מהלכים\_עם\_ציון),!.

פסוק\_מסוג( ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_מהלכים, \_מהלכים\_עם\_ציון), רשימה\_של(מהלך\_וציון(\_מצב, \_בינה), \_מהלכים\_עם\_ציון), ['ישנם ', \_מספר, ' מהלכים']):-לא משתנה(\_מהלכים\_עם\_ציון),מספר\_איברים(\_מספר, \_מהלכים\_עם\_ציון),!.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

פסוק\_מסוג( מצא\_הכל(\_מ,\_תנאי,\_רשימה), מתארי\_רשימה, ['מהפעלת מצא\_הכל, מתקבלת הרשימה ',\_רשימה]).

פסוק\_מסוג( מספר\_איברים\_משותפים(\_א,\_ב,\_מספר), מתארי\_רשימה, ['בשתי הרשימות ',\_א,' ו',\_ב,' ישנם ',\_מספר,' איברים משותפים']).

פסוק\_מסוג( עלה(\_צומת), עצים, ['הצומת ',\_צומת,' היא עלה']).

פסוק\_מסוג( שורש(\_שורש), עצים, ['הצומת ',\_שורש,' היא שורש של העץ']).

פסוק\_מסוג( קוד(\_צומת,\_מחלה), לפי\_פרויקט, ['הקוד למחלה ',\_מחלה,' הוא ',\_צומת]).

פסוק\_מסוג( מחלה(\_מ), לפי\_פרויקט, ['מחלה ',\_מ]).

פסוק\_מסוג(קשת\_לפי\_שם(\_שם,\_צומת1, \_צומת2), עצים, [' קיימת קשת בין: [',\_צומת1,'] ל[',\_צומת2,']']).

%פסוק\_מסוג( רשימת\_סימפטומים\_של\_מחלה(\_מ,\_ר), לפי\_פרויקט, ['סימפטומים של ',\_מ,':',\_ר]).

%פסוק\_מסוג(רשימת\_סימפטומים\_לצומת(\_צומת,\_רשימה),לפי\_פרויקט,['רשימת סימפטומים לצומת ',\_צומת,':',\_רשימה]).

% כאשר פסוק אינו מצליח לפרק חוק לראש וגוף אזי החוק הנו פעולת מערכת. לדוגמא "קרא", "כתוב", "פתח\_חלון" וכו' (גם !).

%================================================================================================================

%פסוק\_מסוג(מהלך\_לפי\_בינה(\_, \_בינה, \_מהלך), מהלך, ['מהפעלת בינה ברמה ', \_בינה, ' מתקבל המהלך ', \_מהלך]).

% כאשר פסוק אינו מצליח לפרק חוק לראש וגוף אזי החוק הנו פעולת מערכת. לדוגמא "קרא", "כתוב", "פתח\_חלון" וכו' (גם !).

%================================================================================================================

פסוק\_מסוג(\_מטרה, מערכת, [\_מטרה,' היא פעולת מערכת']):-

\_מטרה \= (\_,\_),

לא (פסוק(\_מטרה,\_)).

% טיפול בחוקים חסומים, חוקים בהם אין אנו רוצים להכנס להסבר בכלל!!!

% =================================================================

% כל אחד יחסום חוקים בהתאם לפרויקט שלו. מה שנראה לו מתאים!!

% =========================================================

% מטרה\_חסומה(\_מטרה).

מטרה\_חסומה(כתוב\_בחלון).

מטרה\_חסומה(שורה\_בחלון).

מטרה\_חסומה(!).

%%מטרה\_חסומה(אחזר\_לוח(\_, \_)).

%%=== ועוד...

%%=========================================

% אין טיפול בכישלון. אי אפשר להוכיח במערכת זו, ביטוי שנכשל!!!

% =============================================================

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%

%%

%% הצגת רשימת ההסבר שהתקבלה

%%

%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%הצג\_הוכחה(הסבר(\_מטרה, \_גוף\_או\_סוג, \_הסבר), \_תשובה).

הצג\_הוכחה(הסבר(\_עובדה, true, []), \_תשובה):-

כתוב\_יפה(\_עובדה),

כתוב\_בחלון(' נתון כעובדה'),

שורה\_בחלון,

כתוב\_בחלון('חזור/סוף'),

שורה\_בחלון,

קרא\_בחלון(\_תשובה),

!.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

הצג\_הוכחה(הסבר(\_פסוק\_מסוג, \_סוג, \_הסבר), \_תשובה\_מוחזרת):-

פסוק\_מסוג(\_פסוק\_מסוג, \_סוג, \_הסבר),

\_סוג = רשימה\_של(\_של, \_רשימה),

חזור,!,

כתוב\_רשימה(\_הסבר),

שורה\_בחלון,

פרט\_רשימה(\_של, \_רשימה,1 ),

כתוב\_בחלון('מספר/חזור/סוף'),

שורה\_בחלון,

קרא\_בחלון(\_תשובה),

כנס\_לתוך\_רשימה\_מורכבת(\_של, \_תשובה, \_רשימה, \_תשובה\_מוחזרת),!.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

הצג\_הוכחה(הסבר(\_פסוק\_מסוג, \_סוג, \_הסבר), \_תשובה):-

פסוק\_מסוג(\_פסוק\_מסוג, \_סוג, \_הסבר),

כתוב\_רשימה(\_הסבר),

שורה\_בחלון,

כתוב\_בחלון('חזור/סוף'),

שורה\_בחלון,

קרא\_בחלון(\_תשובה),

!.

הצג\_הוכחה(הסבר(\_מטרה, \_גוף, \_הסבר), \_תשובה\_מוחזרת):-

חזור,

כתוב\_יפה(\_מטרה),

כתוב\_בחלון(' כי:'),

שורה\_בחלון,

כתוב\_תת\_מטרות(\_הסבר,1),

כתוב\_בחלון('מספר/חזור/סוף'),

שורה\_בחלון,

קרא\_בחלון(\_תשובה),

המשך\_הצג\_הוכחה(\_תשובה, \_הסבר, \_תשובה\_מוחזרת),!.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

פרט\_רשימה(\_של, [], \_):-שורה\_בחלון.

פרט\_רשימה(\_של, [\_איבר|\_שאר], \_מספר):-

כתוב\_בחלון(\_מספר),

כתוב\_בחלון(') '),

פרט\_איבר(\_של, \_איבר),

שורה\_בחלון,

\_מספר\_פלוס הוא \_מספר + 1,

פרט\_רשימה(\_של, \_שאר,\_מספר\_פלוס).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

כתוב\_תת\_מטרות([], \_):-שורה\_בחלון.

כתוב\_תת\_מטרות([הסבר(\_תת\_מטרה,\_,\_)|\_שאר], \_מספר):-

כתוב\_בחלון(\_מספר),

כתוב\_בחלון(') '),

כתוב\_יפה(\_תת\_מטרה),

שורה\_בחלון,

\_מספר\_פלוס הוא \_מספר + 1,

כתוב\_תת\_מטרות(\_שאר,\_מספר\_פלוס).

כתוב\_רשימה([]).

כתוב\_רשימה([\_מילה|\_שאר]):-

משתנה(\_מילה),!,

כתוב\_בחלון(' \_חופשי\_ '),

כתוב\_רשימה(\_שאר).

כתוב\_רשימה([\_מילה|\_שאר]):-

המר\_מספר\_לערך\_תווי(\_מילה, \_טקסט),!,

כתוב\_בחלון(\_טקסט),

כתוב\_רשימה(\_שאר).

%כתוב\_רשימה([\_מילה|\_שאר]):-

% מילול(\_מילה, \_מילול),!,

% כתוב\_רשימה(\_מילול),

% כתוב\_רשימה(\_שאר).

כתוב\_רשימה([\_מילה|\_שאר]):-

כתוב\_בחלון(\_מילה),

כתוב\_רשימה(\_שאר).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

כנס\_לתוך\_רשימה\_מורכבת(\_, ח,\_,ח).

כנס\_לתוך\_רשימה\_מורכבת(\_, ס,\_,ס).

כנס\_לתוך\_רשימה\_מורכבת(\_של, \_מספר,\_רשימה,\_תשובה):-

מיקום\_איבר(\_מספר, \_איבר, \_רשימה),

הוכחה\_לאיבר\_מרשימה\_מורכבת(\_של, \_איבר, \_תת\_הסבר),

הצג\_הוכחה(\_תת\_הסבר, \_תשובה),

\_תשובה = 'ס'.

הוכחה\_לאיבר\_מרשימה\_מורכבת(מהלך\_וציון(\_מצב), ציון\_למהלך(\_מהלך, \_), \_הסבר):-הוכחה(מהלך\_וציון(\_מצב, \_מהלך, \_ציון), \_הסבר).

הוכחה\_לאיבר\_מרשימה\_מורכבת(מהלך\_וציון(\_מצב, \_בינה), מהלך\_עם\_ציון(\_מהלך, \_), \_הסבר):-הוכחה(חשב\_ציון\_למהלך(\_מצב, \_בינה, \_מהלך, \_ציון), \_הסבר).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

המשך\_הצג\_הוכחה(ח,\_,ח).

המשך\_הצג\_הוכחה(ס,\_,ס).

המשך\_הצג\_הוכחה(\_מספר,\_הסבר,\_תשובה):-

מיקום\_איבר(\_מספר, \_תת\_הסבר, \_הסבר),

הצג\_הוכחה(\_תת\_הסבר, \_תשובה),

\_תשובה = 'ס'.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

פרט\_איבר(מהלך\_וציון(\_מצב), ציון\_למהלך(\_מהלך, \_ציון)):-כתוב\_יפה(['הציון למהלך ', \_מהלך,' הוא ', \_ציון]).

פרט\_איבר(מהלך\_וציון(\_מצב, \_בינה), מהלך\_עם\_ציון(\_מהלך, \_ציון)):-כתוב\_יפה(['הציון למהלך ', \_מהלך,' הוא ', \_ציון]).

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

כתוב\_יפה(\_מטרה):-

מילול(\_מטרה, \_מילולי),!,

כתוב\_רשימה(\_מילולי).

%%================================================================

%% מילול(\_מטרה, \_ניסוח מילולי).

%% כל אחד יתמלל כאן את העבודה שלו!!!

% =================================

מילול(\_רשימה,\_רשימה):-\_רשימה=[\_|\_],!.

מילול(ציון\_הפרש(\_,\_שחקן,\_כלים\_שלי), [' מספר הכלים של השחקן על הלוח הוא', \_כלים\_שלי]).

מילול(מרחק\_הכי\_קטן(\_,\_שחקן,\_מרחק), [' הסכום הכולל של כל הפיתגורסים של כל המשבצות של השחקן שעל הלוח ממשבצת המטרה שלו הוא:', \_מרחק]).

מילול(ציון\_מצב\_לשחקן\_לפי\_בינה(\_בינה, \_תור, \_מהלך, \_מצב\_חדש, \_ציון), ['לפי בינה ',\_בינה, 'ינתן הציון ', \_טקסט, 'למהלך זה']):-

המר\_מספר\_לערך\_תווי(\_ציון, \_טקסט).

מילול(חשב\_ציון\_למהלך(\_, \_בינה, \_מהלך, \_ציון), ['הציון למהלך ', \_מהלך,' לפי בינה ', \_בינה, ' הוא ', \_טקסט]):-

המר\_מספר\_לערך\_תווי(\_ציון, \_טקסט).

מילול(מהלך\_לפי\_בינה(\_מצב, \_בינה, \_מהלך), [ 'לפי בינה ברמה - ', \_בינה, ' נבחר המהלך ', \_מהלך]).

מילול(אחזר\_לוח(\_מצב, \_לוח),['מסתכל על הלוח']).

מילול(אחזר\_תור\_מי(\_מצב, \_צבע),['כעת תורו של שחקן ', \_צבע]).

מילול(מהלכים\_חוקיים\_של\_צבע(\_מצב,\_לוח, \_צבע, \_מהלכים),['עבור השחקן ה', \_צבע, ' קיימים ', \_מספר, ' מהלכים.']):-

מספר\_איברים(\_מספר, \_מהלכים).

מילול(ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_רשימת\_מהלכים, \_מהלכים\_עם\_ציון),['לפי המצב ולפי בינה ',\_בינה, ' ניתנו הציונים ל-', \_מספר, ' מהלכים']):-

מספר\_איברים(\_מספר, \_רשימת\_מהלכים).

מילול(בחר\_בטוב(\_מהלכים\_עם\_ציון, מהלך(אכילה,\_מ,\_אל) , \_ציון ),['המהלך אכילה ממשבצת ', \_מ, ' למשבצת ', \_אל, '. קיבל ציון ',\_ציון,' שהוא גבוה יותר משאר המהלכים.']).

מילול(בחר\_בטוב(\_מהלכים\_עם\_ציון, מהלך(תזוזה,\_מ,\_למשבצת) , \_ציון ),['המהלך תזוזה ממשבצת ', \_מ, ' למשבצת ', \_למשבצת, '. קיבל ציון ',\_ציון,' שהוא גבוה יותר משאר המהלכים.']).

/\*

מילול(המהלך\_הטוב\_ביותר(\_מצב, מהלך(\_מתא, \_מספר,\_לתא), \_רשימת\_מהלכים),['הכי כדאי לך להזיז כלי מתא ', \_מתא, ' עם קוביה של ', \_מספר, ' צעדים, לתא ', \_לתא ]).

מילול(בחר\_בטוב(\_מהלכים\_עם\_ציון, מהלך(\_מתא, \_מספר,\_לתא), \_ציון ),['המהלך הזזה מתא ', \_מתא, ' עם קוביה של ', \_מספר, '. קיבל ציון ',\_ציון,' שהוא גבוה יותר משאר המהלכים.']).

מילול(ארוז\_מהלכים\_עם\_ציון(\_מצב, \_בינה, \_רשימת\_מהלכים, \_מהלכים\_עם\_ציון),['לפי המצב ולפי בינה ',\_בינה, ' ניתנו הציונים למהלכים']).

מילול(חשב\_ציון\_למהלך(\_מצב, \_בינה, מהלך(\_מתא, \_מספר,\_לתא), \_ציון),['לפי המצב ולפי בינה ',\_בינה, 'עבור המהלך: הזזה מתא ', \_מתא, ' עם קוביה של ', \_מספר, ' הציון הוא ', \_ציון]).

מילול(אחזר\_תור\_מי(\_מצב,\_תור),['כעת תורו של השחקן ה',\_תור]).

מילול(בצע\_מהלך(\_מצב,מהלך(\_מתא, \_מספר,\_לתא),\_מצב\_חדש),['המחשב מבצע את המהלך: הזזה מתא ', \_מתא, ' עם קוביה של ', \_מספר]).

מילול(ציון\_מצב\_לשחקן\_לפי\_בינה(\_בינה, \_תור, \_מצב\_חדש, \_ציון\_זמני),['הציון של המהלך לפי בינה ',\_בינה, ' הוא ',\_ציון\_זמני]).

מילול(ציון\_מספר\_פתוחים\_לשחקן(\_שחקן,\_מצב,\_ציון),['ציון "פתוחים" במצב הנתון עבור השחקן ה',\_שחקן, ' הוא ',\_ציון]).

מילול(ציון\_מספר\_כלים\_לשחקן(\_שחקן,\_מצב,\_ציון),['ציון "כלים" במצב הנתון עבור השחקן ה',\_שחקן, ' הוא ',\_ציון]).

מילול(ציון\_לאכילת\_כלים\_של\_יריב(\_שחקן,\_מצב,\_ציון),['ציון "אכילה" במצב הנתון עבור השחקן ה',\_שחקן, ' הוא ',\_ציון]).

מילול(שחקן\_נגדי(\_שחקן, \_יריב),['היריב של ה',\_שחקן, ' הוא ה',\_יריב]).

מילול(אחזר\_לוח(\_מצב, \_לוח),['מסתכל על הלוח']).

מילול(אחזר\_אכולים(\_מצב, \_לוח),['בודק אכולים']).

מילול(מספר\_איברים(\_מספר, \_רשימה),['מספר האיברים ברשימה הוא ',\_מספר]).

מילול(סכום\_רשימה(\_מספר, \_רשימה),['סכום הרשימה הוא ',\_מספר]).

מילול(הורדת\_הופעה\_ראשונה(\_, \_, \_),['המחשב הסיר את מספר הקוביה שמסעה שוחק']).

\*/

מילול(אקראי(\_מספר), ['מגריל מספר אקראי ', \_מספר]).

%%== אם אין מילול מיוחד, אזי ירשם המילול של פסוק מסוג!! לכן שני החוקים הבאים מופיעים בסוף. ======

%% ===============================================================================================

מילול(\_מטרה,\_הסבר):-פסוק\_מסוג(\_מטרה, \_, \_הסבר).

מילול(\_מטרה,[\_מטרה]). %%% <<<=============חייב להיות האחרון!!!

%% ===============================================================================================

%% =====================================================================================================================

%% ה-"הוכחה/2" תופעל או מן החוק "מערכת\_מומחית/0" או שכל אחד ישלב אותה במקום הנראה לו. אחרי ההוכחה תבוא ההצגה של רשימת ההסבר..

%% =====================================================================================================================

מע:-מערכת\_מומחית.

מערכת\_מומחית:-

פתח\_חלון\_בשם(50, 50 , 750, 600, 240, 102, 'הסבר'),

נקה,

כתוב\_בחלון('מה להוכיח?'),

שורה\_בחלון,

קרא\_בחלון(\_מטרה),

הוכחה(\_מטרה,\_הסבר),

הצג\_הוכחה(\_הסבר,\_),

%%%=======================

כתוב\_בחלון('האם ברצונך לראות עוד אפשרויות (רשום כן/לא) ? '),

קרא\_בחלון(\_עוד), % במקרה והתקבלה התשובה 'כן'

\_עוד \= 'כן', % הכשלה מכוונת לקבלת תשובות נוספות

!,

%%% ======================= % ביטול תשובות נוספות במקרה של התשובה 'לא'

סגור\_חלון\_בשם('הסבר').

מערכת\_מומחית:-

סגור\_חלון\_בשם('הסבר'),

הודעה('סיים').

%%%==========================================================================

%%%==========================================================================

%%%============================ "הוכחה/2" של אייל ואלה לב =============

%%%==========================================================================

%%%==========================================================================

מהלך\_וציון(מצב\_טוב, רוץ\_בעיגולים, 500).

מהלך\_וציון(מצב\_טוב, קפוץ\_במקום, 100).

מהלך\_וציון(מצב\_טוב, לשיר\_ולירקוד, 200).

מהלך\_וציון(מצב\_טוב, לשבת, 150):-1==1.

אאא:-מצא\_הכל(ציון\_למהלך(\_מהלך, \_ציון), מהלך\_וציון(מצב\_טוב, \_מהלך, \_ציון), \_רשימה).

בבב:-

פתח\_חלון\_בשם(50, 50 , 750, 600, 240, 102, 'הסבר'),

כתוב\_רשימה([-5, ' ',-2.543, ' ',1.5555]),

הודעה('סיים'),

סגור\_חלון\_בשם('הסבר').

**list\_Proj:**

% מתארי רשימות

% ------------

חבר(\_איבר,[\_איבר|\_]).

חבר(\_איבר,[\_|\_תת\_רשימה]):-

חבר(\_איבר,\_תת\_רשימה).

מיקום\_איבר(1,\_איבר,\_רשימה):-

ראשון\_ברשימה(\_איבר,\_רשימה).

מיקום\_איבר(\_מיקום,\_איבר,[\_|\_תת\_רשימה]):-

מיקום\_איבר(\_מיקום\_חדש,\_איבר,\_תת\_רשימה),

\_מיקום הוא 1 + \_מיקום\_חדש.

איבר\_אחרון\_ברשימה(\_איבר\_אחרון,[\_איבר\_אחרון]).

איבר\_אחרון\_ברשימה(\_איבר\_אחרון,[\_|\_שאר]):-

איבר\_אחרון\_ברשימה(\_איבר\_אחרון,\_שאר).

מספר\_איברים(0,[]).

מספר\_איברים(\_מספר,[\_|\_זנב]):-

מספר\_איברים(\_קודם,\_זנב),

\_מספר הוא 1 + \_קודם.

סכום\_רשימה(0,[]).

סכום\_רשימה(\_סכום,[\_ראש|\_זנב]):-

סכום\_רשימה(\_סכום\_זנב,\_זנב),

\_סכום הוא \_ראש + \_סכום\_זנב.

שרשר([],\_רשימה,\_רשימה).

שרשר([\_איבר|\_זנב],\_רשימה,[\_איבר|\_חדש]):-

שרשר(\_זנב,\_רשימה,\_חדש).

%% ----------------------------------------------

%% הפיכת רשימת רשימות לרשימה אחת >> שיטוח רשימה.

שרשר([],[]).

שרשר([[]|\_זנב],\_שטוחה):-

שרשר(\_זנב,\_שטוחה).

שרשר([[\_ראש|\_זנב]|\_רשימות],\_שטוחה):-

רשימה(\_ראש),

שרשר(\_ראש,\_ראש\_משוטח),

שרשר([\_זנב|\_רשימות],\_זנב\_שטוחה),

שרשר(\_ראש\_משוטח,\_שנב\_שטוחה,\_שטוחה).

שרשר([[\_ראש|\_זנב]|\_רשימות],[\_ראש|\_שטוחה]):-

לא רשימה(\_ראש),

שרשר([\_זנב|\_רשימות],\_שטוחה).

שרשר([\_ראש|\_רשימות],[\_ראש|\_שטוחה]):-

לא רשימה(\_ראש),

שרשר(\_רשימות,\_שטוחה).

רשימה([]).

רשימה([\_|\_]).

%% --------------------------------------------

הפוך( \_רשימה, \_הפוכה ) :-

הפוך( \_רשימה, [], \_הפוכה ), !.

הפוך([],\_צובר,\_צובר).

הפוך([\_איבר|\_זנב],\_צובר,\_הפוכה):-

הפוך(\_זנב,[\_איבר|\_צובר],\_הפוכה).

מצא\_כל(\_איבר,\_מטרה,\_רשימה):-

קורא(\_מטרה),

הוסף( מחסנית(\_איבר) ),

הכשל;

!,

הוסף( מחסנית(תחתית) ),

אסוף\_כל(\_רשימה).

אסוף\_כל(L):-

הסר( מחסנית(\_איבר) ),

!,

( \_איבר = תחתית,

!,

L = [] ;

\_איבר \= תחתית,

L = [\_איבר|\_שאר],

אסוף\_כל(\_שאר)

).

%%=========================================

ראשון\_ברשימה(\_איבר, [\_איבר|\_]).

מצא\_הכל(A,B,C):-

findall(A,B,C).

חבר\_אקראי(\_איבר, \_רשימה):-

מספר\_איברים(\_מספר, \_רשימה),

אקראי(\_אקראי),

\_זמני הוא \_אקראי \* \_מספר +1,

קטום(\_זמני, \_מיקום),

מיקום\_איבר(\_מיקום, \_איבר, \_רשימה).

%% בין\_מספרים(\_נמוך,\_מספר,\_גבוה). חוק זה מייצר מספרים עוקבים בין גבולות נתונים.

%%===============================

בין\_מספרים(\_נמוך,\_נמוך,\_גבוה):-\_נמוך=<\_גבוה.

בין\_מספרים(\_נמוך,\_מספר,\_גבוה):-\_נמוך <\_גבוה,

\_נמוך\_פלוס הוא \_נמוך + 1,

בין\_מספרים(\_נמוך\_פלוס,\_מספר,\_גבוה).

ערך\_מוחלט(\_ערך,\_ערך):- \_ערך >= 0 .

ערך\_מוחלט(\_ערך,\_תוצאה):- \_ערך< 0, \_תוצאה הוא \_ערך\*(-1).

%============================ 17. א. ==============================

%%% 17. א. הורדת\_הופעה\_ראשונה(\_איבר\_נתון,\_רשימה\_נתונה,\_רשימה\_חדשה).

%%% חוק מס' 1: נחוץ אם האיבר איננו חבר ברשימה הנתונה ואנחנו רוצים לקבל את הרשימה המקורית ולא 'כשלון'

הורדת\_הופעה\_ראשונה(\_איבר,[],[]).

%%% חוק מס' 2:

הורדת\_הופעה\_ראשונה(\_איבר,[\_איבר|\_רשימה],\_רשימה).

%%% חוק מס' 3:

הורדת\_הופעה\_ראשונה(\_איבר,[\_איבר1|\_רשימה],[\_איבר1|\_רשימה\_חדשה]):-

\_איבר \= \_איבר1,

הורדת\_הופעה\_ראשונה(\_איבר,\_רשימה,\_רשימה\_חדשה).

%%====================================================================

הוצא\_איבר(\_איבר,[\_איבר|\_זנב],\_זנב).

הוצא\_איבר(\_איבר,[\_ראשון|\_זנב],[\_ראשון|\_זנב\_חדש]):-

הוצא\_איבר(\_איבר,\_זנב,\_זנב\_חדש).

%%====================================================================

תרגום([],[]).

תרגום([\_איבר|\_רשימה],[\_ערך\_מלולי|\_רשימה1]):-

קוד(\_איבר,\_ערך\_מלולי),

תרגום(\_רשימה,\_רשימה1).

%%====================================================================

% תרגום\_לפי(\_קריטריון,\_רשימה,\_מתורגמת).

תרגום\_לפי(\_קריטריון,[],[]).

תרגום\_לפי(\_קריטריון,[\_איבר|\_זנב],[\_מתורגם|\_זנב\_מתורגם]):-

קוד\_לפי(\_קריטריון,\_איבר,\_מתורגם),

תרגום\_לפי(\_קריטריון,\_זנב,\_זנב\_מתורגם).

%%% קוד\_לפי(גובה,\_שם,\_גובה):-

%%% משתתף(\_שם,\_,\_גובה).

%%====================================================================

%% חיסור\_רשימות(\_הרשימה\_ממנה\_מחסרים, \_הרשימה\_הנתונה, \_התוצאה).

%% ============================================================

%% א. חוק עצירה. אם הרשימה ממנה מחסרים היא ריקה אז גם רשימת התוצאה היא ריקה.

חיסור\_רשימות([],\_,[]).

%% ב. אם האיבר העומד בראש הרשימה שממנה מחסרים נמצא ברשימה הנתונה

%% אז מורידים אותו וממשיכים בתהליך חיסור הרשימות עם שאר אברי הרשימה הנתונה.

%% סימן ה- ! (cut), משמש כאן בתפקיד else. אם נכשל בחוק שאחריו מופיע ה- ! הפרולוג עובר לחוק האלטרנטיבי.

חיסור\_רשימות([\_איבר|\_שאר\_רשימה],\_רשימה,\_רשימה\_חדשה):-

חבר(\_איבר,\_רשימה),!,

חיסור\_רשימות(\_שאר\_רשימה,\_רשימה,\_רשימה\_חדשה).

%% ג. אם האיבר העומד בראש הרשימה שממנה מחסרים לא נמצא ברשימה הנתונה

%% אז מעבירים אותו לרשימת התוצאה וממשיכים בתהליך חיסור הרשימות עם שאר אברי הרשימה הנתונה.

חיסור\_רשימות([\_איבר|\_שאר\_רשימה],\_רשימה,[\_איבר|\_זנב]):-

חיסור\_רשימות(\_שאר\_רשימה,\_רשימה,\_זנב).

%%=================================================================================================

%%% 18. הורד\_כל(\_איבר\_נתון,\_רשימה\_נתונה,\_רשימה\_ללא\_כל\_הופעות\_האיבר\_הנתון).

%%========================================================================

הורד\_כל(\_איבר,[],[]).

הורד\_כל(\_איבר,[\_איבר|\_רשימה],\_רשימה\_חדשה):-

הורד\_כל(\_איבר,\_רשימה,\_רשימה\_חדשה).

הורד\_כל(\_איבר,[\_איבר1|\_רשימה],[\_איבר1|\_רשימה\_חדשה]):-

\_איבר \= \_איבר1,

הורד\_כל(\_איבר,\_רשימה,\_רשימה\_חדשה).

%%=================================================================================================

%%% 19. ניפוי\_רשימה(\_רשימה\_נתונה,\_רשימה\_חדשה\_ללא\_כפילויות).

%%=========================================================

ניפוי\_רשימה([],[]).

ניפוי\_רשימה([\_איבר|\_רשימה],[\_איבר|\_רשימה\_חדשה]):-

הורד\_כל(\_איבר,\_רשימה,\_רשימה1),

ניפוי\_רשימה(\_רשימה1,\_רשימה\_חדשה).

%%=================================================================================================

%%% 37. שרשר(\_רשימה,\_רשימה\_חדשה). - מקבל רשימת רשימות ומשטח אותה לרשימה אחת.

%%===========================================================================

שרשר([\_רשימה],\_רשימה ).

שרשר([\_רשימה|\_שאר\_הרשימות],\_רשימה\_משורשרת):-

שרשר(\_שאר\_הרשימות,\_רשימת\_שאר\_הרשימות\_המשורשרת),

שרשר(\_רשימה,\_רשימת\_שאר\_הרשימות\_המשורשרת,\_רשימה\_משורשרת).

%%=================================================================================================

/\*

%% לולאת while - עם 2 תנאים

%% -----------

%% כל עוד אחד מ-2 התנאים לא מתקיים, הלולאה ממשיכה לפעול.

חזור\_כל\_עוד\_לא(\_דגל1,\_דגל2):-

חזור\_שלי(\_דגל1,\_,\_);

חזור\_שלי(\_דגל2,\_,\_).

חזור\_שלי(\_דגל,\_א,[\_א|\_שאר]):-לא \_דגל.

חזור\_שלי(\_דגל,\_א,[\_|\_שאר]):-לא \_דגל,חזור\_שלי(\_דגל,\_א,\_שאר).

\*/

%%===============================================================================

%%===============================================================================

%%===============================================================================

%%================================ טיפול במחרוזות ===============================

%%===============================================================================

%%===============================================================================

%%===============================================================================

%% אם המספר הוא חד ספרתי.

%% ======================

הפוך\_למחרוזת(\_מספר, \_מחרוזת):-

\_מספר < 10,

ספרה\_למחרוזת(\_מספר, \_מחרוזת).

%% אם המספר הוא בעל יותר מאשר ספרה אחת.

%% =====================================

הפוך\_למחרוזת(\_מספר, \_מחרוזת):-

\_מספר >= 10,

פרק\_מספר(\_מספר, \_אחדות, \_שאר),

ספרה\_למחרוזת(\_אחדות, \_מחרוזת\_אחדות),

הפוך\_למחרוזת(\_שאר, \_מחרוזת\_של\_שאר),

שרשר\_מחרוזות([\_מחרוזת\_של\_שאר,\_מחרוזת\_אחדות], \_מחרוזת).

%% פירוק של ספרת האחדות של המספר.

%% ===============================

פרק\_מספר(\_מספר, \_אחדות, \_שאר):-

\_אחדות הוא \_מספר mod 10,

\_שאר הוא \_מספר // 10.

%% ספרה\_למחרוזת(\_מספר\_רכיב, \_מספר\_כמחרוזת).

%% =========================================

ספרה\_למחרוזת(0, '0').

ספרה\_למחרוזת(1, '1').

ספרה\_למחרוזת(2, '2').

ספרה\_למחרוזת(3, '3').

ספרה\_למחרוזת(4, '4').

ספרה\_למחרוזת(5, '5').

ספרה\_למחרוזת(6, '6').

ספרה\_למחרוזת(7, '7').

ספרה\_למחרוזת(8, '8').

ספרה\_למחרוזת(9, '9').

%%==========================================================================

%% על מנת לאתר טעות/טעויות

%% =======================

כתוב\_שלי(\_מה):-

כתוב(\_מה),שורה.

כתוב\_שלי(\_מה):-

כתוב('חוזר '),

כתוב(\_מה),שורה,הכשל.

%%===============================================================================

חלק\_ושארית(\_מונה, \_מכנה, \_מנה, \_שארית):-

\_מנה הוא \_מונה // \_מכנה,

\_שארית הוא \_מונה mod \_מכנה.