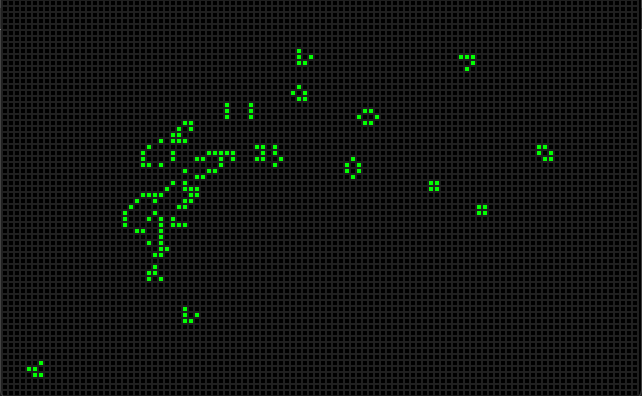
תיק פרויקט אסמבלר

איתמר שוורץ 329842272

מורה המלווה את הפרויקט: מרי גבע

תאריך הגשה: 29.1.23



תוכן עניינים

עמוד-3 הסבר על המשחק

עמוד-4 מהלך המשחק

עמוד-5 התקדמות המסכים

עמוד-6-7 רשימת משתנים

עמוד 8-10 רשימת פעולות

עמוד 11 מדריך למשתמש

עמוד 12 רפלקציה

הפרויקט הוא מימוש של משחק מוכר משנות ה 70 המשחק שנקרא conways game of life. המשחק מתחיל בלוח של משבצות שממלא את כל המסך, כל אחת מהמשבצות יכולה להיות דלוקה או כבויה וכולן מתחילות כבויות. השחקן מתחיל את המשחק בללחוץ על המשבצות בכדי להגדיר את המצב ההתחלתי שלהן. לחיצה על משבצת כבויה תדליק אותה ולהפך. ברגע שהשחקן מוכן הוא יכול ללחוץ על כל כפתור ומשם המשחק ירוץ.

המשחק רץ בסבבים, כל סבב נוצר לוח חדש של משבצות על פי הלוח הקודם.

יש ארבעה חוקים לגבי המצב של משבצת בסבב הבא והם:

1. אם יש למשבצת דלוקה פחות משני שכנים בסבב הבא היא תהיה כבויה
2. אם יש למשבצת דלוקה יותר משלושה שכנים בשבב הבא היא תהיה כבויה
3. אם יש למשבצת דלוקה שניים או שלושה שכנים בסבב הבא היא תישאר דלוקה
4. אם למשבצת כבויה יש בדיוק שלושה שכנים סבב הבא היא תדלק

כל סבב החוקים האלה נבדקים על כל משבצת ונכתבים ללוח בסבב הבא

אפשר לצאת מהמשחק בלחיצה על esc ואפשר להפסיק את הסבבים בלחיצה על p

במימוש שלי הלוח הוא מעגלי, זה אומר שהשכנים של משבצות הקצה הן המשבצות בקצה בצד השני. לעומת המשחק המקורי שבו הלוח היה אינסופי והיה אפשר לגרור את המסך כדי לזוז בתוך הלוח האינסופי

בחרתי במשחק הזה בגלל שהוא הלהיב אותי מאוד אחרי שראיתי את מה שאומרים על המשחק בהרצאה [הזאת](https://youtu.be/6avJHaC3C2U?t=280)https://youtu.be/6avJHaC3C2U?t=280 (הקישור מתוזמן לרגע הנכון בהרצאה) הלהיב אותי העובדה שהוא turing complete ושאפשר לכתוב בו תוכנות.

מהלך המשחק

המשחק רץ?

לא

לא

כן

נגמר המשחק

לחצו esc?

כן

המשחק לא רץ.

לחצו p?

ייצור של לוח המשחקים החדש ציור הלוח החדש והחלפת תפקידי הלוחות

כן

כן

הפיכת המשבצת שלחצו עליה לצבע ההפוך ממנה.

לא

לחצו על המסך?

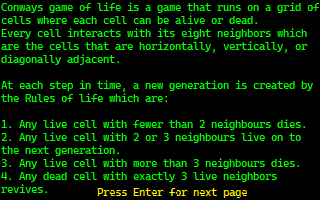
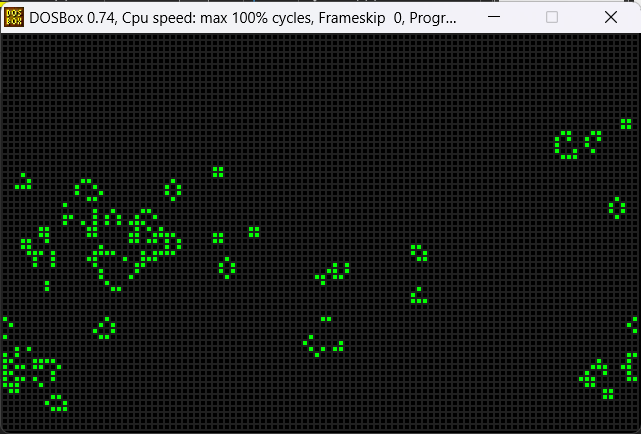
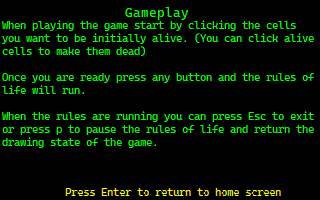
לא

המשחק רץ.

כן

לחצו על כפתור במקלדת?





לחיצה עם העכבר על משבצות ואז לחיצה על אחד הכפתורים המקלדת (כל כפתור עובד)

לחיצה על enter

לחיצה על enter

לחיצה על enter

לחיצה על r

רשימת משתנים

|  |  |
| --- | --- |
| SQUARE\_SIZE | Number of pixels the width and height of each square are |
| LINE\_COLOR | Color of the lines of the grid |
| FILLED\_SQUARE\_COLOR | Color of filled Squares |
| NON\_FILLED\_SQUARE\_COLOR | Color of non-filled Squares |
| SCREEN\_WIDTH | 320 |
| SCREEN\_HEIGHT | 200 |
| VERTICAL\_OFFSET | Amount of pixels vertically between edge of screen to end of grid, (gridWidth != screenWidth), grid is calculated to fit the maximum number of squares according to square size  SCREEN\_WIDTH%SQUARE\_SIZE |
| HORIZONTAL\_OFFSET | Amount of pixels horizontally between edge of screen to end of grid, (gridWidth != screenWidth), grid is calculated to fit the maximum number of squares according to square size  SCREEN\_HEIGHT%SQUARE\_SIZE |
| LEN\_VERTICAL | Number of pixels in grid vertically.  SCREEN\_HEIGHT - VERTICAL\_OFFSET |
| LEN\_HORIZONTAL | Number of pixels in grid horizontally.  SCREEN\_WIDTH - HORIZONTAL\_OFFSET |
| AMOUNT\_OF\_SQUARES\_HORIZONTAL | Number of Squares that fit in grid horizontally  floor(SCREEN\_WIDTH / SQUARE\_SIZE) |
| AMOUNT\_OF\_SQUARES\_VERTICAL | Number of Squares that fit in grid vertically  floor(SCREEN\_HEIGHT / SQUARE\_SIZE) |
| SquareColor | A parameter for DrawSquare procedure tells the procedure what color to draw the square |
| Board0  Board1 | Arrays that hold the state of the grid  There are two because each iteration one is drawn and one is written to and at the and of the iteration they switch role  Picked arbitrary size of 8000 not recomended to make square size less than 3 bad things will happen! |
| CurrentBoard | Holds current board that is shown: 0 | 1 |
| Clock | Address of clock |
| LongDelay | Amount of tick to wait in Delay procedure |
| Palbuffer | Used to save initial pallet and restore it so that mouse ins't red and stays white |
| CurrentFile | Holds address of string that it the name of the bmp file to show |
| RulesScreen1  RulesScreen2  OpeningScreen | File names for bmp files |
| Filehandle  Header  Palette  ScrLine | Variables for reading and showing bmp file |
| ErrorMsg | Error message to show when there is a problem opening file |

רשימת פעולות

Board coords - coordinates according to squares.

Each square has its own coordinate top left is (0, 0)

Screen coords - coordinates according to pixels on screen

|  |  |
| --- | --- |
| SavePal | Palbuffer: a buffer that will hold current RGB values of palette.  Saves current palette in Palbuffer. |
| RestorePal | Palbuffer: a 256\*3 buffer with saved RGB values of a 256 color palette.  Changes program palette to the one in the Palbuffer. |
| ColorPixel | bx: X - Screen coords  dx: Y - Screen coords  al: Color  Draws pixel at the coordinates of x,y with the color |
| DrawLineHorizontal | dx: Y for line - Screen coords  al: Color  Draws horizontal line from 0 to LEN\_HORIZONTAL |
| DrawLineVertical | bx: X for line - screen coords  al: Color  Draws vertical line from 0 to LEN\_VERTICAL go to vars file for explanation for LEN\_VERTICAL |
| DrawSquare | ax: X - Board coords  dx: Y - Board coords  SquareColor: Color of square  Draws a square at coordinates in the color of whatever is stored in SquareColor |
| DrawGrid | Draws grid based on square size |
| SetBoardAtCord | ax: X - Board coords  dx: Y - Board coords  bl: FILLED\_SQUARE\_COLOR | NON\_FILLED\_SQUARE\_COLOR state of square  Changes cell at (x,y) on CurrentBoard to what is stored at bl |
| InitVars | Initializes: AMOUNT\_OF\_SQUARES\_HORIZONTAL, HORIZONTAL\_OFFSET, AMOUNT\_OF\_SQUARES\_VERTICAL, VERTICAL\_OFFSET |
| SetOtherBoardAtCord | ax: X - Board coords  dx: Y - Board coords  bl: FILLED\_SQUARE\_COLOR | NON\_FILLED\_SQUARE\_COLOR    Changes cell at (x,y) on opposite board to what is stored in bl |
| ToggleBoardAtCord | ax: X - Board coords  dx: Y - Board coords  Toggles Color of squares at (x,y) to opposite color |
| GetBoardAtCord | ax: X - Board coords  dx: Y - Board coords  returns byte that is the color of the board at that coord inside bl, bh is 00H |
| SwapBoard | Swaps current board to be next board  1 - [CurrentBoard] == OtherBoard |
| Delay | [Clock],es:6ch : Current Clock ticks  [LongDelay]: Number of clock ticks to wait until the procedure returns.  Performs a time delay according to the value in [LongDelay] |
| AddneighboursToBx | ax - X Board coords  dx - Y Board chords  Increments bx if the point at those coordinates is alive  Coordinates come incremented or decremented from ConwaysGameOfLifeSingleCell  So it also checks if those coordinates are inside grid and if they aren't it uses the cell from the other side of the board making the game cyclic |
| ConwaysGameOfLifeSingleCell | ax: X - Board coords  dx: Y - Board coords  Runs the rules of life on (x,y) in current board and writes results to other board |
| ConwaysGameOfLife | Runs rules of life on all cells in grid |
| ShowBmp | Shows bmp file saved in CurrentFile |
| OpenFile | CurrentFile: file name of bmpfile to draw  Opens file, puts handle in filehandle |
| ReadHeader | Read BMP file header, 54 bytes  Put BMP file Header in Header variable |
| ReadPalette | Read BMP file color palette, 256 ; colors \* 4 bytes (400h)  Puts BMP file Pallete in Pallete variable |
| CopyPal | Copy the colors palette to the video memory registers  The number of the first color should be sent to port 3C8h  The palette is sent to port 3C9h |
| CopyBitmap | BMP graphics are saved upside-down.  Read the graphic line by line (200 lines in VGA format),  displaying the lines from bottom to top. |
| CloseFile | filehandle - filehandle of bmp file  closes the bmp file |

מדריך למשתמש

קבצים ששימושים למשתמש:

|  |  |
| --- | --- |
| comp.bat | סקריפט שמקמפל את הקוד לי בינארי שאפשר להריץ בשם base.exe |
| vars.asm | בתחילת קובץ הזה יש מספר משתנים שמגדירים דברים באפליקציה אפשר לשחק עם זה אם המשתמש רוצה לדוגמא שהקוביות יהיו גדולות יותק |

קבצי הפרוקייט:

|  |  |
| --- | --- |
| base.asm | קובץ הכניסה של הקוד מגדיר את לוגיקת המסכים ומריץ את הפרוצדורות שמשחקות את המשחק |
| Bmpfile.asm | מגדיר פרוצדורות שמציגות קובץ bmp |
| Board.asm | מגדיר פרוצדורות שקשורות לעבודה עם הלוח משבצות כמו לערוך אותו |
| Delay.asm | קובץ המגדיר פרוצדורה של דיליי שמחכה כמות טיקים שך השעון |
| Drawing.asm | מגדיר פרוצדורות שקשורה לציור של הפיקסלים על המסך |
| Fixmouse.asm | מגדיר פרוצדורות שמתקנות את הpallete כדי שהעכבר לא יהיה אדום |
| Game.asm | מגדיר פרוצדורות שקשורות לחוקי המשחק ולהרצה שלהם |
| Initvars.asm | מגדיר פרוצדורה שמהתחלת את המשתנים |
| Macros.asm | מגדיר כמה מקרואים ששימושיים בקוד |
| Vars.asm | מגדיר את המשתנים בקןד |

אפשר להריץ את המשחק dosbox עזרת שתי הפקודות האלה:

comp

base

רפלקציה

בהתחלה חששתי מפיתוח הפרוייקט ופחתי שיהיה לי מאוד כשהוא אבל ברגע שהתחלתי הבנתי שזה לא מפחיד כמו שחשבתי ואפילו יחסית פשוט אז יצא שסיימתי לכתוב את כל הקוד בארבעה ימים. למדתי בפרוויקט בעיקר שאני לומד הרבה יותר טוב מעשייה מאשר לשבת ולעבור על מצגות/לשבת בכיתה. אתגר דיי גדול שהיה לי בפיתוח הוא השילוב של פיתוח הפרויקט עם העומס הגדול שיש לי לכן החלטתי לעשות פרוייקט יחסית מינימלי ועשיתי את הפרוייקט הזמן חופש חנוכה שלא הייתי מאוד עסוק. אם היה לי עוד זמן לפתח את הפרוייקט אולי הייתי מוסיף אופציה לשנות את המהירות של המשחק וגם הייתי מנסה לעשות את המשחק אינסופי כמו המקורי אבל שם זה נהיה הרבה יותר מסובך. תודה ליותם גל שעזר לי המון בהתחלה של הפיתוח של האפליקציה וישב איתי מספר שעות ברציפות כדי לגרום לדברים לעבוד.