תרגיל 3 – עיבוד מידע תלת מימדי בביולוגיה מבנית

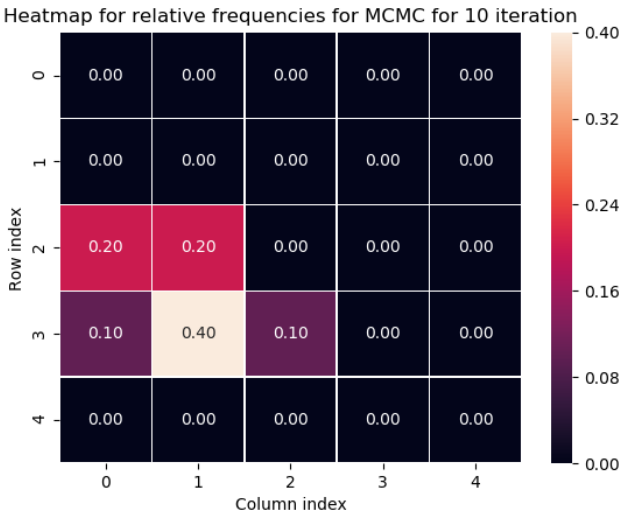
דרור בר (203523352)

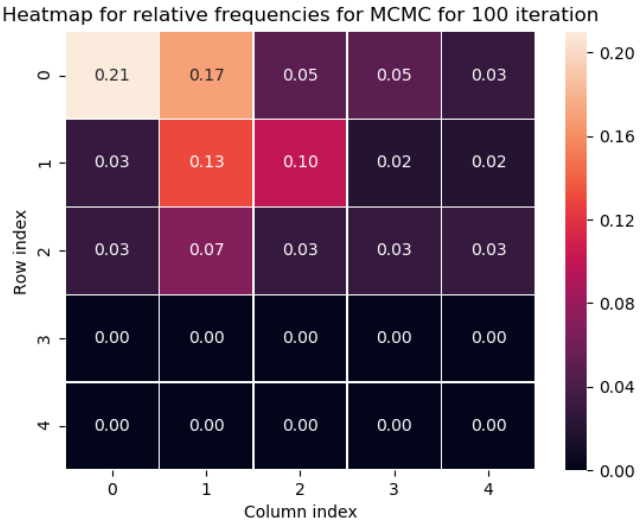
לירון גרשוני (308350503)

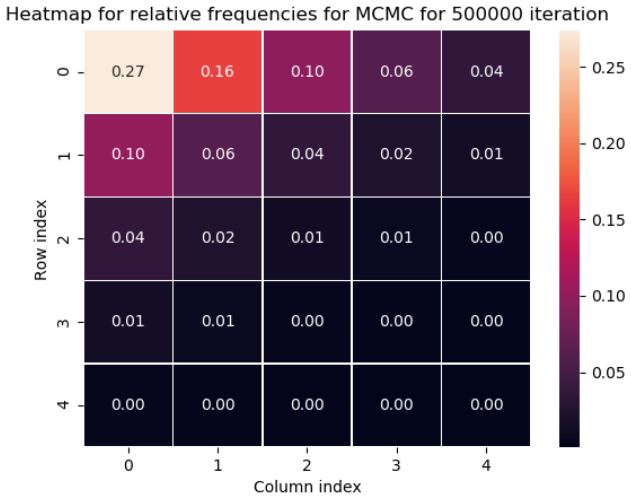
חלק ב'

שאלה 4

מצורפים מפות החום של הריצות השונות



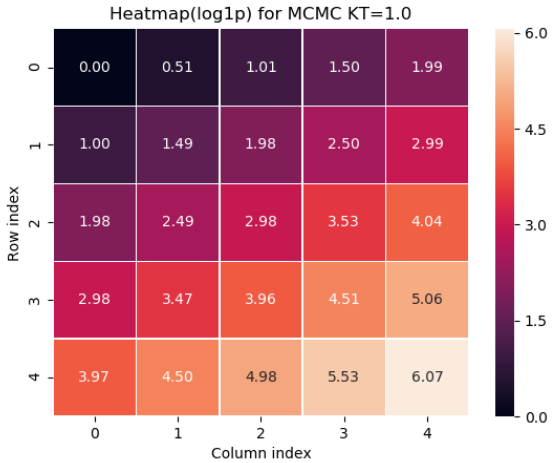




שאלה 5

שני שינויים בולטים שרואים : ככל שמספר הריצות עולה יש נטייה חזקה יותר להתקרב לתא (0,0), התא בעל האנרגיה הנמוכה ביותר. כאשר מספר הריצות קטן (10) אנחנו עדיין קרובים למקום ההתחלה, כאשר אנחנו ב100 ריצות אנחנו מתפזרים בדרך ל0,0 ורק כאשר מגיעים ל500000 רוב הזמן היה סביב התא הנ״ל. בנוסף אפשר לראות שב10 איטרציות לא היה מספיק זמן לקבל הסתברויות רבות, ב500,000 הייתה התכנסות כבר ולכן גם קיבלנו מעט הסתברויות וב100 שאנחנו עדיין בתהליך ההתכנסות יש פיזור רחב יותר של הסתברויות.

שאלה 6



שאלה 7

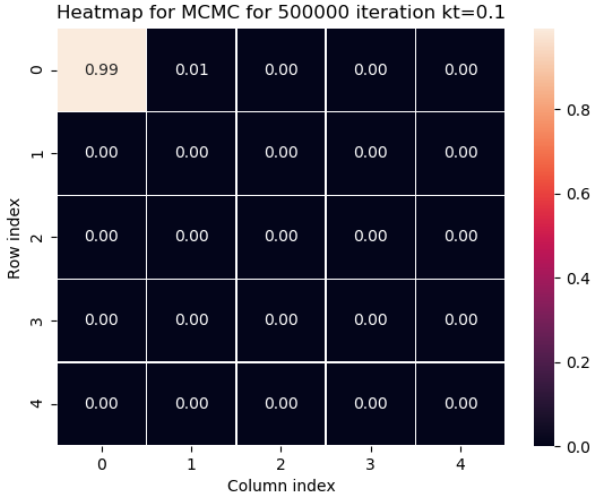
ההפרש בין שורות ועמודות צמודות הוא 0.5. כל ערך במפת חום מסמל את ההפרש בין לוג ההסתברות המקסימלית לבין לוג ההסתברות להיות באותו תא. לדוגמה עבורה 0,0 ההפרש הוא 0 כי זה התא עם ההסתברות המקסימלית. ועבור 4,4 ההפרש הוא הכי גדול כי ההסתברות להיות ב4,4 הוא המינימלי. לדעתי ההפרש שאנחנו מקבלים נובע מהתפלגות אחידה על מעבר ב grid.

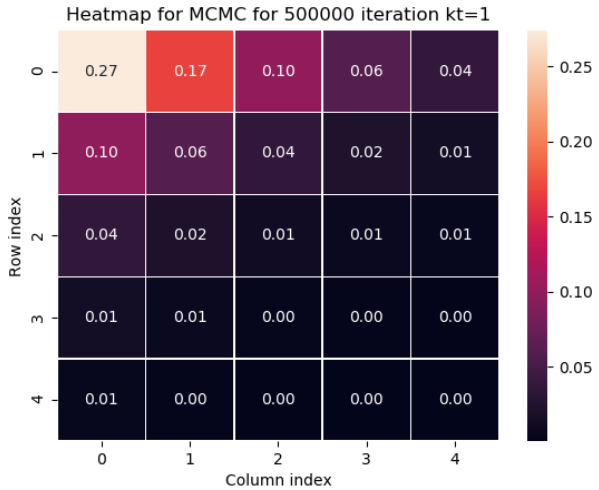
שאלה 8

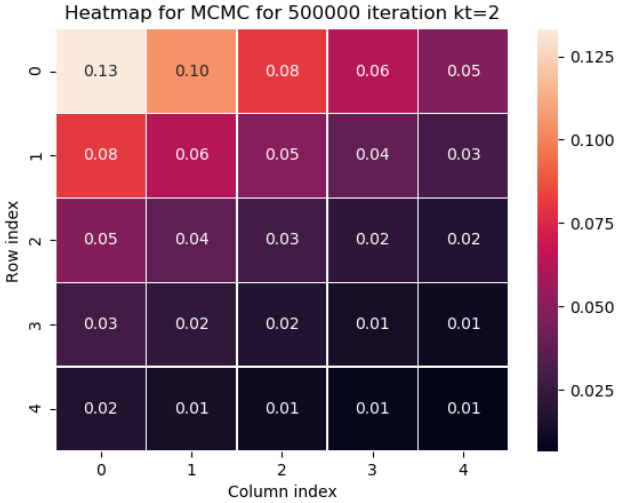
נהוג להשתמש בפונקציה numpy.log1p במקרים כאלו כי ההסתברויות והערכים שמקבלים הם נמוכים נורא (קרובים ל0) והפעלת לוג ישירות תביא מספרים שליליים גבוהים (קרוב למינוס אינסוף), שימוש בפונקציה log(1+x) מאפשר הקטנה של הערכים האלו ושמירה על סדר גודל נוח לעבודה וקונסיסטנטי – שומר על ההפרשים בין ההפרשים.

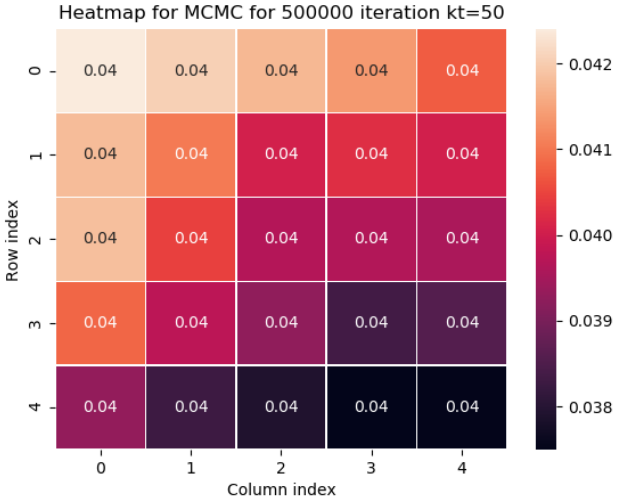
שאלה 9

התמונות מצורפות ובסופם הסבר





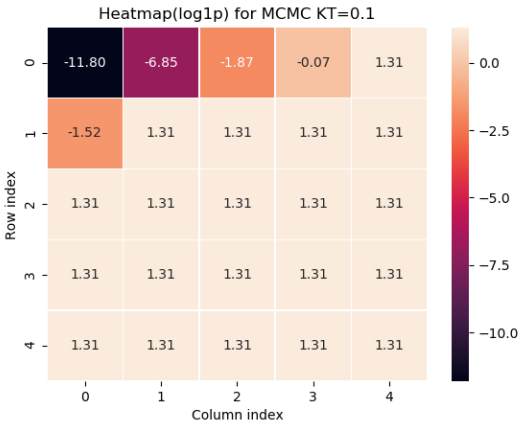


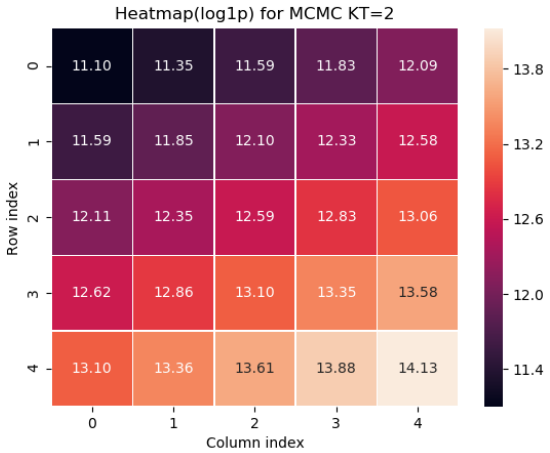


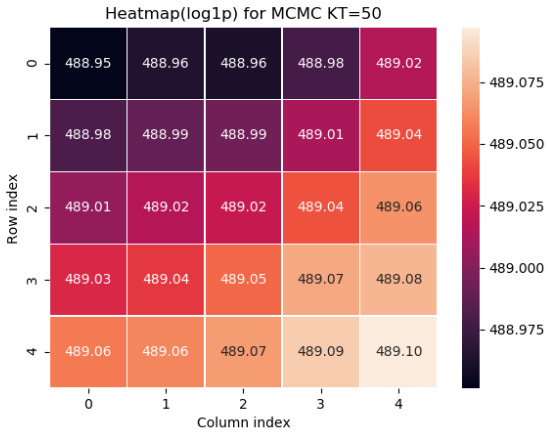
ביחס לערך הדיפולטי (1) כאשר אנחנו מקבלים $kT=0.1$, כלומר נמוך יותר אין הרבה אנרגיה במערכת (טמפרטורה) ולכן אנחנו מקבלים מפת חום ללא פיזור רחב, כלומר הרוב מתרכז סביב המינימום האנרגטי (0,0) ולא מצליח לברוח משם. לעומת זאת כאשר ה kT גבוה יש פיזור הרבה יותר רחב כי יש יותר אנרגיה במערכת, כלומר יש יותר יכולת "בריחה" למקומות עם אנרגיה גבוהה יותר למרות שעדיין יהיה נטייה לשורות ועמודות עם אנרגיה נמוכה יחסית מאחרים – ככל שכמות האנרגיה גדלה הנטייה הזאת נהיית עדינה יותר ויותר.

שאלה 10

מצורפות התמונות







שאלה 11

הכפלנו ב K_BT כדי לתת דגש להבדלים הקטנים בין תאים שונים. הבדלים גדולים בין התאים לא משתנים כמעט למרות ההכפלה (בkT קטן) בעוד הבדלים הקטנים מקבלים מתיחה משמעותית שמאפשרת לראות את ההבדל. לדוגמה אפשר לראות שרק במתיחה המסיבית (kT=50) רואים ש0,0 מקבל את הערך הנמוך ביותר.

שאלה 12

הדבר המוזר בנוגע להפרשים הוא שהם לא אחידים, יש הבדל משמעותי מאוד לשורה ולעמודה ואי אפשר להגיד שכל שורה היא החסרה של ערך מהשורה הקודמת (מה שהיינו יכולים להגיד בשאלה 7). יש מספר הסברים למה קרה: הרנדומיות של האלגוריתם יכולה ליצור מקרים מיוחדים ונדירים שאנחנו נתקעים בהם מה שמשפיע על הפיזור וההסתברות, בנוסף חוסר האנרגיה של המערכת מקשה על יציאה ממקומות ונותן דגש יתר לאירועים לא צפויים.

שאלה 13

ניזכר כי פונקציית האנרגיה שלנו היא $E(x,y) = 1 \cdot x + \frac{1}{2} \cdot y$ ולכן בשביל לקבל את וקטור הכוח נבצע גזירה חלקית ונקבל כי הוקטור הוא $f(x,y) = \left(-1, -\frac{1}{2}\right)$

שאלה 14

מצורף הפסודו קוד

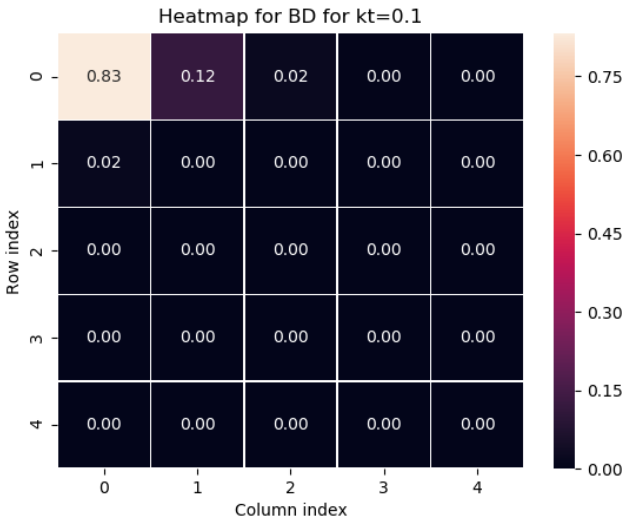
```
MIN_POS = -0.5
MAX_POS = 4.5
FORCE_VECTOR(0.5-,1-) =
DIFFUSION_COEFFICENT = D

def BD_psuedocode(num_of_steps, delta_t, kbt):
    start_configuration = Vector(random(MIN_POS, MAX_POS),  random(MIN_POS, MAX_POS))
    current = start_configuration
    movement_list = []
    movement_list.append(current)

    for i in range(num_of_steps):
        current += delta_t / kbt * DIFFUSION_COEFFICENT * FORCE_VECTOR+
        math.sqrt(6*DIFFUSION_COEFFICENT * delta_t) * random2DVector.Norm(0,1)
        current[0] = max(MIN_POS, current[0])
        current[0] = min(MAX_POS, current[0])

        current[1] = max(MIN_POS, current[1])
        current[1] = min(MAX_POS, current[1])
        movement_list.append(current)

    return movement_list
```

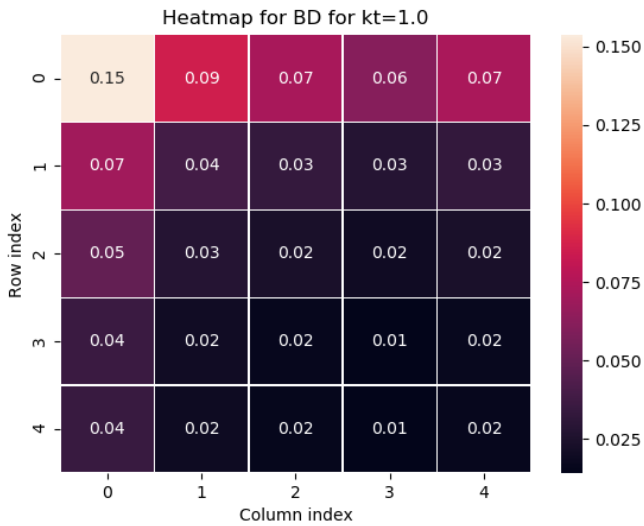
שאלה 15

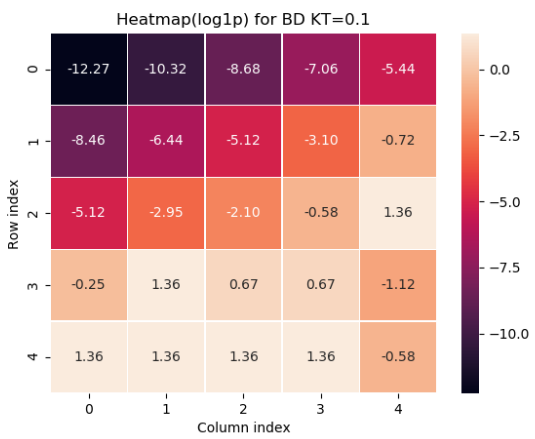
אם הK_bT שואף לאפס והוא פרופורציונלי לD נקבל כי החלק האמצעי במשוואה הוא רק $\Delta tf(x)$ והחלק הימני במשוואה (הרנדומי) מתאפס כיD גם שואף לאפס. אנחנו למעשה מקבלים שהמקום החדש הוא המקום הישן + הגרדיאנט * כמה זמן עבר. למעשה מדובר על אלגוריתם מבוסס גרדיאנט שהוא *gradient descent* .

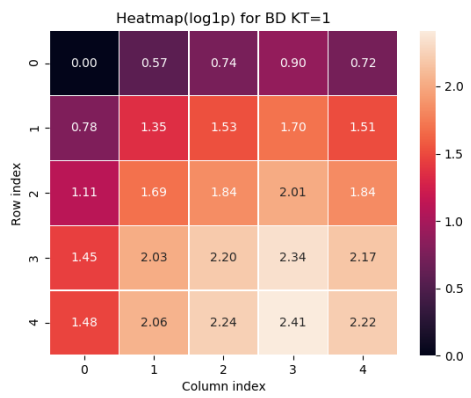
שאלה 16

מצורף בקובץ bd_algorithm.p









התוצאות דומות חלקית לתוצאות ממקודם. כן רואים נטייה לאזורים היותר נמוכים אנרגטית (שורה ועמודה 0 ותא 0,0) אבל כן יש יותר פיזור יחסית לכמות הריצות שהיו. כנראה בגלל החלק הרנדומי של המשוואה והתוספת של מספר האיטרציות. אפשר לראות שכאשר ה ktנמוך אין יכולת לצאת מהתא בעל האנרגיה הנמוכה ביותר.

חלק ג'

שאלה 17

כמובן

שאלה 18

מרחב קונפיגורציות הוא מרחב וקטורי בו כל קונפיגורציה ממופה למערכת כלשהי. בעולם של חלבונים יש שתי מרחבים משמעותיים – מרחב קרטזי (מכיל את הקורדינטות התלת מימדיות), מרחב אחר הוא הspace internal coordinates (שזה הוקטורים והזוויות שלהם).

שאלה 19

- Ab initio folding
  - קלט: הרצף של החלבון ופונקציית score של האנרגיה של הקונפיגורציה
  - פלט: המבנה המקופל של החלבון – קורדינטות במרחב.
- Comparative modeling
  - קלט: הרצף של החלבון ו"ספרייה" של חלבונים נוספים עם רצף דומה ומבנה מרחבי שאליהם נשווה.
  - פלט: המבנה המקופל של החלבון – קורדינטות במרחב.
- Protein-protein docking
  - קלט: 2 מולקולות (או יותר) במצב הטבעי שלהם, כלומר הקורדינטות שלהם במרחב
  - פלט: קורדינטות של שני החלבונים כשהם אחד ליד השני (מחוברים), כלומר האלגוריתם מבצע שינוי קורדינטות לאחד החלבונים.
- Flexible peptide docking – refinement
  - קלט: מבנה הפפטיד ומבנה הרצפטור ופונקציית אנרגיה
  - פלט: קורדינטות של שני החלבונים כשהם אחד ליד השני (מחוברים), ייתכנו מספר פלטים שידורגו ע"י פונקציית הscore.
- Flexible peptide docking – ab initio
  - קלט: מבנה הרצפטור + רצף הפפטיד + קורדינטות מקורבות לאתר הקישור
  - פלט: קורדינטות של שני החלבונים כשהם אחד ליד השני (מחוברים), ייתכנו מספר פלטים שידורגו ע"י פונקציית הscore.
- Flexible peptide docking – blind
  - קלט: מבנה הרצפטור + רצף הפפטיד
  - פלט: קורדינטות של שני החלבונים כשהם אחד ליד השני (מחוברים), ייתכנו מספר פלטים שידורגו ע"י פונקציית הscore.
- Molecular dynamics
  - קלט: החלקים השונים במערכת, האינטרקציות ביניהם (מחושבים ע"י פונקציות אנרגיה), והדינמיקה ביניהם (על פי חוקי התנועה של ניוטון או מכניקת הקוונטים)
  - פלט: סט קורדינטות של כל החלקים לאורך הריצה (או בסופה) או סימולציה ויזואלית של התנועה שהתבצעה.
- Motion planning
  - קלט: נקודת התחלה (source), נקודת סוף (goal) ופונקציית אנרגיה כלשהי שתיצור גבולות ואזורים שאי אפשר לעבור דרכם

- פלט: כל המסלולים בין הsource ל goal.

שאלה 20

- Dock – בהינתן מבנה של חלבון נצבע את המשטח שלו. נחפש את הכיסים (איפה שאפשר לשים את המולקולה השנייה). נבצע אלגוריתם התאמה ולאחר מכן בדיקה שאין התנגשויות + בדיקת אנרגיה. נזרוק כדורים על המשטח ונחפש איפה יש cluster כי הוא כנראה אתר הקישור.
- FFT – משתמשים בטרנספורם פוריה בשביל לבצע חיפוש brute force על כל הצורות במרחב. לאחר שנמצא פתרון נעשה לו עידון. בכללי ממפים את החלבון לgrid שאחד מהם לא זז (R) והשני כן (L) באמצעות פעולות מתמטיות על החפיפה אפשר לבצע translation של כל האופציות.
- Ab initio – בהיתן חלבונים קטנים ניתן להכניס למחשב את כל הכוחות הפיזקליים והדנימיקה בין החלבונים ולתת לסימולציה לרוץ. מסוגל לתת תוצאות יחסית מדויקות לחלבונים קטנים.

שאלה 21

שיטות מבוססות גרדיאנט מיועדות למציאת מינימום אנרגטי מקומי. החסרונות/מגבלות שלהם לעומת שיטות אופטימיזציה אחרות שהם נתקעות במינימום מקומי ולאו דווקא ימצאו את המינימום הגלובלי ללא שימוש בrandom walk, אתחול בהרבה מקומות שונים או נתינת אופציה לקפיצה אנרגטית (כמו שעשינו במונטה קרלו).

שאלה 22

החלק הרנדומי נועד לפשט את המשוואות והסימולציות ע"י "ויתור" של החישובים לכל מולקולות המים. בצורה כללית המים נמצאים בכל המערכת ומתנגשים בכל הכיוונים במולקולות שאנחנו מסתכלים עליהם, אפשר לחשוב עליהם כאוסף של מ"מ שדוחפים לכל אחד מהכיוונים. על פי חוק המספרים הגדולים אפשר לסכום אותם למשתנה רנדומי אחד שמתפלג נורמלי – זה משמאות החלק הזה.

שאלה 23

האלגוריתם ששומר זיכרון הוא motion planning. המבנה נתונים שמשמש להחזקת המפה הוא עץ (כל האלגוריתם מתבסס על RRT- random tree)