**שאלה 1 סעיף א'**

תרגיל 1.2-2

כדי שמיון-הכנסה ירוץ מהר יותר מהר ממיון-מיזוג נפתור את האי-שיוויון הבא:

הפונקציות עולות ממש ולא חסומות לכן נחפש עבורו מתקיים (1) וגם עבור מתקיים .

עבור :

עבור :

לכן, עבור מיון-הכנסה רץ יותר מהר ממיון-מיזוג. צריך להיות גדול ממש

בסדר

תרגיל 1.2-3

נפתור את האי-שיוויון הבא:

הפונקציות עולות ממש ולא חסומות לכן נחפש עבורו מתקיים (1) וגם עבור מתקיים .

עבור :

עבור :

לכן, זהו הערך הקטן ביותר שבו אלגוריתם שזמן ריצתו רץ מהר יותר. בסדר

**שאלה 1 סעיף ב'**

נוכיח את הטענה כי .

עלינו למצוא כך שיתקיים . לכל .

כאשר ניקח

מתקיים .

הפונקציה עולה ממש ולא חסומה, בנוסף, לכל מתקיים

ולכן, האי-שיוויון ממשיך להתקיים כי כל פעם אנחנו מוסיפים עוד איבר חיובי (שגדול ממש מ 1) לצד ימין של האי-שיוויון, בעוד שצד שמאל גדל בדיוק ב 1. בסדר

**שאלה 1 סעיף ג'**

הזוג הראשון: בסדר

הזוג השני: בסדר

כל הפונקציות חיוביות, עולות ולא חסומות

נבדוק :

נגדיר

ולכן,

נבדוק :

נגדיר

ולכן,

נבדוק :

מתקיים:

נבדוק :

מתקיים:

**שאלה 2 סעיף א'**

שמורת הלולאה החיצונית (שורות 3 – 7):  
בסוף כל איטרציה, מספר הפעמים שעברנו על כל תא בתת-מערך הוא זוגי כאשר הוא אינו ריבוע שלם או א"ז כאשר הוא ריבוע שלם.

**אתחול:** כאשר , התת-מערך שווה לתא , והלולאה עברה על התא מספר א"ז של פעמים.

**תחזוקה:**

טענת עזר: לכל מספר שלם שהוא ריבוע שלם, יש מספר א"ז של מחלקים בניגוד למספר שלם שאינו ריבוע שלם שלו יש מספר זוגי של מחלקים.

הוכחה: לכל מספר שלם שאינו ריבוע שלם מתקיים , כאשר , לכן ישנם זוגות מספרים, ולכן מספר המחלקים של הוא זוגי.  
לכל מספר שלם שהוא ריבוע שלם מתקיים הנ"ל, אבל יש עוד מחלק אחד שנכפל בעצמו המקיים , כאשר , וזה המחלק שמפר את הזוגיות, ולכן מספר המחלקים של הוא א"ז.

נניח ששמורת הלולאה מתקיימת בסוף איטרציה , ונראה שהיא ממשיכה להתקיים בסוף איטרציה .

כאשר אינו ריבוע שלם השגרה FLIP נקראת לפי טענת העזר מספר זוגי של פעמים ולכן ערכו של התא אינו משתנה מערכו ההתחלתי (FLASE) – שמורת הלולאה ממשיכה להתקיים בתת-מערך .

כאשר ריבוע שלם השגרה FLIP נקראת לפי טענת העזר מספר א"ז של פעמים ולכן ערכו של התא משתנה מערכו ההתחלתי FALSE לערך חדש TRUE – שמורת הלולאה ממשיכה להתקיים בתת-מערך .

**סיום:** כאשר , התת-מערך מקיים את שמורת הלולאה, אבל התת-מערך שווה בעצם לכל המערך, ולכן, מספר הפעמים שעברנו על כל תא ריבוע לא שלם / ריבוע שלם הוא זוגי / א"ז בהתאמה.

בסדר

שמורת הלולאה הפנימית (שורות 5 – 7):  
בסוף כל איטרציה הערך שבכל תא בתת-מערך שווה TRUE אם"ם ריבוע שלם.

**אתחול:** כאשר , התת-מערך הוא בעצם איבר יחיד שערכו שווה ל TRUE.

**תחזוקה:** נניח ששמורת הלולאה מתקיימת בסוף איטרציה k, ונראה שהיא ממשיכה להתקיים בסוף איטרציה .

בתחילת האיטרציה ה , לפי טענת העזר ושמורת הלולאה החיצונית, במידה ו הוא מספר ריבוע שלם, זה אומר שעברנו על התא מספר זוגי של פעמים ולכן בסוף האינרטציה ה נעבור עליו שוב וערכו ישתנה ל FALSE.

בתחילת האיטרציה ה , לפי טענת העזר ושמורת הלולאה החיצונית, במידה ו הוא לא מספר ריבוע שלם, זה אומר שעברנו על התא מספר א"ז של פעמים ולכן בסוף האינרטציה ה נעבור עליו שוב וערכו ישתנה ל TRUE.

**סיום:** כאשר , התת-מערך מקיים את שמורת הלולאה, אבל התת-מערך שווה בעצם לכל המערך, ולכן, הערך שבכל תא במערך שווה TRUE אם"ם ריבוע שלם.

בסדר

**שאלה 2 סעיף ב'**

נסתכל על השגרה FLIP:

השגרה מבצעת 2 פעולות (השווה והשמה) ולכן זמן הריצה שלה הוא .

נסתכל על השגרה TRUE\_SQUARES:

לולאת ה for הראשונה (שורות 1 – 2) רצה מ 1 עד ומבצעת פעולה אחת בסדר-גודל של .

לולאת ה for השניה (שורות 3 – 7) רצה מ 1 עד ועל כל איטרציה לולאת ה while (שורות 5 – 7) מבצעת פעולות.

לכן זמן הריצה של השגרה הוא

ולכן הסדר-גודל של השגרה הוא . בסדר

**שאלה 3 סעיף א'**

בעיה 2-4

1. בסדר
2. מספר ההיפוכים הרב ביותר הוא כאשר המערך ממוין בסדר יורד.

לאיבר הראשון במערך יהיה היפוכים

לאיבר השני המערך יהיה היפוכים וכן הלאה

לדוגמה,

לכן מספר ההיפוכים הוא: בסדר

1. מיון-הכנסה מבצע החלפה עבור כל היפוך שקיים במערך, ככל שיהיו יותר היפוכים זמן הריצה של מיון-הכנסה יהיה יותר גדול, בסעיף ב' הראינו את המקרה הגרוע שעליו אפשר להריץ את מיון-הכנסה והתוצאה תואמת (בסדר גודל) לזמן הריצה של מיון-הכנסה .

מיון הכנסה עובר מהאיבר השני במערך עד הסוף. עבור *הוא משווה ל .* אם מדובר בהיפוך - – הוא מחליף ביניהם מקום ומשווה ל *וכן הלאה עד שמגיע לאיבר שאינו יוצר עם* (המקורי, ללא התחשבות בהחלפות המקום) היפוך, או לתחילת המערך. כלומר, מספר החלפות המקום במיון הכנסה הוא מספר ההיפוכים במערך. בסעיף הקודם ראינו את מספר ההיפוכים במקרה הגרוע ביותר, ואכן במיון הכנסה מספר ההשוואות (והחלפות המקום) הוא סדרה חשבונית זהה לסדרה הנ"ל.

1. ניעזר במיון מיזוג, בשלב המיזוג נבדק איזה איבר מ 2 המערכים צריך להכניס קודם למערך החדש, שזה בדיוק היפוך.

נעזר באלגוריתם שנמצא בספר הקורס בעמוד 27:

Merge-Sort(A, p, r)

1. If (p < r)
   1. q 🡨
   2. return Merge-Sort (A, p, q) + Merge-Sort (A, q + 1, r) + Merge (A, p, q, r)
2. return 0

בנוסף נעזר באלגוריתם של המיזוג הנמצא במדריך הלמידה בעמוד 15:

1. לפני שורה 1  
   נוסיף את השורה: counter 🡨 0 המשתנה אחראי על ספירת ההיפוכים.
2. בין שורות 11 – 12: בתוך ה else  
   נוסיף את השורה: counter 🡨 counter + 1, כיוון שהמיקום הזה אומר שיש מספר במערך השמאלי שיותר גדול ממספר מהמערך הימני (היפוך)
3. בין שורות 15 – 16: בתוך ה do  
   נוסיף את השורה: counter 🡨 counter + 1, כיוון שהמיקום הזה אומר ששאר המספרים שנותרו במערך השמאלי גדולים יותר מכל המספרים שבמערך הימני (היפוך).
4. לאחר שורה 22: מחוץ ל for  
   נוסיף את השורה: return counter, על מנת להחזיר את מספר ההיפוכים.

בסדר

כאשר מבקשים לכתוב אלגוריתם בקורס, נדרש:

1. תיאור הרעיון הכללי
2. פירוט האלגוריתם – אפשר בפסודו קוד
3. דיון/הוכחת נכונות – ניתן לרשום מבוסס על האלגוריתם בספר בעמוד... וההוכחה בעמוד ...
4. דיון בסיבוכיות– ניתן לרשום מבוסס על האלגוריתם בספר בעמוד... וניתוח הסיבוכיות בעמוד ...

**שאלה 3 סעיף ב'**

כמות השגיאות במערך הוא המשלים לכמות השגיאות במערך , כיוון שכל זוג שהוא היפוך במערך הוא לא היפוך במערך וכן ההפך.

מצאנו בסעיף א' שמספר הזוגות שיש במערך הוא . בסדר

לכן,

**שאלה 4 סעיף א'**

IS\_V (A)

לשגרה יש רק 2 פעולות (חיסור והשוואה), ולכן זמן הריצה של השגרה הוא .

הסבר האלגוריתם:

השגרה מקבלת בפרמטר מערך ממוין A אשר כל מספר מופיע בו פעם אחת בלבד.

נתייחס ל 2 מקרים אפשריים:

מקרה 1 – ההפרש בין כל ערך במערך הוא 1 (לדוגמה, ) ולכן אין שום מספר חסר שנמצא בטווח המערך.  
השגרה תחזיר FALSE, כי , (בדוגמה, )

מקרה 2 – ההפרש בין כל ערך במערך לא קבוע (לדוגמה, )  
השגרה תחזיר TRUE, , (בדוגמה, ), כי ההפרש בין המספר האחרון לראשון תמיד יהיה יותר גדול / שווה לכמות האיברים במערך.  
כאשר חסר מספר אחד, ההפרש יהיה שווה למספר האיברים במערך וככל שיהיו חסרים יותר איברים כך יגדל ההפרש. בסדר

**שאלה 4 סעיף ב'**

הסבר האלגוריתם:

השגרה מקבלת בפרמטר מערך ממוין A אשר כל מספר מופיע בו פעם אחת בלבד.

נעזר באלגוריתם של חיפוש בינארי ונפצל בעזרת אינדקסים את המערך, אם בצד שמאל של המערך חסרים איברים (שורה 3.2), נחזיר את end אחורה ונסתכל על התת-מערך השמאלי, באותה מידה, אם בצד ימין של המערך חסרים איברים (שורה 3.3) נקדם את start קדימה ונסתכל על התת-מערך הימני.

כאשר נגיע לתת מערך בעל 2 תאים (שורה 3.4) מובטח לנו שההפרש בין תא 2 לבין התא שמחוץ לתת מערך מצד ימין גדול לפחות מ 1 ולכן נוכל להוסיף 1 לערך תא 2 (שורה 3.4.1)

כאשר נגיע לתת מערך בעל תא 1 (שורה 3.5) מובטח לנו שההפרש בין תא 1 לבין התא הקודם לו מצד שמאל גדול לפחות מ 1 ולכן נוכל להחסיר 1 לערך התא (שורה 3.5.1)

Binary\_V(A)

1. Start 🡨 1
2. End 🡨 length[A]
3. While (start <= end)
   1. Mid 🡨
   2. If (A[mid] – A[start] >= mid)
      1. End 🡨 mid - 1
   3. Else If (A[end] – A[mid] >= end)
      1. Start 🡨 mid + 1
   4. Else if (end – start = 1)
      1. Return A[end] + 1
   5. Else if (end = start)
      1. Return A[end] – 1
4. Return -1

סיבוכיות האלגוריתם:

הוספתי רק פעולות בסדר גודל של , לכן זמן הריצה של האלגוריתם המקורי (חיפוש בינארי) נשמר ולכן זמן הריצה של האלגוריתם הנ"ל הוא .

בסדר

**שאלה 5 סעיף א'**

כאשר :

מתקיים

לכן לפי מקרה 1 של שיטת האב (עמוד 45 במדריך):

כאשר :

מתקיים

לכן לפי מקרה 2 של שיטת האב (עמוד 45 במדריך):

כאשר :

מתקיים

לכן לפי מקרה 3 של שיטת האב (עמוד 45 במדריך):

בסדר

**שאלה 5 סעיף ב'**

נבדוק תחילה את המקסימום בין :

(כל פולינום גדל מהר יותר מכל פולילוג)

ולכן, .

נבדוק אם כאשר זוגי:

*בנוסף מתקיים*  *ולכן לא ייתכן כי קיים חיובי שגדול מהביטוי לא משנה כמה יהיה גדול.*

*לכן גם* *לא מתקיים משיקולים סימטריים.*

לכן גם לא מתקיים כי לא קיים אפילו אחד, כמובן שלא יתקיים לכל .

נבדוק אם כאשר א"ז:

*בנוסף מתקיים*  *ולכן לא ייתכן כי קיים חיובי שקטן מהביטוי לא משנה כמה יהיה גדול.*

*לכן גם* *לא מתקיים משיקולים סימטריים.*

לכן גם לא מתקיים כי לא קיים אפילו אחד, כמובן שלא יתקיים לכל .

נבדוק אם :

לא מתקיים כיוון ש וגם לא מתקיימים. בסדר