בס"ד

הצעת פרויקט WinSMARTS

ברוך בורשטיין ואלעד יחיאל, מנחה- ד"ר אריאל שטולמן

# מקום ביצוע הפרוייקט

מכון לב / בבית

# רקע

בקורס "הנדסת מערכות תגובתיות וזמן אמת" יש כלי שמיועד ללימוד פעולת task scheduler ודימוי אלגוריתמים שונים שלו ועוד כמה פעולות של מערכת הפעלה. לכלי הזה קוראים **SMARTS** (**Sma**ll **R**eal **T**ime **S**ystem for the real-time laboratory) ומטרתו היא ללמד על שיטות שונות לתזמון תהליכים מחזוריים סינכרוניים ומקביליים א-סינכרוניים, להראות את אופן הפעולה שלהם ובאיזה מצבים הם מתאימים יותר או פחות.

בכלי הנ"ל ישנם מספר בעיות, כאשר המרכזית היא שהוא יכול לרוץ רק בסביבת DOS. דבר זה מקשה על השימוש בו. צורת ההפעלה המומלצת היא לפתוח תוכנת DOSBOX (אמולטור של DOS), לקמפל בTurbo C++ בתוך הDOSBOX, לראות פלט, לערוך את הקבצים בעורך טקסט בWindows, וחוזר חלילה.

שימוש כזה מקשה מאוד על העבודה מהסיבות הבאות:

1. בעיה הכי גדולה - יש DEBUGGER פרימטיבי, חלק גדול מחיפוש השגיאות הוא בשיטת הניסוי וטעיה, שתילת פקודות פלט למסך, כאשר בכל ניסוי צריך שוב לקמפל ולהריץ בTurbo C++. כשהתוכנית נתקעת צריך לסגור את כל הDOSBOX ולהפעיל שוב.
2. הפיתוח נעשה בעורך טקסט ולא בסביבת פיתוח משולבת (IDE), מה שאומר שאין כלים שמקובלים היום בכל סביבת פיתוח, אינטגרציה עם פרויקטים אחרים, מעקב אחר מצב Registers וה Flags של המעבד, השלמות אוטומטיות ועוד.
3. לא ניתן לעשות Reuse לקוד.
4. מבנה המעבד באמולטור מיושן ושונה בהרבה מהמצוי בשוק, אין התאמה לארכיטקטורת x86-64.
5. הקימפול הוא מאוד לא נוח. חלק ממקשי הקיצור לא פועלים או גורמים לשגיאה הסוגרת את ה DosBox. צריך כל פעם לפתוח מחדש את הקבצים, לקמפל ולהריץ. לפעמים צריך להגדיר הגדרות מיושנות כמו far, tiny memory model
6. מכיוון שמקמפלים בקומפיילר ישן, אין אפשרות להשתמש בפונקציונליות "חדשה" של C++, כאשר "חדשה" כולל גם דברים ישנים דוגמת STL או exception.

# המטרה

אנחנו רוצים בשלב ראשון לשפר את הכלי שעומד לרשות המרצה והתלמידים בקורס הנ"ל ע"י מתן מענה לבעיות הנ"ל. הפתרון הוא בהכרח בכתיבת הכלי מחדש תוך התאמתו למערכות הפעלה מודרניות.

בשלב השני נרצה לחבר גם GUI, שיציג תוך כדי ריצה את הנתונים על התהליכים וה context switchs ע"מ להקל בדיבוג של הscheduler ולראות בצורה קלה את הפעולה שלו.

המטרה היא שהכלי יוכל להיות בשימוש לפחות על Windows במצב 32 ביט בשימוש בVisual Studio, כי זוהי הפלטפורמה שבה רוב התלמידים רגילים לפתח. נשתדל גם לאפשר פיתוח בפלטפורמות או קומפיילרים מודרניים אחרים.

# האמצעים

העברת הSMARTS למערכת הפעלה מודרנית בהכרח תיצור גם איזשהו אי-דיוק במימוש כי נצטרך לדמות מנגנונים שאין (או שאין גישה אליהם) במערכות מודרניות. אבל מכיוון שמדובר בכלי לימודי אנחנו מרגישים שהצורך בכלי שקל לעבוד אתו גובר לאין-ערוך על הצורך בסימולציה מדויקת של מערכת זמן-אמת.

נרצה להיצמד כמה שיותר לתוואי של הSMARTS המקורי, ע"מ שהקורס במתכונתו הנוכחית יוכל להשתמש בכלי שלנו. עם זאת, במקומות שנרגיש שניתן להשתמש בפרדיגמות תכנות מודרניות ע"מ לשפר את הקוד, נשנה לפי הצורך. גם זה בעצם מתוך מחשבה על הכלי ככלי שמשמש להוראה, בהבנה שקוד שכתוב טוב גם הוא מלמד משהו, גם אם זאת לא מטרת הקורס. כמובן שקוד טוב לפעמים קשה יותר להבנה או למעקב אחרי אופן הפעולה שלו, ומכיוון שמדובר בכלי לימודי, פעולה יעילה שלא מבצעת בפירוש כל שלב עלולה לפספס את המטרה. נצטרך למצוא את שביל הזהב בין תכנות יעיל ומודרני לבין שמירה על קלות ההבנה. נצטרך לזכור שהמטרה היא לא רק כלי סימולציה טוב, אלא כלי עזר ללימוד טוב.