פרויקט יחידה אסמבלי32 2018 - Chrome dino

כיתה: י'3

מנחה: יהודה אור

מגיש: אופק כפיר – 323821991

תאריך הגשה – 20.06.2018

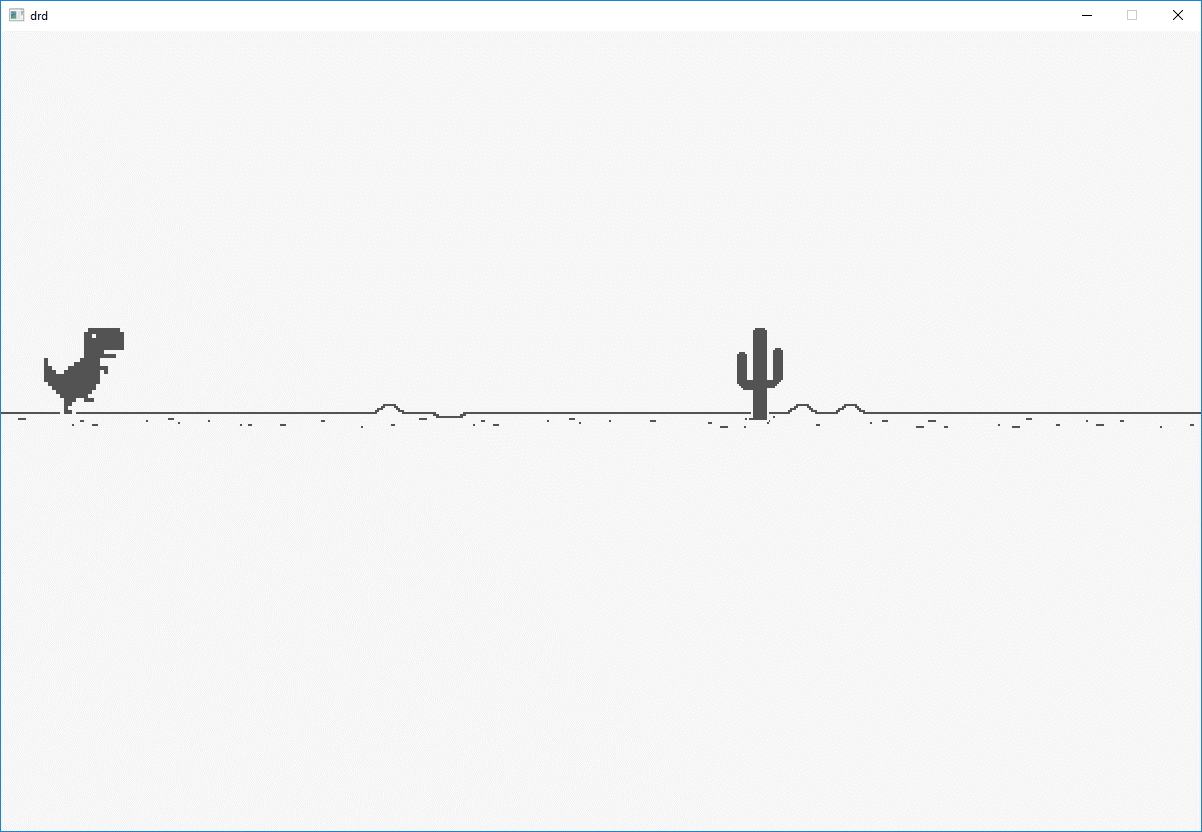


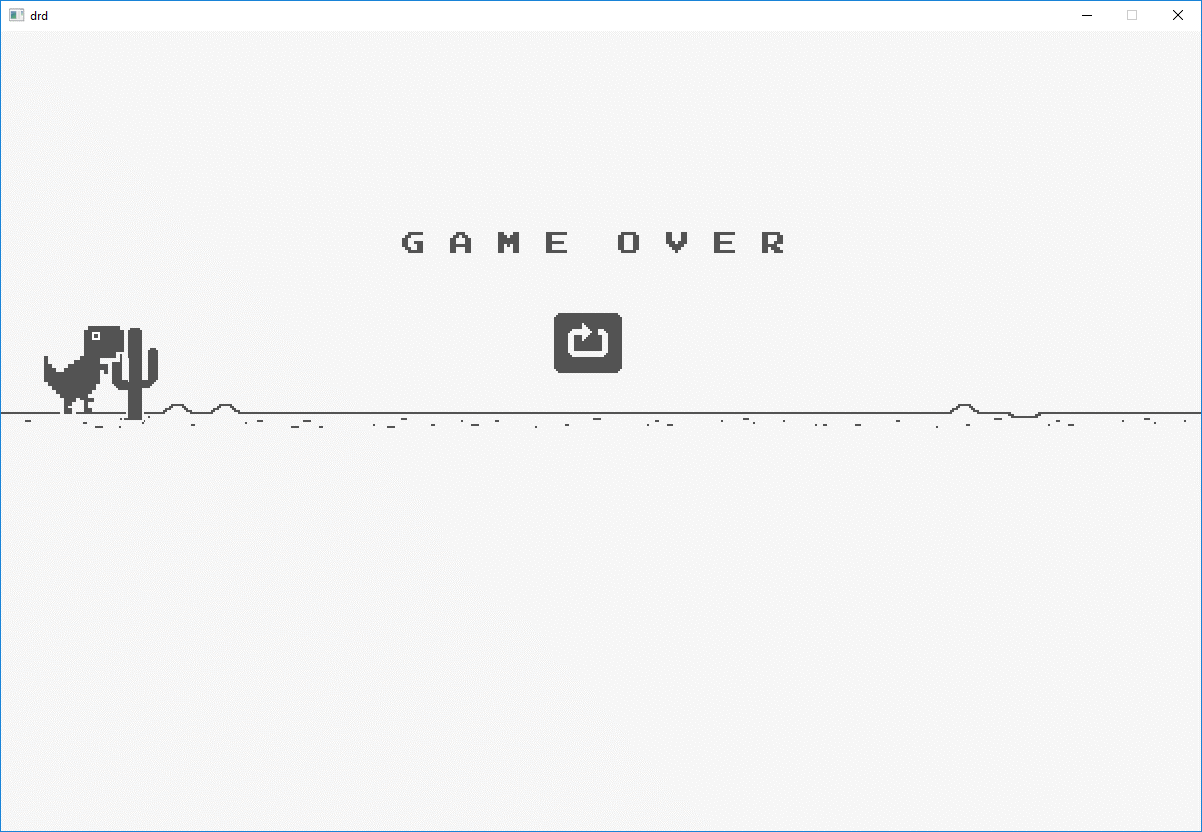
תוכן עניינים:

רפלקציה:

במהלך העבודה על המשחק, נתקלתי בבעיות רבות שלא ידעתי את פתרונם והייתי צריך להשתמש באינטרנט פעמים רבות על מנת למצוא פתרונות לבעיות אשר אתגרו את המחשבה שלי. (למשל: לא ידעתי שedx מאחסן את השארית של המספר ולכן צריך לאפס אותו לפני פעולות מתמטיות). למדתי איך לחפש ממוקד באינטרנט פתרונות לבעיות שלי, ולשאול בפורומים. במהלך העבודה השתמשתי במדריכים רבים ברשת אשר עזרו לי בהבנת השפה ואיך הדברים עובדים במחשב ובראשם **ארגון המחשב ושפת סף מאת ברק גונן** במהלך העבודה גם למדתי איך להשתמש בסביבת עבודה של Masm32 וVisual studio ולמדתי גם איך להשתמש בכלי השימושי שנקרא Git על מנת לשמור את העבודה שלי ולתעד את השינויים עם הזמן. (הקישור לrepo יצורף בסוף הספר תחת "קישורים".)

תמונות מהמשחק:





קצת על אסמבלי והמחשב

המחשב כיום מורכב מרכיבים רבים: מעבד, דיסק קשיח, ראם, לוח אם, מערכת הפעלה, המעבד הגרפי, המסך ועוד. בין כל הרכיבים קיימת הרמוניה ולכל רכיב תפקיד שונה במחשב הייחודי לו, אני ארחיב על כל אחד מהרכיבים אשר אנו מתעסקים בהם כיום.

המעבד - CPU (Central processing unit )

המעבד הינו היחידת עיבוד המרכזית של המחשב, בו מתבצעים כל החישובים של המחשב והפקודות ההכרחיות לכל יחידה דיגיטלית. בכל מעבד ישנם כמה מרכיבים הכרחיים:

* אוגרים – מיועדים לאחסון ערכים ופעולות מתמטיות (הרחבה בהמשך)
* יחידה לוגית – אחראית על הפעולות המתמטיות ועל השערים הלוגים (XOR,AND,NOR,OR... (
* יחידת בקרה – אחראית על קריאת ופיענוח הפקודות.

לכל מעבד ארכיטקטורה בלעדית משלו על פי היצרן של המעבד, בפרויקט זה השתמשנו בארכיטקטורת 8086 מאת אינטל אשר יצאה ב1976 (עובדת בונוס: לכבוד ה40 שנה יצא השנה מעבד i7-8086K אשר נמצא עכשיו בשוק במהדורה מוגבלת). ל8086 של אינטל יצאו המשכים רבים עד היום והם נחשבים כחלק ממשפחת X86 שהיא הארכיטקטורה הנפוצה ביותר כיום במחשבים.

זיכרון (ראם)

הראם הינו הזיכרון המהיר במחשב, והוא נועד על מנת לגשת במהירות אל התכנית ולאחסון הנתונים של התכנית במקום לגשת כל פעם אל הדיסק הקשיח שנחשב זיכרון איטי. הזיכרון המהיר במחשב מחולק ל3 חלקים:

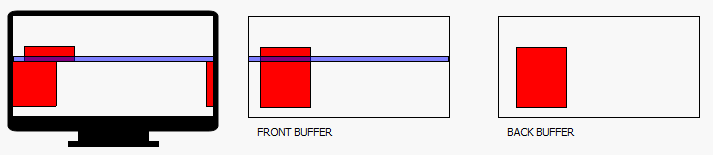
* הראם- נועד על מנת לרשום את השפת מכונה ואת הקוד שלנו.
* קלט/פלט – נועד על מנת לקלוט את המכשירים השונים של המחשב, כמו המקלדת.
* ROM (Read only memory )- חלק זה בזיכרון אינו ניתן לכתיבה ומאוחסנים בו הוראות של היצרן על מנת שהמחשב יפעל כמו שצריך, זיכרון זה ניתן לקריאה אך לא לכתיבה.

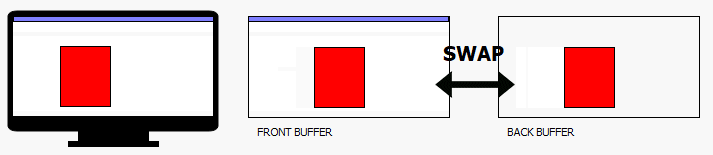
תוכן הראם נמחק ברגע שמנתקים אותו מהחשמל, ולכן הוא זיכרון זמני.

המעבד הגרפי

המעבד הגרפי הינה יחידת עיבוד שנמצאת במעבד או ביחידה נפרדת מהמעבד והיא אחראית על התמונה במחשב והצגתו למסך, המעבד הגרפי מציג את התמונה כמטריצה גדולה של פיקסלים למסך והוא יכול לבצע חישובים מסובכים שהCPU לא יכול.

המעבד הגרפי עובד ככה שהוא מצייר לבאפר האחורי את התמונה הבאה שהמחשב רוצה להראות, ומעביר אותה לבאפר הקדמי שמשם המסך לוקח את התמונה. בזמן שהמסך מצייר את התמונה מהבאפר הקדמי, הבאפר האחורי כבר עסוק בלצייר את התמונה הבאה וכך הלאה.





הדיסק הקשיח

הדיסק הקשיח הינו המקום בו מאחסנים את הקבצים הדרושים להפעלת התוכנה, כמו התמונות והסאונד. הדיסק הקשיח הוא אחסון תמידי ונועד לטווח ארוך והוא נשמר גם כאשר הוא מנותק מהחשמל.

אז מה זה אסמבלי?:

אסמבלי היא שפת סף אשר נחשבת הכי קרובה למחשב אחרי שפת מחשב (בינארית). אסמבלי לעומת שפות עיליות כמו Java, C#, C++ נחשבת שפה שקשה לקרוא, אם תסתכל במבט ראשון על קוד באסמבלי ככל הנראה שלא תבין כלום רק אחרי קריאה והבנה מעמיקה, לעומת Java למשל שכל פקודה שם ברורה למוח האנושי גם אם אין רקע נרחב בתכנות. באסמבלי אתה נדרש להגיד למערכת מה לעשות בדיוק ובלי ידע נרחב באיך עובד המחשב והרכיבים בו לא תצליח לכתוב קוד. באסמבלי יש שימוש נרחב באוגרים ובזיכרון ואם אתה עובר על משהו שמנוגד לשפת המחשב המחשב יודיע לך על זה מיד.

אז כמו שאתם שואלים, למה בכלל צריך לתכנת באסמבלי אם אפשר להשתמש בשפות עיליות אשר יותר שימושיות וקלות? ישנן כמה סיבות מרכזיות:

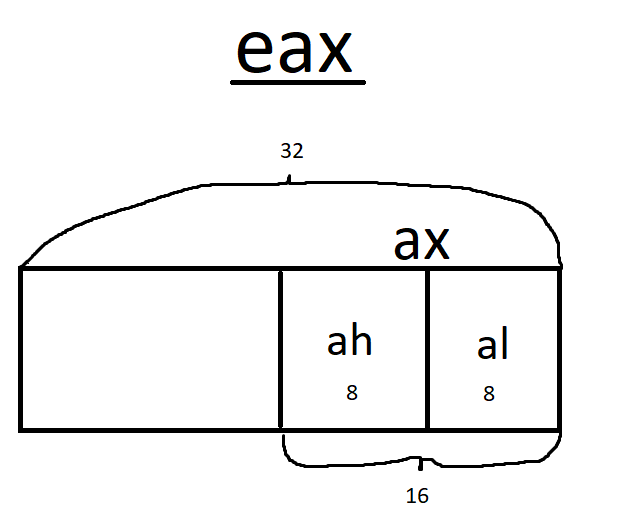
* אסמבלי עוזרת להבנה מעמיקה של המחשב וחלקיו, דבר שיעזור בעתיד בהבנה של בעיות תכנות לא שגרתיות.
* בגלל שאסמבלי כל כך קרובה למחשב, השפה הינה יעילה מאוד מבחינת משאבים לעומת שפות עיליות.
* אם צריך לרשום משהו שקרוב למחשב כמו דרייבר, דברים יכולים להשתבש מאוד מהר ברגע שרושמים אותם בשפות עיליות, ולכן כדאי לדעת אסמבלי על מנת לתקן בעיות אלו.

במהלך העבודה השתמשתי בסביבת העבודה של מיקרוסופט Visual studio ביחד עם ספריית masm32 שהיא בעצם הספרייה של מיקרוספוט משנות ה90 ועם הספרייה הגרפית drd מאת שי שלום. Visual studio הינה סביבת עבודה שימושית ביותר הכוללת Git, דיבאג יעיל שכולל צפייה בערכים, סדר של הקבצים והמשאבים.

פונקציות של DRD אשר השתמשתי בהם בפרויקט והסבר:

|  |  |
| --- | --- |
| drd\_init | פונקצייה אשר פותחת חלון על פי x ו y |
| drd\_imageLoadFile | פונקצייה אשר טוענת את התמונה מהספריית קבצים בתוך התוכנה |
| drd\_imageSetTransparent | פונקצייה אשר מגדירה את הצבע שיש להסיר מהתמונה על מנת שהחלק המרכזי בתמונה יהיה לבד |
| drd\_pixelsClear | פונקצייה שמנקה כל דבר שיש בחלון |
| drd\_imageDrawCrop | פונקצייה שחותכת חלק מהתמונה ומציירת אותו על המסך |
| drd\_imageDraw | פונקצייה שמציירת תמונה על המסך |
| drd\_flip | פונקצייה אשר מציגה את מה שיש בפרונט באפר על המסך |
| drd\_processMessages | פונקצייה שמאפשרת לחלון לקבל פקודות מהמשתמש |

האוגרים:

במשפחת המעבדים x86 משתמשים באוגרים אשר הם 32 ביט. אוגר של 32 ביט מורכב משני אוגרים של 16 ביט ואוגר של 16 ביט מורכב מאוגרים של 8 ביט למשל מצורף סרטוט של האוגר eax (32 ביט) ואת האוגרים אשר כוללים אותו

מצורף רשימת האוגרים ב16 ביט לקוחה מהספר של ברק גונן – ארגון המחשב ושפת סף

