**תרגיל 2 – מכונת אניגמה:**

**מגישים:**

שם: ניקול ספוז'ניקוב ת"ז: 212196372 מייל: [nikolsa@mta.ac.il](mailto:nikolsa@mta.ac.il)

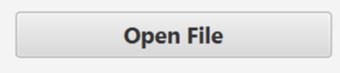
שם: ברק גולן ת"ז: 313170730 מייל: [barakgn@mta.ac.il](mailto:barakgn@mta.ac.il)

**הסבר על המערכת:**

הפרויקט מורכב מ-3 מודולים : מודול של ה-UI , מודול של ה-engine ומודל של ה-DTO.

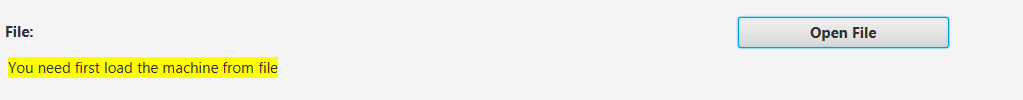
המודול הלוגי ששמו Engine מכיל את כל הלוגיקה מתרגיל 1 ובנוסף מכיל תייקיה ששמה decryptionManager ובה מחלקה הממשמשת את לוגיקת ה-brute force ומחלקות נוספות העוזרות למימוש ה-Brute force.

מודול ה-DTO מכיל את כל ה-DTO הנדרשים למערכת. בנוסף ל-DTO מתרגיל 1 נוספה תייקיה בשם dmDTO ובה DTO עבור ה-DM .

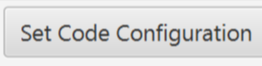


בהפעלת המערכת נפתח מסך ובו יש ללחוץ על כפתור ולבחור את קובץ ה-XML הרלוונטי עבור הגדרת המכונה. במידה והקובץ אינו תקין המשתמש יקבל הודעת שגיאה, אחרת יטען לו הקובץ בהצלחה.

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**TAB 1- Machine configuration:**

ברגע שהקובץ נטען בהצלחה, יופיעו פרטי המכונה תחת machine details

ותתאפשר בחירת קונפיגורציה התחלתית של המכונה, בצורה ידנית ע"י המשתמש ולחיצה על

או רנדומלית ע"י לחיצה על .

לאחר בחירת קונפיגורציה התחלתית, Tab2 ו- tab3 יהפכו ל-enable והמשתמש יוכל לעבור אליהם.

לחיצה על מאפסת את כל הבחירה של הקונפיגורציה.

**TAB2 - Encrypt/Decrypt:**

ב-TAB זה מתבצעת הצפנה ופענוח של מחרוזת שנבחרה ע"י המשתמש ומוצגת הסטטיסטיקה של ההצפענה\פענוח עבור כל קוד.

ב- ההצפנה מתבצעת ע"י לחיצה על process , המשתמש מקליד

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטיאת כל המחרוזת המעוניין להצפין ובלחיצה על הוא יקבל ב-output את המחרוזת המוצפנת.

ב- בעת כל הקלדת תו המשתמש רואה ישירות את התו המוצפן.

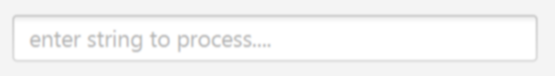
לחיצה על מודיעה כי המשמש סיים את המחרוזת אותה רצה להצפין, ומחרוזת זו נכנסת לסטטיסטיקה תחת הקוד הרלוונטי.

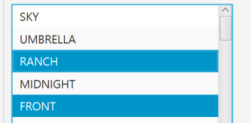
תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטילחיצה על מחזירה את הקוד הנוכחי למצב ההתחלתי.

**TAB3 – Automatic Encrypt/Decrypt:**

ב-TAB זה מתבצע הפענוח האוטומטי ע"י brute force.

בוחרים את המחרוזת לפענוח מהמילים במילון ע"י חיפוש מילה רצויה והקלדתה במקום המתאים

או ע"י לחיצה על המילים מהמילון בעזרת ctrl+click .

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

לחיצה על מאפסת את הקוד הנוכחי לקוד ההתחלתי.

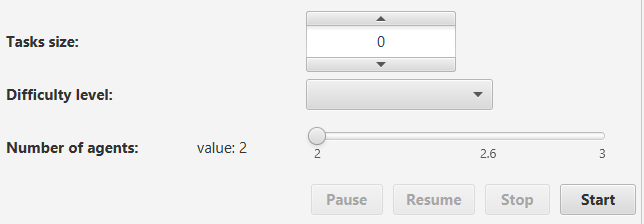


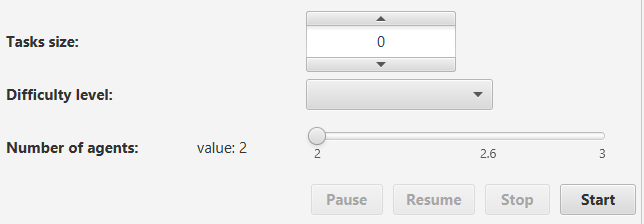
לחיצה על מצפינה את המחזורת שנבחרה.



לחיצה על מאפסת את המחרוזת שנבחרה והוצפנה ואת הקונפיגורציה שנבחרה עבור ה-brute force.

לאחר הצפנת המחרוזת יש לבחור את הקונפיגורציה של ה-brute force :

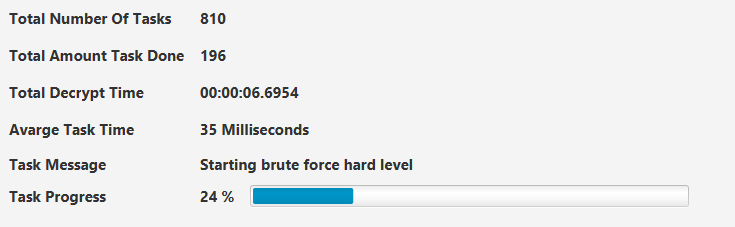


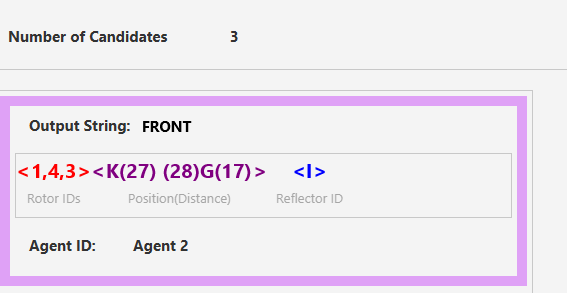


לאחר בחירת הקונפיגורציה לוחצים על והתהליך מתחיל.

במהלך התהליך יש אפשרות ללחוץ על **pause** שעוצר את התהליך, על **resume** שממשיך את התהליך מהנקודה בה נעצר ועל **stop** שעוצר את התהליך לגמרי.

במהלך התהליך ניתן לראות את התקדמות התהליך תוך עדכון נתונים אלו:



ובנוסף בעת כל מציאת מועמד אפשרי , נוסף מועמד אפשרי בצורה זו:

כך שמוגדר מי המועמד שנמצא, מה המחרוזת שנמצאה והקוד בה נמצאה המחרוזת האפשרית.

**מידע קצר על המחלקות החדשות:**

**UI:**

אוסף התיקיות המגדירות כל חלון ב-UI ובכל תייקיה כזו יש את קובץ ה-fxml , קובץ css ומחלקת ה-controller .

machineConfTab – מכיל את כל הרכיבים המגדירים את tab 1 .

filePathComponent – מכיל את הרכיבים המגדירים טעינת קובץ.

EncryptTab – מכיל את כל הרכיבים של טאב 2.

DMtab- מכיל את כל הרכיבים של טאב3.

**DTO:**

BruteForceLevel – זה ה-enum שמחזיק את הרמות השונות ב-brute force.

CandidateDTO – מכיל מחרוזת וקוד בו נמצאה המחרוזת. נועד עבור מציאת מועמד אפשרי.

TaskFinishDataDTO - מחזיק את כל הנתונים שיש להעביר ל-UI בעת סיום משימה. הוא מחזיק את רשימת המועמדים האפשריים שנמצאו במהלך המשימה, את הסוכן שמצא אותם ואת הזמן שלקח לסוכן למצוא כל מחרוזת אפשרית.

**Engine:**

decryptionManager – מחלקה זו אחראית על תהליך הפענוח האוטומטי ע"י brute force. במחלקה זו מחשבים את כמות המשימות שיש לבצע כפונקציה של רמת הקושי שהמשתמש בחר ומכניסים את המשימות לthread pool בכדי שיתבצעו.

בנוסף ישנן מחלקות עזר נוספות ב-engine שתפקידן לנהל את הthreads , את המשימות, להעביר את הנתונים לאחר סיום משימה וכו.

**בונוסים:**

* מימשנו את בונוס מספר 3 – בחירת מילים ישירות מהמילון.

המשתמש יכול לבחור מילה אחת ע"י לחיצה על המילה במילון או לבחור מספר מילים מהמילון ע"י ctrl+לחיצה על המילים הרצויות , והמילים שנבחרו יכתבו ישירות לתיבת הטקסט של קליטת המחרוזת.