

מבוא לתכנות מערכות 10010 סמסטר א' – תשפ"ד

תרגיל בית מס' 1

נושא התרגיל: מטריצות מצביעים

תרגיל הבית יוגש בזוגות על פי רישום מוקדם
יש להגיש את התרגיל בשני מקומות:

1. בסביבת casain - מערכת לבדיקה אוטומטית, הקישור ב Moodle
2. תיבת הגשה מתאימה של מטלה ב Moodle עבור בדיקה ידנית

הנחיות להגשה עבור הדיקה הידנית:

- התרגיל ייבדק בסביבת ubuntu
- הקוד חייב לעבור קומפילציה, קוד שאינו מתקמפל לא ייבדק.
- יש להגיש קוד ללא הערות קומפילציה, warnings, קוד בו יהיו הערות יגרור הורדה של 10 נקודות.
- יש להגיש רק קבצי Makefile ו h i c מקובצים לקובץ אחד ששמו שם פרטי ומשפחה של שני המגישים. לדוגמא: Efrat_hertzberg_Moshe_Cohen.zip
- שני המגישים צריכים לעלות את העבודה למודל.
- אין להגיש פרויקט ב eclipse או קבצי obj והרצה (יגרור הורדה של 5 נקודות).
- ב Makefile חייבת להיות אפשרות ל clean

הוראות כלליות:

1. יש להקפיד על כללי הנדסת התוכנה:
 1. פונקציה לא אמורה להיות ארוכה מ 25 שורות. אם יש קטע קוד שניתן לתת לו כותרת מה הוא מבצע יש לכתוב אותו בפונקציה נפרדת.
 2. יש לחלק את הקוד לתת פונקציות מתאימות. אין לשכפל קוד. קוד זהה אמור להיות בפונקציה מתאימה.
 3. יש לחלק כל שאלה לקבצי h i c נפרדים, בנוסף יש לכתוב פונקציות כלליות בקובץ נפרד וכן קובץ c ראשי עם התפריט וקריאה לפונקציה הראשית של התרגיל אותו המשתמש בחר.
 4. מבנה התוכנית (הזחות) ותיעוד במידת הצורך.
 5. חובה להשתמש בקבועים במקומות המתאימים.
 6. יש להשתמש בפונקציות קצרות, כלליות, קריאות ושימושיות.
 7. יש להקפיד על בדיקת תקינות קלט, אפשר להניח שאם ביקשו מהמשתמש מספר הוא הכניס מספר אך יתכן ולא בטווח הנכון.
 8. הפלט צריך להיות כפי שניתן בתרגיל במידה וניתן.
 9. קוד קצר, לא מסורבל ויעיל הן מבחינת כתיבתו והן מבחינת ריצת התוכנית.

בית זה יש 3 משימות

יש לכתוב תכנית שמציגה תפריט המאפשר למשתמש לבחור את המשימה אותה הוא רוצה להריץ וכן תהיה אפשרות לצאת. התפריט הראשי יוצג כל עוד לא בחרו לצאת. כלומר בסיום כל משימה יוצג התפריט הראשי שוב. בחירת E/e תצא הודעה למסך "Bye Bye" ונצא מהתוכנית.

לפי ההוראות הראשיות כמה קבצים צריכים להיות בפרויקט???

התפריט הראשי

Please choose one of the following options:

S/s – Sudoku

A/a – Add Lines

P/p – Picture Manipulation

E/e – Exit

להצגת התפריט יש להשתמש בתבנית switch, יש לדאוג שהמשתמש יוכל להקליד אות קטנה או גדולה.

בפונקציה main, עבור כל אפשרות בתפריט יש לכתוב פונקציה נפרדת, אין קוד ב switch למעט קריאה לפונקציה זו.

1. מטריצה ו/או מערך יוגדרו רק בתחילת הפונקציה הראשית של כל תרגיל.

2. כל הפונקציות יקבלו את המטריצה ככתובת תחילת מערך.

3. למרות ש ubuntu מאפשר – אין להגדיר מטריצה עם פרמטר, רק ע"י קבוע!!!

~~int x=5;
int arr[x];~~
#define N 5
int arr[N];

4. שים לב: בתתי פונקציות אף פעם אין שימוש בקבועים!!!

משימה 1 - Sudoku:

יש לכתוב תוכנית שבודקת נכונות של לוח תשבץ סודוקו. בגודל $N \times N$ כך ש \sqrt{N} מספר שלם. המטריצה תייצג את לוח הסודוקו.

את הבדיקות יש לבצע ע"י מעבר בודד על כל שורה, עמודה ותת ריבוע.

שים לב שהמטריצה נשלחת לפונקציות ככתובת תחילת מערך!!!

בפונקציה הראשית של המשימה מוגדרת המטריצה (לוח המשחק) וכן ניתן להגדיר עוד מערך עזר אחד לצורך עבודה יעילה.

את לוח המשחק הראשי נגדיר בגודל מקסימאלי של $N \times N$, ($N = 25$ בבדיקות האוטומטיות), אך בפועל המשתמש יבחר גודל לוח משחק החייב להיות שווה או קטן מ N , וכן מספר שיש לו שורש ריבועי.

לדוגמא: N יהיה 25 אך המשתמש יוכל להחליט לעבוד עם מטריצה בגודל 4×4

מהלך checkSudoku:

הערה: כל שלב הוא לפחות פונקציה אחת נפרדת

- יש לקלוט מהמשתמש גודל לוח תקין לסודוקו.
- יש לאתחל את המטריצה תוך קליטת נתונים מהמשתמש.
- יש להציג את המטריצה.
- ישלכתוב פונקציה אשר מקבלת את הלוח ומשתנים נוספים הנחוצים ומחזירה האם הלוח תקין.
- לוח סודוקו תקין אם כל השורות, עמודות וריבועים תקינים.
- יש להדפיס את תוצאת הבדיקה

יש לחלק את העבודה לפונקציות עזר לפחות 5. (לי יש 8)

| | | | |
|---|---|---|---|
| 2 | 1 | 4 | 3 |
| 4 | 3 | 2 | 1 |
| 3 | 2 | 1 | 4 |
| 1 | 4 | 3 | 2 |

לוח תקני

| | | | |
|---|---|---|---|
| 2 | 1 | 4 | 3 |
| 4 | 3 | 2 | 1 |
| 4 | 2 | 1 | 4 |
| 1 | 4 | 3 | 2 |

לוח לא תקני

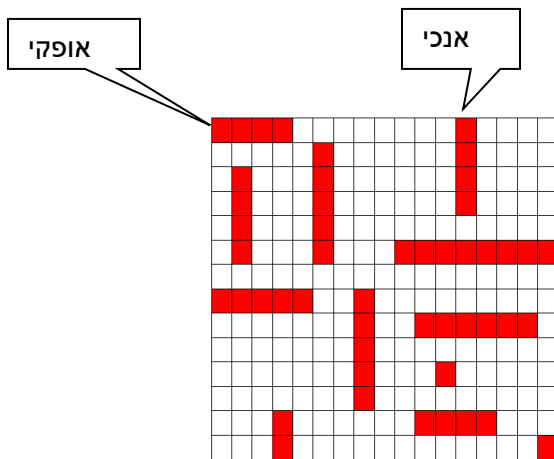
$n = 4$

רמז: פונקציה memset המוגדרת string.h יכולה לשמש אתכם לאיפוס מערך העזר.

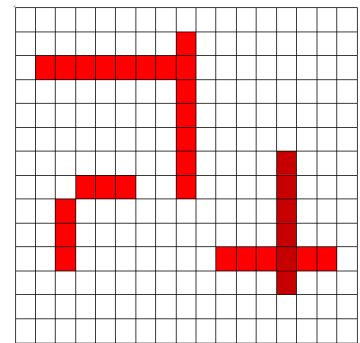
משימה 2 – Add Lines:

בשאלה זו נטפל בקווים בתמונה. קו מיוצג ע"י 4 מספרים המציינים את ה (x,y) של נקודות הקצה של הקו, $(x1,y1,x2,y2)$. במידה והקו אופקי ידוע שנקודה 1 שמאלית ל 2, במידה והקו אנכי ידוע שנקודה 1 עליונה לנקודה 2. מדובר באינדקסים במטריצה המתחילים ב 0.

- הצורה היחידה האפשרית היא קו אופקי או אנכי.
- כל ה"פיקסלים" בקו צמודים.
- אין חיתוך בין הקווים או הצמדה של קו לקו אחר.
- קו לא "גולש" מהתמונה.



התמונה תקינה



התמונה לא תקינה

המשימה מבצעת את הפעולות הבאות:

הערה: כל שלב הוא לפחות פונקציה אחת נפרדת

1. הגדר מטריצה בגודל ROWS שורות ו COLS (50, 50 בבדיקות האוטומטיות). הדפס ערכים אילו למסך. צור את המטריצה כך שכל ערכה יהיו 0.
2. קלוט מהמשתמש שני מספרים אשר יגדירו את גודל המטריצה האפקטיבי כך שמספר השורות יהיה בין 1 ל ROWS כולל ומספר העמודות יהיה בין 1 ל COLS כולל. הדפס ערכים אילו למסך.

אין להגדיר עוד מטריצה מלבד המטריצה הגדולה.

3. כל עוד המשתמש מעוניין להמשיך ולהכניס קווים לתמונה קלוט 4 מספרים מהמשתמש המייצגים את הקו בסדר: $(x1,y1,x2,y2)$.
- חובה לממש זאת בפונקציה נפרדת, `getLine`, אשר מקבלת את גודל המטריצה האפקטיבית ומחזירה את נתוני הקו (4 מספרים). הפונקציה בודקת את חוקיות המספרים, מחזירה 1 במידה וחוקיים ו 0 אחרת.
4. אם הסעיף הקודם החזיר 0, הקו לא חוקי, יש להודיע הודעה למשתמש ולבקש שנית נתוני קו.
5. בדוק אם ניתן להכניס את הקו לתמונה לפי החוקיות שהוסברה.
6. אם ניתן להכניס את הקו לתמונה יש להכניס את הקו אחרת יש להודיע למשתמש שלא ניתן להכניס את הקו לתמונה.

7. יש החזר לפונקציה הראשית כמה קווים יש בתמונה ולהדפיס נתון זה לפני סיום המשימה.

משימה 3 - Picture Manipulation:

נתייחס למטריצה כתמונה ונאפשר למשתמש לסובב אותה בכמה אופנים:

1. סיבוב של 90 מעלות עם כיוון השעון

2. סיבוב של 90 מעלות נגד כיוון השעון.

3. היפוך סביב ציר אנכי.

4. היפוך סביב ציר מאוזן.

לצורך המשימה המטריצה תהיה מטריצה של `int`, המטריצה ריבועית אך אין להניח שהיא בעלת גדלים זוגיים.

יש לכתוב פונקציה ראשית למשימה המציגה תפריט עם ארבעת האפשרויות ומבצעת את האפשרות אותה המשתמש בחר.

יש לאתחל את המטריצה ע"י מספרים רנדומלים, בחר טווח כרצונך.
יש לחלק לתתי פונקציות. קוד זהה צריך להיות בפונקציה.

בהצלחה