



# WAR

**תיק פרויקט – מלחמה ( WAR )**

**מגיש: אופק הראל**

**תעודת זהות: 214206971**

**בית ספר: איש שלום**

**מקצוע: סייבר / אסמבלר**

**מנחה: אופיר שביט**

# תוכן עניינים:

2 ..... תיאור הפרויקט:

4 ..... הוראות הפעלה:

5 ..... תיאור הפתרון:

7 ..... תרשים זרימה:

8 ..... פרוצדורות:

8 ..... *MOUSE.ASM*

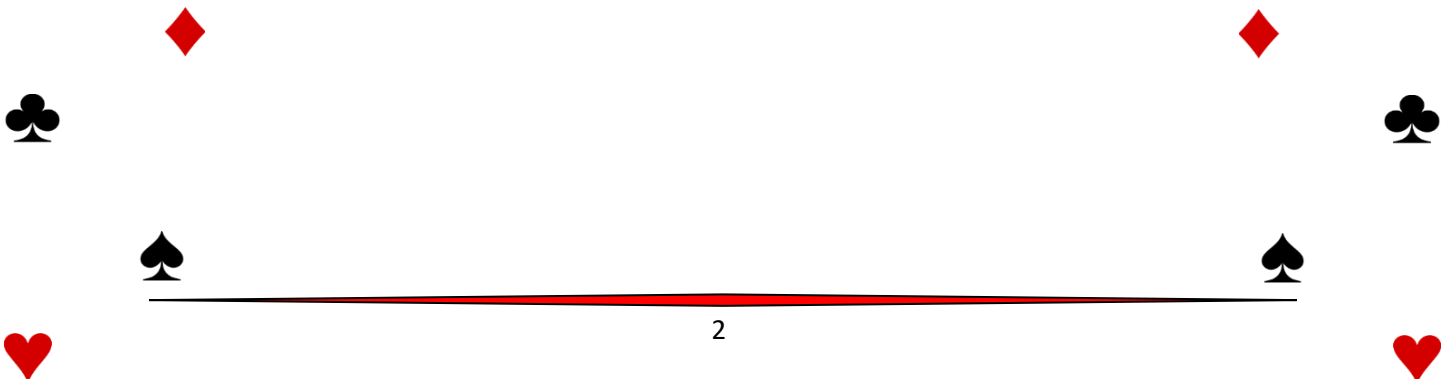
9 ..... *MENU.ASM*

9 ..... *GAME.ASM*

13 ..... *PRINT.ASM*

16 ..... דוגמת ריצה:

18 ..... סיכום אישי:



# תיאור הפרויקט:

מלחמה, אחד מהמשחקים הוותיקים ביותר כמשחק קלפים מודרני, ידוע גם בשם קרב, זה משחק המיועד לשתי אנשים המשוחק לרוב על ידי ילדים. למלחמה יש המון סוגים, אני בחרתי בגרסה שאותה אני מכיר מהיותי ילד, 52 קלפים.

**מטרת המשחק:** להשיג את כל הקלפים (52)

## חוקי משחק:

1. בתחילת המשחק מחולקים 52 הקלפים שווה בשווה באופן רנדומלי (מעורבב).
2. מהלך המשחק מתבצע על ידי כך שכל שחקן שם בתורו קלף, השחקן שיצא לו קלף בדירוג הכי גבוהה לוקח את הקלפים שהושמו.
3. במקרה ויש קלפים שווים אז זהו מצב הנקרא "מלחמה" שבו כל שחקן בתורו שולף 3 קלפים הפוכים (עם הגב כלפי מעלה) והקלף הרביעי רגיל (תוכן הקלף כלפי מעלה), מי שהקלף הרביעי שלו בעל הערך הכי גבוהה זוכה בכל הקלפים שהושמו.
4. דירוג הקלפים: (הכי נמוך) 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A (הכי גבוה).
5. יש ארבע סוגי קלפים עבור כל ערך: יהלום, עלה, לב, תלתן. (סוג זה לא משפיע כלל על הערך של הקלפים).

## הוראות משחק מחשב:

1. העכבר הוא הכלי משחק שלך (לחיצה שמאלית).
2. הקלפים שלך הם בצד שמאל של המסך בעוד כי קלפי היריב בצד הימני.
3. לביצוע תור יש ללחוץ על חפיסת הקלפים שלך (צד ימין למטה).
4. תור זהו סבב מלא של השמת קלף שלך ושל היריב, כלומר אתה משחק נגד המחשב.

# הוראות הפעלה:

<b>קובץ הרצה</b>	<b>קבצי BMP</b>						<b>סך כל הקבצים:</b>
<b>ASM</b>							
run	WinCrw BlankB Escbnp Escbip GameFd warGGG ofekRL homeSc hmPlay hmlInfo infoSc infEsc	JunHRC JunLBC JunCBC QueDRC QueHRC QueLBC QueCBC KinDRC KinHRC KinLBC KinCBC StackC CrownP	EigDRC EigHRC EigLBC EigCBC NinDRC NinHRC NinLBC NinCBC TenDRC TenHRC TenLBC TenCBC JunDRC	FouLBC FouCBC FivDRC FivHRC FivLBC FivCBC SixDRC SixHRC SixLBC SixCBC SevDRC SevHRC SevLBC SevCBC	AceDRC AceHRC AceLBC AceCBC TwoDRC TwoHRC TwoLBC TwoCBC ThrDRC ThrHRC ThrLBC ThrCBC FouDRC FouHRC		BMP: 66  ASM: 8  ALL: 74
<b>קבצי ASM</b>							
Screens Game Menu Print Mouse Const DataSeg							

## הפעלה:

ברגע ההפעלה, יפתח מסך טעינה, ברגע סיומו יפתח תפריט שבו ניתן לנווט על ידי העכבר ולחיצה שמאלית על הכפתור תכניס אותך בהתאם. בתפריט יש 3 אפשרויות ניווט:

1. Play: יכניס אותך למשחק, במשחק יש שני אפשרויות ניווט:

a. על החפיסה שלך (שמאלי למטה) תריץ תור מלא.

b. Esc: יוציא אותך חזרה למסך הראשי ותוכל לבצע את הניווט משם בהתאם.

(הערה: יציאה מהמשחק ב ESC שומרת את התקדמות המסך עד יציאה ממשית מהאמולטור. כלומר ניתן לחזור חזרה למקום שבו הפסקת ברגע שיצאת ממשחק המשחק הממשי, אך ברגע עזיבה מוחלטת התקדמות כלשהיא תמחק לצמיתות!)

2. Info: יכניס אותך למסך המידע, שם נמצא כל המידע שהמשתמש צריך לדעת כדי לשחק,

כולל הוראות וחוקים. במסך המידע יש עוד אפשרות ניווט:

a. Esc: יוציא אותך חזרה למסך הראשי ותוכל לבצע את הניווט משם בהתאם.

3. Exit: התוכנית תיסגר והאמולטור יחזור למצב טקסט.

# תיאור הפתרון:

## פרוצדורות ראשיות:

יש לי שתי פרוצדורות ראשיות:

1. Menu: פרוצדורת התפריט זוהי פרוצדורה שאיתה השחקן משתמש כדי לנווט בתפריט, בין 3 אפשרויות, יציאה, מידע, ושחקן. הפרוצדורה בנויה כך שלמשתמש יהיה קל להשתמש ב-Graphic User Interface ואני נותן לו חיוויים על המסך על היותו עומד על כפתור מסוים, בעת הלחיצה התוכנית במגיבה בהתאם לבחירת המשתמש ומנווטת בתפריט בהתאם. בתוך המסך המידע ישנו גם כפתור שגם כן פונקציה זו אחראית על ניווט במסך המידע. הפרוצדורה מחולקת ל-5 שלבים מחזוריים:
  - a. בדיקת המצאות העכבר (האם ב-PLAY האם ב-EXIT)
  - b. הדפסת המסך שבו הכפתור מודגש בהתאם לשלב a (אם העכבר לא נמצא על כפתור אז לא קורה כלום)
  - c. בדיקה מחזורית על העכבר אם עדיין נמצא על הכפתור, אם לא, חוזר לשלב a
  - d. בדיקה האם העכבר נלחץ, אם לא אז חוזר לשלב c
  - e. הדפסת המסך המתאים / יציאה במקרה ומדובר על EXIT

2. Game: פרוצדורה זו מנהלת את מהלך המשחק. היא אחראית על שתי דברים עיקריים:
  - a. ניווט העכבר במשחק
  - b. הפעלת תור

הפרוצדורה בודקת את המצאותו של סמן העכבר והאם נלחץ ומחליטה מה לעשות. זוהי הפרוצדורה הראשית של המשחק היא קוראת לכל שירשור פרוצדורות העזר.

## קלט פלט:

הקלט בפרוייקט שלי הוא אך ורק מהעכבר. אני מסוגל לזהות על ידי פונקציות שכתבתי:

- האם העכבר נמצא בשטח מסוים על המסך ומחזירה ערך בוליאני בהתאם
- האם העכבר נלחץ ומחזירה ערך בוליאני בהתאם

הפלט שלי נעשה על ידי תמונות וחיוויים על המסך,

- הבלטת כפתור על המסך כאשר נקלט שהעכבר נמצא עליו
- נצחון שלי או של היריב ניתן לזהות על ידי כתר שמודפס ליד החפיסה בהתאם

## הנתונים:

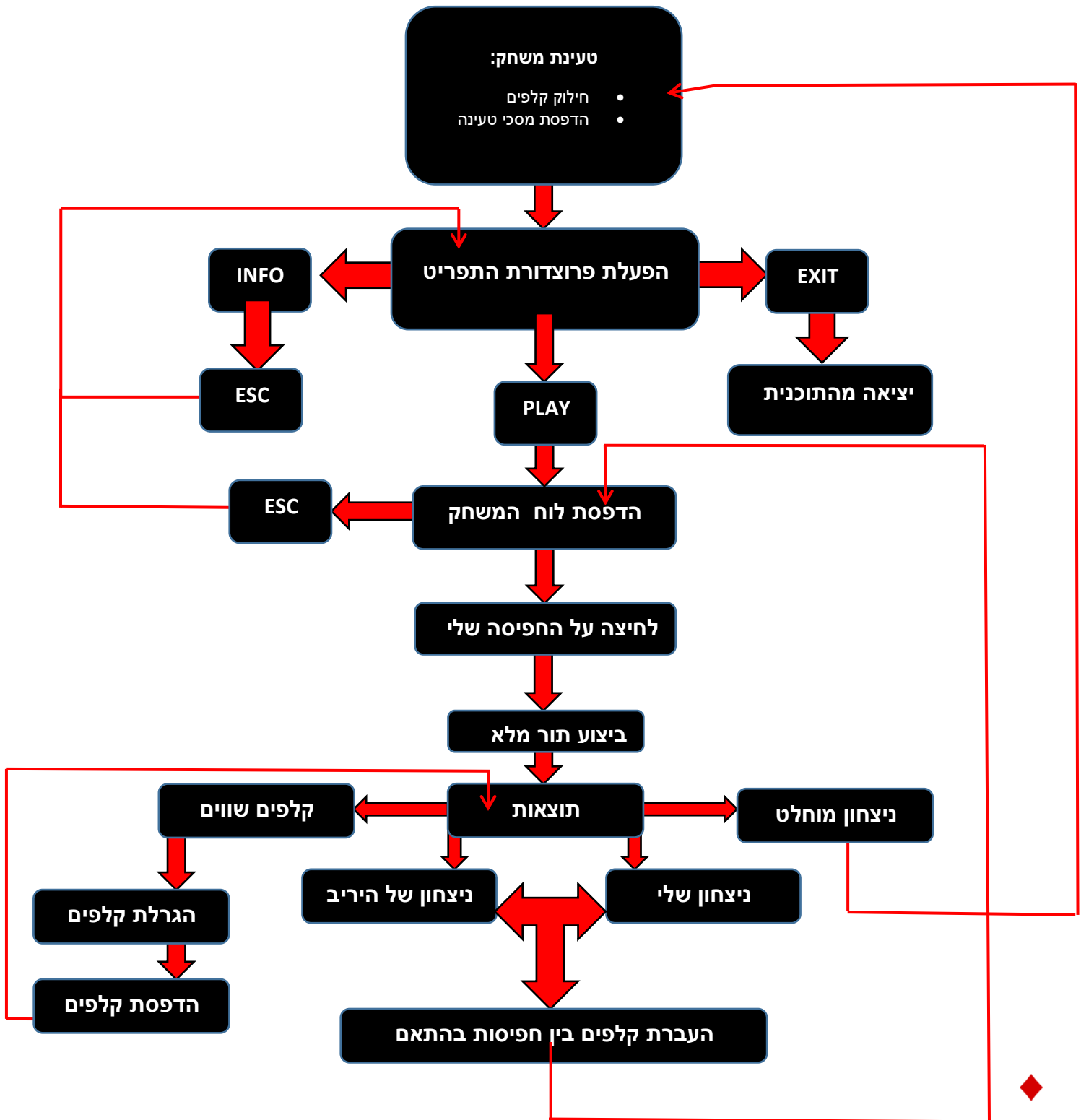
הנתונים מתחלקים לשני קבוצות:

1. הקבועים: הקבועים נמצאים בקובץ נפרד (Const) ומאוד קל לנווט בקובץ זה ולראות את כל הקבועים, ועל ידי שם משמעותי ניתן לזהות גם למה הם משמשים ומייצגים.
2. נתונים ממשיים (נתונים שרובם מתעדכנים ומשתנים עם התקדמות המשחק) הנתונים הממשיים של המשחק מתחלקים לשתי קבוצות:
  - a. ייצוג מידע קבוע – זהו מערך ענק המכיל במקום הראשון שלו את מספר הקלף (מ 1 עד 52), בשש מקומות הבאים הוא מכיל את שם הקובץ המתאים לקלף, במקום השמיני הוא מכיל את הערך הממשי של הקלף. מערך זה משמש לקישור בין הקלף לבין הקובץ שלו בצורה נוחה וקלה לשימוש, מערך זה מכיל מידע חשוב על כל קלף.
  - b. בקרה חייה על קלפים – שימוש בשני מערכים לשני חפיסות, ביניהם מתבצעת ההעברה של הקלפים ובדיקות שונות במשחק, בנוסף למערכים אלו, קיימים עוד שני מערכים המכילים את הקלפים העכשוויים בתור זה, כך שמאוד נוח לעשות השוואת ערכים והעברת מידע בין מערך למערך.

## גראפיקה:

הגראפיקה במשחק שלי מורכבת אך ורק מתמונות BMP, ולא מערכי פיקסלים בשל הנראות המורכבת של הקלפים. בגלל שיקול זה הייתי צריך לפתח פרוצדורה שמדפיסה תמונה בגודל מסויים ובמיקום מסויים על המסך, כך שכל קלף שאנירוצה להדפיס מוגדר לו לפי המצב הספציפי איפה הוא מודפס לפי מיקומים בקבועים שאותם מדדתי לפני כן. את מסכי התפריט אני מדפיס בעזרת מאקרויים כך שבשורה אחת יש לי פקודה של הדפסה למסך ספציפי, זה יוצר יעילות ויכולת הבנת קוד יותר טובה ככה שניתן להבין בצורה ממש קלה איזה מסך עכשיו מודפס.

# תרשים זרימה:



# פרוצדורות:

## *Mouse.asm*

שם הפרוצדורה: InitMouse

קלט: -- אין --

ביצוע: איתחול והפעלת העכבר.

פלט: -- אין --

שם הפרוצדורה: GetMouseData

קלט: -- אין --

ביצוע: לקיחת מידע מהעכבר, המחשב מעדכן ושם את המידע ברגיסטרים המתאימים על מיקום העכבר ועל הלחיצות שלו.

פלט: -- אין --

שם הפרוצדורה: IsMouseClicked

קלט: דחיפת 0 למחסנית כדי ליצור מקום לערך מוחזר.

ביצוע: בדיקה בוליאנית, אם הכפתור השמאלי של העכבר נלחץ (לחיצה בלבד ולא לחיצה ועזיבה).

פלט: אמת במקרה ונלחץ, שקר אם אחרת.

שם הפרוצדורה: HideMouse

קלט: -- אין --

ביצוע: העלמת סמן העכבר מן המסך.

פלט: -- אין --



שם הפרוצדורה: ShowMouse

קלט: -- אין --

ביצוע: הופעת סמן העכבר על המסך.

פלט: -- אין --

## ***Menu.asm***

שם הפרוצדורה: Menu

קלט: -- אין --

ביצוע: אחראית לטיפול העברת המסכים בתפריט והדפסתם, מאפשרת ניווט בתפריט.

פלט: -- אין --

## **Game.asm**

שם הפרוצדורה: Game

קלט: -- אין --

ביצוע: אחראית לניווט במשחק, (יציאה או ביצוע תור). זו הפונקציה הראשית של המשחק כך שהיא מריצה את כל פרוצדורות העזר שהמשחק צריך.

פלט: -- אין --

שם הפרוצדורה: Random

קלט: מספר אקראי מקסימלי , מספר אקראי מינימלי.

ביצוע: יוצרת מספר רנדומלי שעונה לטווח המספרים שנקלט.

פלט: מספר רנדומלי .

### שם הפרוצדורה: PrintCard

**קלט:** איזה קלף, ובאיזה חפיסה (אמת בשביל החפיסה שלי, שק בשביל החפיסה שלו, ושתי מצבים מיוחדים במלחמה).

**ביצוע:** הדפסת הקלף במיקום המוגדר שהתקבל.

**פלט:** -- אין –

### שם הפרוצדורה: TakeACardFromMy

**קלט:** -- אין –

**ביצוע:** אחראית על בחירת הקלף הרנדומלי במערך של החפיסה שלי והדפסתו (על ידי קריאה PrintCard ) ולעדכן את הפרמטרים המתאימים בהתאם. אחראית על ביצוע התור שלי.

**פלט:** -- אין –

### שם הפרוצדורה: TakeACardFromOpponent

**קלט:** -- אין –

**ביצוע:** אחראית על בחירת הקלף הרנדומלי במערך של החפיסה של היריב והדפסתו (על ידי קריאה PrintCard ) ולעדכן את הפרמטרים המתאימים בהתאם. אחראית על ביצוע התור של היריב.

**פלט:** -- אין –

### שם הפרוצדורה: FindActualValue

**קלט:** איזה קלף.

**ביצוע:** מוצאת את הערך האמיתי של הקלף .

**פלט:** הערך האמיתי של הקלף.

### שם הפרוצדורה: WhoWins

**קלט:** לאיזה ערך במערך המכיל את הקלפים העכשוויים להתייחס (הקלפים שמוצגים על המסך והוצאו מהחפיסה על ידי המשתמש בתור זה).

**ביצוע:** פעולה בוליאנית שמוצאת מי ניצח בתור זה או אם יש מלחמה, פעולה המשווה בין הערכים האמתיים של הקלפים המוצגים.

**פלט:** אמת אם אני ניצחתי, שקר אם היריב ניצח, ערך המייצג מצב מיוחד: קלפים שווים.

### שם הפרוצדורה: IsValid

**קלט:** קלף רנדומלי לבדיקה, לאיזה מערך של קלפים עכשוויים להתייחס בבדיקה (שלי או של היריב).

**ביצוע:** פעולה בוליאנית שבודקת אם הקלף הרנדומלי שהוגרל הוא תקין לביצוע מהלך המלחמה (לא שווה לאף מהקלפים שהושמו כבר במהלך).

**פלט:** אמת אם תקין, שקר אם לא תקין.

### שם הפרוצדורה: GetCardIndexValue

**קלט:** מיקום במערך, איזה מערך (אמת למערך שלי, שקר למערך של היריב).

**ביצוע:** מוצאת את הערך שנמצא בתוך התא הנקלט בתוך המערך.

**פלט:** מחזירה את הערך הנמצא במיקום במערך.

### שם הפרוצדורה: MakeRandomsForWarState

**קלט:** -- אין –

**ביצוע:** מייצרת קלפים רנדומלים תקינים למצב בו הקלפים שווים.

**פלט:** -- אין –

**שם הפרוצדורה:** TransferAtWarState

**קלט:** -- אין –

**ביצוע:** מטפלת בהעברת הקלפים בין החפיסות במצב בו הקלפים שווים.

**פלט:** -- אין –

**שם הפרוצדורה:** WarStateHandler

**קלט:** -- אין –

**ביצוע:** הפרוצדורה הראשית לטיפול במצב בו הקלפים שווים קוראת לכל פרוצדורת העזר לביצוע המהלך.

**פלט:** -- אין –

**שם הפרוצדורה:** FindEmptySlots

**קלט:** איזה מערך למצוא מקומות ריקים

**ביצוע:** מוצאת מקומות ריקים במערך (בעלי ערך 0).

**פלט:** הכתובת בזיכרון בה במערך מופיע לראשונה מקום ריק, אם אין מקומות ריקים אז תחזיר קבוע, אין אפסים במערך.

**שם הפרוצדורה:** MakeTheTransfer

**קלט:** -- אין –

**ביצוע:** הפרוצדורה הראשית לביצוע המעברים וההתפלגויות למצבי קיצון בעת התור, פרוצדורה זו אחראית למעבר הקלפים לחפיסה של המנצח, ובנוסף אחראית למציאת המנצח הסופי.

**פלט:** -- אין –

שם הפרוצדורה: GameOverHandler

קלט: --אין--

ביצוע: איפוס המערכים להתחלת משחק מחדש.

פלט: --אין--

## *Print.asm*

שם הפרוצדורה: OpenFile

קלט: -- אין --

ביצוע: פותחת את הקובץ .

פלט: -- אין --

שם הפרוצדורה: ReadHeader

קלט: -- אין --.

ביצוע: קוראת את הHeader של קובץ הBMP 256

פלט: -- אין --

שם הפרוצדורה: ReadPalette

קלט: -- אין --.

ביצוע: קבלת צבעי הפלטה המתאימים לתמונה.

פלט: -- אין --



**שם הפרוצדורה:** CopyPal

**קלט:** -- אין --.

**ביצוע:** העברת צבעי הפלטה לזיכרון.

**פלט:** -- אין --

**שם הפרוצדורה:** CopyBitmap

**קלט:** -- אין --.

**ביצוע:** הופך את הדפסת התמונה להדפסה הרצויה (הצבעים מועתקים הפוך לזיכרון).

**פלט:** -- אין --

**שם הפרוצדורה:** CloseFile

**קלט:** -- אין --.

**ביצוע:** סוגר ומאפס את הקובץ של התמונה הפתוחה וכל הפרמטרים שקשורים להדפסה זו.

**פלט:** -- אין --

**שם הפרוצדורה:** DoDealy

**קלט:** כמות הפעמים שיש לכפול 0.055 כדי להגיע לדיליי הרצוי.

**ביצוע:** תוקעת את המחשב בלולאה למשך הזמן הרצוי, ככה שנראה כי המחשב "עושה פריז".

**פלט:** -- אין --.

**שם הפרוצדורה:** CopyBitmapForPrintByPosition

**קלט:** קורדינטות X ו Y והרזולוציה של התמונה (אורך ורוחב).

**ביצוע:** מעתיקה את קובץ ה BMP למיקום ובגודל הרצוי

**פלט:** -- אין --.

**שם הפרוצדורה:** IsInBoundaries

**קלט:** כאורדינטות  $X$  של הפינות (שמאלית וימנית) , כאורדינטות  $Y$  של הפינות (עליונה ותחתונה).

**ביצוע:** פעולה בוליאנית שבודקת אם סמן העכבר נמצא בטווח שנקלט.

**פלט:** אמת אם העכבר נמצא בשטח, שקר אחרת.

# דוגמת ריצה:

התפריט הראשית:



המשתמש בוחר PLAY:





## מסך המשחק:

ESC



## ביצוע תור וחיווי הניצחון:

ESC



# סיכום אישי:

פרויקט זה, הוא הפרויקט הכי גדול שעשיתי אי פעם בתחום הבית ספר, כמות השעות שעבדתי עליו מטורפת. בהתחלה שניתנה לנו האופציה לבחור משחק, בחרתי ב-DINO RUN, הדינוזאור המקפץ של גוגל אפליין.

אני יכול להגיד שאת החודש וחצי הראשונים, לא באמת התקדמתי כל כך, כי הייתי כמו ילד שזרקו אותו למים ללא שום מצופים, לא ידעתי איך לחבר את החומר שלמדנו לתוך משחק בסדר גודל כזה. כאשר הסתיימה עונת הרובוטיקה הבנתי שאני צריך להתקדם ומהר, החלפתי משחק, קצת יותר פשוט, מלחמה, בחרתי דווקא לעשות את המשחק הזה ולא את הדינוזאור כי הדינוזאור זה המון התעסקות עם לוגיקות גרפיקה, ופחות התחברתי, רציתי משחק פשוט גראפי ומסובך לוגית, מלחמה.

במהלך החודש וחצי שנותרו לי כתבתי בעצם את כל הפרויקט, נתקלתי בהמון קשיים, אבל בזכות הדיבאגר והמון המון, שעות ולפעמים גם עזרה מחברים ומהאינטרנט הצלחתי לעלות על התקלה, שכן ברוב המקרים הייתה די טיפשית. אני יכול להעיד שיותר מפעם אחת קרה כי מחקתי פרוצדורות שלמות כי "איבדתי עליהן שליטה" הרגשתי שאני כבר לא יודע לדבג את מה שכתבתי אז כתבתי מחדש, בצורה יותר יעילה, מובנת וקריאה.

את הפרויקט אני כותב לאחר שנה שלמה של התעסקות רבה בשפת ב JAVA ברמה די גבוהה אז היה לי מאוד חשוב ליישם כמה שיותר דברים שלמדתי משם, יעילות, סדר, מובנות ובעיקר הקפדתי לרשום מילולי, כלומר שאין מקומות עם מספרים מפוזרים, הכל עם קבועים כך שניתן ממש בקלות להבין את הקוד.

נעזרתי בשני מקורות מידע אינטרנטיים עיקריים:

1. ספר הלימוד "ארגון המחשב ושפת סף" שכתב ברק גונן

[http://www.lamed-oti.com/school/gvahim\\_assembly\\_book.pdf](http://www.lamed-oti.com/school/gvahim_assembly_book.pdf)

2. אתר Stack Overflow

<https://stackoverflow.com/>

