**מסמך פרויקט דיקומפלייר - תשע"ח**

**צוות:** יובל אלבר

**כיתה:** עכו 8

**ראש צוות:** אלכס זקשנסקי

**מנטור:** שם המנטור

*הסבר כללי על המסמך*

*מסמך שילווה אתכם לשבועות הקרובים וגם בהמשך השנה.*

*אין ליצור עותקים של המסמך, ויש להשאירו בפורמט של גוגל-דוק! כל דרך אחרת עלולה ליצור בעיות לראש הצוות או למנטור שלכם אשר עוקב אחר ההתקדמות בעבודה במסמך ובכלל. בזכות העבודה דרך גוגל-דוקס, יש גיבוי אוטומטי, סנכרון תוכן המסמך בין כל המשתתפים בו, ומעקב גרסאות.*

*בכל פעם שמשתנה או מתווסף תוכן מסוים למסמך, יש לציין זאת בתמציתיות בטבלת המעקב.*

# טבלת מעקב/עדכון גרסאות

|  |  |  |  |
| --- | --- | --- | --- |
| **תאריך** | **מבצע/ת העדכון** | **תוכן/מהות העדכון** | **הסעיפים שעודכנו** |
| 21.8.2018 | יובל אלבר | התחלת עבודה על מסמך ייזום | תיאור כללי, מטרת הפרוייקט... |
|  |  |  |  |
|  |  |  |  |

פרק 1: ייזום + אפיון

*הסבר על המסמך:*

*מטרת המסמך היא לפרט יותר לעומק על הפרויקט. דמיינו שמסמך זה יינתן למפתח שיפתח במקומכם את הפרויקט. עליכם לפרט באופן מלא ככל האפשר את מגוון הפיצ'רים שברצונכם שהמערכת תציע.*

*עליכם לתאר את הפונקציונליות שיש במערכת, מבחינת תהליכי העיבוד המתרחשים בה או מתוך הזווית של המשתמשים שהולכים להשתמש בה בסופו של דבר. במהלך מחנה הקיץ מילאתם שני קנבסים. זה המקום לפרט לעומק על כל ריבוע שמילאתם. יש להתמקד ב****מה*** *המערכת עושה, ולא* ***איך*** *היא תעשה זאת (לכך נגיע בשלבים הבאים).*

# הקנבס הרזה

# תיאור כללי

*דיקומפלייר אשר הופך משפת אסמבלי 32 ביט לפסדו סי*

# מטרת הפרויקט

*הצורך שהוא עונה הוא על לקחת קוד באסמבלי ובערך לראות איך הוא היה נראה בפסדו סי או למתכנת שכותב באסמבלי ורוצה להשוואות את הקוד שלו לסי*

*הוא מועיל להראות את הקוד בסי (שהוא בערך) ממה שהוא היה באסמבלי*

*לא היה לי רעיון טוב לפרויקט , אז יאן פורן הציע לי את הרעיון אחרי שדנתי איתו על נושאים לפרויקט*

# סוגי משתמשים / קהל היעד של היישום

*מתכנתים וחוקרי חולשות של סי*

*כן, בתכנות בסי ובתחנות אסמבלי*

*מתכנת שרוצה להבין את הקוד שלו בצורה יותר קריאה או חוקר חולשות בסי שיש לו רק קובץ באסמבלי*

# ידע, סיכונים ושאלות

*שלא אסיים בזמן*

*סיפיפי, אסמבל וmysql*

*לפרש את החלקים באסמבלי 32 ביט לפסדו סי בגלל החלקים השונים שיש שם*

*כן , אילו פיצ'רים עדיף להתמקד*

# פתרונות קיימים

*כן IDA חלקית , אני רוצה להוסיף פיצטר אבטחה על פונקציות מסוכנות בסי*

**מטרה**

מטרה

*אני מנסה להשיג עם הפרויקט הזה (בערך) המרה מאסמבלי לפסדו סי וללמוד איך זה בהשוואה לסי והפרויקט הזה הוא בדיוק הדרך הטובה לעשות את המשימה בגלל שהוא יודע לקחת קוד באסמבלי ולתרגם אותו לכמעט בלי טעויות מה שאדם רגיל לא יכול*

**תפקוד**

תפקוד

*הפרויקט הופך קוד מאסמבלי 32 ביט לפסדו סי ומציג את התוצאה ליוזר שרוצה את התוצאה*

קלט\פלט

*הקלט של הפרויקט(מהיוזר) הוא קובץ באסמבלי שאותו הוא מכניס לתוכנה (הדיקומפלייר) והפלט שהוא מוציא ממנו הוא הקובץ אחרי שהוא דיקומפלייר לפסדו סי*

ממשק משתמש

*המשתמש מתקשר עם הפרויקט דרך זה שהיוזר נכנס לדיקומפלייר ושם מכניס קובץ באסמבלי והוא מציג לו את הקובץ הדיקומפל למסך וככה התקשורת עובדת (דרך תוכנה שמקבלת קובץ ומוציאה אותו למשתמש)*

קוד

*אשתמש בסביבת עבודה windows ונכתוב את הפרויקט בc++ (כולל c++17) ,*

*Assembly 32/64 bit*

*וכ"שפה" pesudo c*

*כנראה שאעבוד בספיריות boost , parsers algoritem , map ועוד..(לא ידוע בניתיים מה עוד)*

אחסון

*כמות המידע שהפרויקט שלי צריך הוא לוגים קוד*

*ואשמור אותו בgitlab*

# צורה ונראות

צורה ונראות

*זה תוכנה שלוקחת קובץ מסוים ומראה אותו תיאורטית איך הוא היה נראה בשפת המקור שנכתב בו*

# דרכי שימוש

*כדי לבצע את המשימה, היוזר מכניס קובץ באסמבלי ומקבל את הקוד שלו בפסדו סי, במקרה והקוד לא נכון הדיקומפלייר מתריע זאת , הפרויקט מבצע את המשימות כאשר הוא עושה דיקומפלצזיה ויוזרים שונים ללא קשר אחד לשני יכולים לתקשר עם הפרויקט בזמנים שונים עם קבצים שונים*

*מה קורה במידה והרצף אינו נכון? הדיקומפלייר מתריע למשתמש*

סביבה

*וינדוס*

מבנה

*לפרויקט יהיה כמה מודולים אשר לכל אחד יש תפקיד שונה בדיקמופל, כלומר רכיב אחד שמתרגם ורכיב שני שמקבל את הקובץ לתרגם וככה הלאה. הם יהיו מחוברים בסדר היררכי , אחרי שרכיב אחד מסיים לעשות משהו בא תורו של הרכיב האחר*

*התרשים של הפרויקט כרגע הוא קבלת קובץ אסמבלי->קריאת הקובץ ->דיקמפול*

מודולים

*התרגום מאסמבלי לסי על ידי התוכנה,הצגת הoutput ליוזר*

*הוא יהיה בנוי אך ורק מתרגום וקריאה*

פיצ'רים

*מהם הפיצרים המרכזיים?*

1. *תרגום מאסמבלי לסי*
2. *הצעת קיצור של הקוד באסמבלי*
3. *הצעת תחלופות לקוד בסי*
4. *הצגת האוטפוט כמו IDA*
5. *שימוש באסמבלי 32 ביט*
6. *שימוש באסמבלי 64 ביט*

*7. דיקומפול של סיפיפי*

*בעיקרון התוכנה מתרגמת מאסמבלי לפסדו סי*

תהליכים

*התהליכים העיקריים שהתוכנה יודעת לעשות הוא לתרגם מאסמבלי לפסדו סי הפונקציונליות שלה תרגום משפת אסמבלי לפסדו סי*

*והמשתמש חווה את הדיקומפייל של אסמבלי מפסדו סי*

דרישות חומרה\תוכנה

*נדרש מעבד שיכול להכיל 32 או 64 ביט (כלומר x86 או x64 ) ווינדוס 10*

בדיקות

*הבדיקות הם כאלה: קודם אני אכניס קובץ באסמבלי (שאני יודעת איך הוא נראה בסי) ואבדוק את התוצאה על ידי השוואה בין דיקומפלייר קיים לתוצאה שייצאה לי , בדיקת שגיאות בקומפלצייה ובדיקות דיבוג בעזרת breakpoints*

*ובסוף הפיתוח אני אריץ כמה קודים באסמבלי ואראה אם זה נכון , השיטת דיבוג שלי יהיה step by step ואחרי שאוסיף עוד רכיבים אריץ ואבדוק.*