יומן

## לינקים:

* [מאגר קוד](https://github.com/turner11/BWA-Final_Project)
* [משימות פתוחות](https://github.com/turner11/BWA-Final_Project/issues?q=is%3Aopen+is%3Aissue).
* [משימות סגורות](https://github.com/turner11/BWA-Final_Project/issues?q=is%3Aissue+is%3Aclosed).

30/10

סיכום עד כה:

* 3 פגישות עם המנחה בהם עברנו :
  1. ‏‎תאוריה ותיאור הצורך בפרויקט.
  2. פגישת תכנון ומשימות.
  3. פגישת עדכון התקדמות
* מבחינת עבודה:

1. הוגש [מסמך התנעה](http://projects.jce.ac.il/moodle/mod/assign/view.php?id=100)
2. הותקן VM עם לינוקס + eclipse
3. הורד [הקוד של BWA](http://sourceforge.net/projects/bio-bwa/?source=navbar).
4. נפתח [מאגר קוד בgithub](https://github.com/turner11/BWA)
5. ישנו גם [readme](https://github.com/turner11/BWA/blob/master/README.md) (הערה:זה read me של הפרויקט המקורי, לא שלי...)
6. הורדת קבצי גנום אנושי מלא.
7. תחילת נסיון להכנס לקוד התכנה - כרגע אין עדיין תובנות.

בשלב זה אנו מתמקדים בנסיון להבין קצת את הקוד ולאתר את צווארי הבקבוק שבו ע"מ שנוכל לייעל אותם

1/11

חיפוש קצר העלה שיש את ה[פרויקט במעטפת של DotNet](https://github.com/evolvedmicrobe/BWA-Sharp" \t "_blank). אם זה יעבוד, אני חושב שזה יזרז מאוד את קצב הפיתוח ויקל על הבנת הפרויקט.

מחקר קצר + ניסוי וטעי ה העל ה את הממצאים הבאים:

1. יש צורך להתקין את [Microsoft Biology Foundation](http://research.microsoft.com/en-us/downloads/5ced2711-313a-4dcb-84aa-adb87c9ea0a9/) כדי לקבל את הקובץ Bio.Dll ולהוסיף אותו כרפרנס.
2. מיפוי הגנום (רפרנס) לא מתוכנן כרגע לעבוד על windows (זו לא בעיה מפני שזה תהליך חד פעמי...).
3. [הקוד של הalign](https://github.com/evolvedmicrobe/BWA-Sharp/blob/master/Bio.BWA/Bio.BWA/BWA.cs) עצמו מאוד קצר, אבל יצריך הכרה והבנה של אוביקטים שמשתמשים בהם (בקישור זה מופיע בפונקציה SAMAlignedSequence ).
4. אני מקווה שהAlign עצמו יעבוד על windows.
5. הקוד של הפרויקט BWA המקורי ב++C לא מתועד וקצת קשה להכנס אליו ללא ידע מעמיק באלגוריתם.
6. [מצאתי הערה](https://bow.codeplex.com/discussions/395108) בנוגע לביצועים של האלגוריתם במחלקות Managed. לא ברור לי אם מדובר במיפוי בלבד או גם בהתאמה. על פניו נראה לי שמדובר במיפוי של הרפרנס בגלל גישות מרובות לדיסק הקשיח אם זה לא המצב, אני חושב שזה לא קריטי בגלל שאנו מתכננים בכל מקרה למקבל את החלק הזה (ייתכן וSSD יעזור גם כן...)

2/11

סיכום פגישת מנחה:

1. בחלוקת העבודה בפרויקט אני אתמקד בהתאמה של מחרוזות דומות.
2. עד הפגישה הבאה בעוד שבועיים (אחרי המילואים) נתקדם ביישום הפונקציה של התאמה של מחרוזות דומות. לשם כך:

* לקרוא במסמך שד"ר חסין שלח עמ' 1755: **Exact matching: backward search**
* לישם, בינתים בC# את הפונקציה Inexact search (עמ' 1756).
* להוסיף לפונקציה Unit test

18/11

 1. חזרתי מהמילואים, מאז עבדתי על מימוש **InexactSearch**. יש כרגע מימוש לאב טיפוס כולל ממשק גרפי שיעזור לדבג תוצאות (בDotNet). האב טיפוס עוד לא מחזיר תוצאות נכונות, אבל זה כבר קרוב לכך.

 2. השלב הבא יהיה לעשות את האינדוקס באב הטיפוס לפני ההרצה כך שנוכל לבדוק עד כמה המקבול עוזר בתנאים קרובים ככל האפשר לתוצר הסופי.

 3. קיים עדיין פער בהבנת האלגוריתם שאצטרך לגשר עליו בהמשך.

 4. בתיאום עם ד"ר חסין דחינו את הפגישה השבוע מיום ראשון לרביעי כדי שאוכל להספיק קצת עבודה לפני הפגישה כדי שהיא תהיה משמעותית.

 5. אני מניח שבשבוע הקרוב אעבוד קודם כל על טפסי התנעת הפרויקט שיש להגיש בקרוב ורק לאחר מכן אתפנה להמשך עבודה על האלגוריתם.

 6. את ההתקדמות ניתן לראות [בדף הcommits](https://github.com/turner11/BWT/commits/master/Code/BWT.Net)

19/11

את סיכום הפגישה היום ניתן לראות בצורת [issues ב github](https://github.com/turner11/BWA-Final_Project/issues?q=is%3Aissue+created%3A2014-11-19).

 28/11

התקדמתי עם הפרוטוטייפ, ונראה שהתוצאות מבטיחות...

עבור הקלט הבא:

* רפרנס באורך 2000 (זה לא אמור לשנות לזמן ריצה שתלוי בw...)
* 25 קריאות.
* כל קריאה באורך 35 (כמו שמבצעים בהדס),
* אפשור ל 2 טעויות.
* סיכוי של 3% לטעות בכל תו בריאה.

קיבלתי את התוצאות הבאות:

1. עבור אלגוריתם לא ממוקבל, זמן ריצה:00:01:17.8920683
2. עבור אלגוריתם ממוקבל (ההרצה על הקריאות ממוקבלת), זמן ריצה:00:00:23.1024634

כלומר שיפור של פי 3.39~. בעזרת המחשב הבא:

* זכרון:
  + 8GB DDR3 SDRAM
* מעבד:
  + 2nd Gen Intel(r) Core™ i7-2670QM
  + 2.2GHz processor speed
  + Turbo Boost up to 3.1GHz
  + 8-way processing

5/12

התחלתי לנתח את הקוד שמסביב לאלגוריתם של BWA. ממצאים:

1. מספר הטעויות המותר נקבע על פי אורך הדגימות.
2. בפורומים מומלץ לא לשנות ערך זה אלא להשתמש בחישוב ברירת המחדל.
3. אנו נשתמש בפקודה [bwa aln](https://github.com/turner11/BWA-Final_Project/raw/master/Documents/BWA%20aln%20MAN.docx).
4. הטעויות המותרות מחלקות את הדגימות ל2 חלקים, למיטב הבנתי:

* החלק הראשון שבו מאפשרים פחות טעויות. מספר הדגימות ומספר הטעויות המותרות נקבעות על ידי הארגומנטים seedLen ו maxSeedDiff בהתאמה.
* החלק השני. מספר הדגימות ומספר הטעויות המותרות נקבעות על ידי הארגומנטים: המשלים ל seedLen ו maxDiff בהתאמה.

9/12

בפגישה סיכמנו שנתחיל על מקבול נאיבי של האלגוריתם - תהליכון עבור כל קריאה.

בכל מקרה לשם כך נהיה צריכים לברר:

1. מה נמצא בזכרון כשמריצים את האלגוריתם.
2. האם האלגורית הנוכחי רקורסיבי או איטרטיבי.

12/12

לאחר בדיקה של הנושאים שהועלו בפגישה, התוצאות נרשמו בתור issues במאגר הקוד.

* [מה נמצא בזכרון כשמריצים את האלגוריתם.](https://github.com/turner11/BWA-Final_Project/issues/7)
* [האם האלגורית הנוכחי רקורסיבי או איטרטיבי.](https://github.com/turner11/BWA-Final_Project/issues/8)

19/12

הושלמה הורדת הגנום + אינדוקס שלו.

כרגע אני ממשיך לנסות לראות מימדים של האוביקטים המוחזקים בזכרון.

2/1

בפגישה האחרונה עברנו על קודה בקווים כללים. וסיכמנו שהשלב הבא יהיה כניסה מעמיקה יותר לקוד ונסיון להבין אותו.

מסתבר שזו לא משימה כל כך פשוטה....

בינתיים, קיבלנו מהמשתמש בהדסה את קובץ הקריאות (9.5 GB), ויכולנו להריץ הרצה משמעותית של התכנה. מסקנות:

1. אורך קריאה הוא 45 ולא 35 כפי שנאמר לנו.
2. הקוד לא מתועד ולא מדבר בעד עצמו. הבנה שלו תהיה משימה מורכבת. ייתכן והמיקבול הנאיבי לא ידרוש הבנה מעמיקה.

פתחתי דף עם תיאור ההתקדמות בהבנה + [תיעוד](http://projects.jce.ac.il/moodle/mod/assign/view.php?id=160) של אובייקטים מרכזיים שחשובים להבנת התהליך. הדף זמין [פה](https://github.com/turner11/BWA-Final_Project/wiki/CodeAnalysis)

9/1

בפגישה האחרונה עברנו יחד על הקוד. המסקנות:

1. הקוד מסובך להבנה בצורה יוצאת דופן. הוא עבר כנראה הרבה יעולים של השפה + יעולים של האלגוריתם ללא שום [תיעוד](http://projects.jce.ac.il/moodle/mod/assign/view.php?id=160).
2. הקוד מבצע רקורסיה ע"י ניהול מחסנית משל עצמו.

כתוצאה, התקבלו ההחלטות:

1. אנו נתחיל במקבול של הפונקציה שעושה את העבודה המשמעותית, מבלי להבין אותה לעומק (לפחות בשלב ראשון).

בנוסף, פתחנו את המשימות הבאות בגיטהאב: [[1](https://github.com/turner11/BWA-Final_Project/issues/11" \t "_blank)] [[2](https://github.com/turner11/BWA-Final_Project/issues/10" \t "_blank)] [[3](https://github.com/turner11/BWA-Final_Project/issues/9" \t "_blank)]

17/1

בשבוע החולף:

 1. עבודה על מסמך לקראת הגשת אבטיפוס.

 2. הפיכת האבטיפוס לידידותי יותר למשתמש (הצגת התקדמות יותר ברורה ובלי עבודה על התהליכון של ה GUI)

20/1

לאחר הפגישה היום סיכמנו על שיפורים במסמך האב טיפוס, ודיברנו על ההצגה והדגשים במבחן האמצע, וכיצד לחלק את הזמן עקב המילואים.

המסמך זמין [במאגר הקוד](https://github.com/turner11/BWA-Final_Project/blob/master/Documents/2%20Prototype.docx?raw=true)(לאחר תיקון).

23/1

המשך עבודה על מסמך + מצגת לבוחן אמצע. יצאתי השבוע למילואים...

24/1

המשך עבודה על מסמך לבוחן אמצע והגשה.

1/2

מאז החזרה ממילואים - עבודה על מצגת ומסמך לבוחן האמצע.

2/2

הצגת פרויקט גמר.

6/2

בוצע מעקב אחר הריצה של  הקוד המקורי של BWA, בנסיון לפענח את הפלט של הפונקציה שתמוקבל.

ממצאים:

* הפלט נשמר ב: bwt\_seq\_t->n\_aln
* הפלט נשמר כאינדקס במערך סיפות (זה היה צפוי).
* הערך נשמר בצורה בינארית (צריך עדיין להבין את הקידוד).
* ישנם שדות מעניינים נוספים עם מידע שכנראה רלוונטי.
* המחלקה המלאה זמינה [פה](https://github.com/turner11/BWA-Final_Project/blob/master/Documents/bwa_seq_t_Class.txt).

13/3

במהלך חופשת הסמסטר, העבודה התמקדה בסריאליזציה של האובייקטים הרלווטים לתהליך המיקבול למערף של בייטים וחזרה לאובייקט מהבייטים.

האתגר היה שההאובייקטים הרלוונטים מכילים מצביעים לאובייקטים, שצריכים לעבור את אותו התהליך.

פעולה זו תאפשר העתקרה של האוביקטים הנ"ל לזכרון של הGPU לצורך המקבול.

הקוד של הסריזליזציה זמין [פה](https://github.com/turner11/BWA-Final_Project/tree/master/Code/Struct_to_and_from_bytes/testStructConversions/testStructConversions).

29/4

עדכון עבודה מהחודש האחרון:

במהלך החודש האחרון (שכלל גם את פסח) התבצע מאמץ לבצע את המקבול עצמו ע"ג כרטיס מסך.

נסיון זה לא צלח. לאחר בדיקה, מתברר שבפרויקט Barracuda, שמקבל את אותו הקוד, עובדים כבר 6 שנים למעלה מ14 מפתחים בעלי ידע מקדים מהתחום - והפרויקט טרם הסתיים. בנוסף - כדי לגרום לפרויקט שלהם לעבוד - הם ביצעו שינווים נרחבים בקוד - שכאמור לא מספיק ברור לנו. במבט לאחור נראה שזוהי דרך ללא מוצא במסגרת הזמן וכוח האדם המוקצים לפרויקט.

בזמנו בחרנו שלא להסתכל על פרויקט זה ע"מ לא להתקבע לצורת מחשבה כלשהי - בדיעבד נראה שזו היתה טעות, ויכולנו לעמוד על כך זה לא היה הכיוון הנכון להתקדם בו עוד בשלב מוקדם יותר של הפרויקט.

החלטה להמשך:

אנו נתקדם בפרויקט ונייעל את משך החיפוש ע"י שימוש בפרויקט Barracuda, תוך הוספת ייעול משלנו - מיון קריאות הDNA בצורה ממוקבלת כך שימנע ה Lock-step במהלך ביצוע האלגוריתם ע"ג כרטיס המסך.

הקוד שלהם זמין לנו...

15/5

קיבלנו מחשב עם כרטיס מסך בשביל הרצת הקוד של פרויקט barracuda.

29/5

על המחשב קיימת מערכת הפעלה windows 7 בעוד הפרויקט דורש linux.

במהלך השבועיים האחרונים התקנתי מספר Virtual Machines בנסיון להריץ את הפרויקט, אך בכולם התקנת הדרייבר של כרטיס המסך נכשל.

2/6

לאחר התייעצות עם אנשי מקצוע מתחום התכנות המקבילי, מתברר שלא ניתן לבצע זאת על גבי virtual Machine.

פנינו למחלקת המחשוב בבקשה להתקין את מערכת ההפעלה הנדרשת.

19/6

סיכום השבועות שעברו:

המחשוב לא הצליחו להתקין את UBUNTU הנדרש להרצת החומרה, ובמקום זאת התקינו Fedora, בגרסה לא נתמכת.

בגלל קוצר הזמן, בקשנו מהם להתקין גרסה נתמכת ובמקביל התחלנו להתקדן עם הפרויקט בכיוון של מקבול על CPU (כיוון שממילא התקדמנו בו, במיוחד עד לשלב האב טיפוס).

העבודה שהתבצעה מאז העדכון האחרון:

 1. השגנו דגימות מרובות שנוכל להפעיל עליהן את האלגוריתם.

 2. מיינו את הדגימות (19,000,000)

בשלב הבא נשווה את זמני הריצה של של:

1. האלגוריתם הלא ממוקבל.

2. האלגוריתם הממוקבל.

3. האלגוריתם הממוקבל הרץ על הדגימות הממוינות.

3/7

ישנן תוצאות מבטיחות לאלגוריתם הממוקבל (יציבות ללא תלות במס' / אורך הדגימות).

נראה שהמיון עוזר, אך רק במעט לזמן הריצה.

אנו חושבים שהמיון יהיה משמעותי הרבה יותר במיון ע"ג כרטיס מסך מפני שהוא ימנע חלק ניכר מה Lock-Step .

בינתיים עדכנתי את התכנה שמראה תוצאוץ – כולל גרפים להשוואת זמני ריצה + הוספת כל הטסטים.

את התוצר של השבועות האחרונים ניתן לראות [במאגר הקוד](https://github.com/turner11/BWA-Final_Project/commits/master)