

**תרגיל 4 – מצביעים והקצאה דינאמית****הגשה עד 13.01.2020 בשעה 23:50****הוראות הגשה:**

1. שאלות ובקשות בקשר לעבודה להפנות אך ורק למרצה האחראית, גב' סבטלנה חסין, במייל: [sceassign2016@gmail.com](mailto:sceassign2016@gmail.com).
  2. תרגילים הם ביחידים! כל עבודה משותפת היא אסורה ותיענש בחומרה!
  3. להגיש רק תכניות שעוברות קומפילציה על מהדר שפת C gcc Ubuntu Linux.
  4. ההגשה היא של קבצי הקוד (קובץ c). בלבד. יש ליצור 3 קבצים: part1.c, part2.c ו-part3.c, לכוון את כל הקבצים לקובץ אחד בפורמט RAR או ZIP, ולהגיש רק קובץ זה.
  5. בתחילת הקובץ יש להוסיף את התיעוד הבא:
- /\* Assignment: 4  
Author: Israel Israeli, ID: 01234567  
\*/
- כמובן שיש לעדכן את השמות ומספרי תעודות הזוהות שלכם.
6. הארכות יינתנו אך ורק במקרים חריגים (מילואים, אבל על קרובים ומחלה חריפה!) ובצרוף אישורים מתאימים. כמו כן במקרה של ידע מוקדם חובה ליצור קשר עם המרצה האחראית על התרגיל לפחות יומיים לפני חלוף הدد-ליין!
  7. ההגשה היא עד התאריך האחרון לתרגיל: 13/01/20 בשעה 23:50. הגשה מאוחרת אפילו בדקה – לא תתקבל (המערכת חוסמת את אפשרויות ההגשה!). קחו זאת בחשבון ותכננו את זמנכם בהתאם!
  8. מותר לכם ומומלץ ליצור פונקציות עזר.
  9. אין להשתמש בתרגיל בחומר שטרם נלמד, או שנלמד לאחר נושא התרגיל, אלא אם נכתב במפורש בתרגיל שמותר.
  10. הקלטים יהיו מהטיפוסים החוקיים. ז"א בכל מקום שצריך להכניס מספר שלם – נכניס מספר שלם (ולא שבר או אות). אנחנו לא מתחייבים שהוא יהיה חיובי או א-שלילי, או בטווח מסוים – אלא אם כן נאמר אחרת בשאלה עצמה.
  11. בכל פעם שהשתמש מקליד קלט שגוי התוכנית מבקשת קלט חוזר.
  12. אחרי כל הדפסה יש לבצע ירידת שורה.
  13. בתרגיל יש להשתמש בספריות malloc, stdlib, string, stdio בלבד!
  14. יש להקפיד על תכנות נכון:
- a. כל הערכים שהם קבועים, (מבחינה לוגית הם לא אמורים להשתנות), חייבים להיות מוגדרים כ: define, const או enum, בהתאם לצורך.
  - b. יש לרשום הערות.

- c. יש להקפיד על הזחות!!! כיתוב נכון וקריא! ושמות משמעותיים!
- d. יש לנסות ולייעל את הקוד והתוכנית ככל שניתן.
- e. לפני בקשת קלט (scanf) יש להדפיס למשתמש הוראה (printf) איזה קלט מבוקש.
- f. יש להקפיד על מוסכמות התכנות הנכון (שמות כמו שצריך וכו').
- g. יש להקפיד על כל כללי התכנות הנכון כפי שנלמדו בכיתה.

בהצלחה ☺

## חלק א – מערכים חד-ממדי דינאמיים – מענה בקובץ part1.c – 31 נק'

### משימה מס' 1 – מחרזת דינאמית :

✓ ( 10 נק') כתבו פונקציה בשם **CreateNewString** (char\* str) char\* מקבלת מחרוזת המכילה רצפים של תווים באזורים שונים, מייצרת ומחזירה מחרוזת חדשה כך שכל רצף תווים זהים מהמחרוזת הקודמת יוחלף בשני תווים: התו שחוזר ברצף, וסיפורה המייצגת את אורך הרצף. למשל :

מחרוזת המתקבלת: aaabbbbccddddd

מחרוזת המוחזרת: a3b4c2d6

מחרוזת המתקבלת: aaabbc\*\*\*\*\*@@@

מחרוזת המוחזרת: a3b2c1\*5@3

הערה: ניתן להניח שלא יהיה רצף של יותר מ-9 תווים זהים במחרוזת המתקבלת, כלומר אורך הרצף יהיה לכל היותר באורך תו אחד בלבד (ספרה אחת).

### משימה מס' 2 – מערך חד-ממדי דינאמי :

✓ ( 4 נק') כתבו פונקציה **BuildArray**(int\* size) int\* שקולטת מהמשתמש את גודל המערך, מקצה זיכרון בהתאם, ממלא אותו בערכים כקלט מהמשתמש ומחזירה את המערך החדש.

✓ ( 2 נק') כתבו פונקציה **PrintArray**(int\* group, int size) void המקבלת כפרמטר מערך, כולל הגודל שלו, ומדפיסה את אברי המערך בפורמט הבא:

( 1 , 3 , 4 , 5 )

מערך ריק תודפס:

( )

✓ ( 10 נק') כתבו פונקציה **MergeTwoArrays**(int\* arr1, int size1, int\* arr2, int size2, int\* pSize) int\* המקבלת שני מערכים של מספרים שלמים arr1 ו-arr2 בגדלים size1 ו-size2 ממיינים בסדר עולה ( ניתן להניח שהמערכים ממיינים ומכילים ערכים שאינם חוזרים על עצמם - אין צורך לבדוק זאת), יוצרת ומחזירה את מערך החדש הממוין המכיל את איברי המערכים arr1 ו-arr2 ללא איברים חוזרים, למשל הפונקציה תקבל:

( 1 , 2 , 5 , 8 )

( 2 , 4 , 6 , 8 )

הפונקציה תחזיר מערך :

( 1 , 2 , 4 , 5 , 6 , 8 )

( 5 נק') כתבו תוכנית הראשית שקולטת מחזרת ומעבירה אותה לפונקציה הראשונה ומדפיסה את המחזרת החדשה שהפונקציה החזירה. לאחר-מכן, בעזרת הפונקציות של המשימה השנייה התוכנית הראשית יוצרת 2 מערכים, ממזגת אותם למערך אחד, מדפיסה את המערך הממוזג ובסוף משחררת את כל הזכרון .

### חלק בי – מערכים דו-ממדי דינאמיים – מענה בקובץ - part2.c – 30 נק'

#### משימה מס' 3 – מערך דו-ממדי דינאמי :

✓ ( 6 נק') כתבו פונקציה `void BuildMatrix(int*** matrix, int row, int column)`. הפונקציה תקבל מצביע למערך דו-מימדי דינאמי, כמות השורות וכמות העמודות. הפונקציה תקצה זיכרון בהתאם, ותמלא את המטריצה בערכים.

✓ ( 5 נק') כתבו פונקציה `void FreeMatrix(int*** matrix, int row)`. הפונקציה תקבל מצביע למערך דו-מימדי דינאמי וכמות השורות, ותשחרר את כל הזיכרון שהמטריצה תפסה .

✓ ( 3 נק') כתבו פונקציה `void PrintMatrix(int** matrix, int row, int column)`. הפונקציה תקבל מערך דו-מימדי דינאמי, כמות השורות וכמות העמודות. הפונקציה תדפיס את איברי המטריצה בתצוגת טבלת דו-ממדית .  
למשל,

1 2 4 7

8 7 6 1

✓ ( 13 נק') כתבו פונקציה `int** SubMatrix(int** matrix, int sizeMatrix, int* pSizeSubMatrix)`. הפונקציה מקבלת מערך דו- מימדי - מטריצה ריבועית של ערכים שלמים (מספר השורות במטריצה זהה למספר העמודות). הפונקציה מחזירה את תת-המטריצה הריבועית שסכום איבריה הוא המקסימלי.

לדוגמא: הפונקציה מקבלת מטריצה הבאה -

4	-9	20	6
0	15	-3	7
3	-50	9	10
-19	140	70	-80

הפונקציה מחזירה את תת-המטריצה:

-50	-9
140	70

(3 נק') כתבו תוכנית ראשית הקולטת גודל המטריצה, יוצרת מטריצה ריבועית (מספר השורות במטריצה זהה למספר העמודות), מדפיסה אותה, מחזירה את תת-המטריצה הריבועית שסכום איבריה הוא המקסימלי, מדפיסה אותה ובסוף משחררת את כל הזכרון.

## חלק ג' – מערך מחרוזות דינאמי – מענה בקובץ part3.c – 39 נק'

✓ (10 נק') כתבו פונקציה `GetAllNames(int* countNames, char** GetAllNames)`. הפונקציה תקלוט את מספר שמות המבוקש, תיצור מערך בהתאם ותמלא אותו בנתוני הקלט מהמשתמש. יש להכניס את הטקסט בפורמט הבא: שם משפחה ושם פרטי וביניהם לפחות רווח אחד. הפונקציה תחזיר את הכתובת של המערך החדש. השם יכול להיות עד 80 תווים לכל היותר. ניתן להניח ששמות המשפחה והשם הפרטי מתחילים מאות גדולה ושאר התווים הם אותיות קטנות. למשל,

How many names do you want?

3

Enter 3 names:

Sarussi Yossi

Afflek Ben

Sarussi Ron

✓ (3 נק') כתבו פונקציה `FreeAllNames(char*** names, int countNames)`. הפונקציה תקבל מערך שמות ואת גודלו ותשחרר את כל הזיכרון שהמערך תופס.

✓ (3 נק') כתבו פונקציה `PrintAllNames(char** names, int countNames)`. הפונקציה תקבל מערך שמות ואת גודלו ותדפיס את השמות השמורים במערך.

✓ (10 נק') כתבו פונקציה `UpdateAllNames(char** names, int countNames)`. הפונקציה תקבל מערך שמות ואת גודלו ותמחק את כל הרווחים המיותרים במידה וישנו יותר מרווח אחד בין שם המשפחה ופרטי ואת כל הרווחים בהתחלה ובסוף.

למשל עבור דוגמת הקלט המערך יתעדכן בצורה הבאה:

Sarusi Yossi

Afflek Ben

Sarussi Ron

✓ ( 10 נק') כתבו פונקציה `void SortAllNames ( char*** names, int countNames )`. הפונקציה תקבל מערך שמות ואת גודלו ותמיין אותו בסדר עולה לקסיקוגרפית לפי שמות המשפחה ושמות הפרטיים. למשל עבור דוגמת הקלט (אחר העידכון) המערך יתעדכן בצורה הבאה:

Afflek Ben

Sarussi Ron

Sarussi Yossi

( 3 נק') כתבו תוכנית הראשית שבעזרת פונקציות שיצרתם, בונה מערך שמות, מתקנת אותו, ממיינת, מדפיסה ובסוף משחררת את כל הזכרון.