

מטלה מסכמת עקרונות התכנות ושפת C

יש לשים לב בהגשת המטלה לדגשים הבאים:

- אנא קראו את המטלה בעיון רב. מומלץ לקרוא את המטלה במלואה מספר פעמים לצורך הבנת התמונה הכוללת.
- שמות משתנים משמעותיים
- יש לתת הודעות קלט ופלט ברורות
- טיפול נכון בזיכרון – כולל בדיקות שהקצאה הצליחה ושחרור זיכרון. במקרה והקצאת זיכרון לא מצליחה יש להדפיס הודעת שגיאה ולסיים את פעולת התכנית.
- כתובות המועברות לפונקציות (מצביעים) – יש לוודא כי כל מצביע שהועבר לפונקציה שאינה עורכת שינוי בערך הנמצא בכתובת זו יועברו כ-const.
- כ-20% מציון כל סעיף יוקדש **להערכת קריאות הקוד** – תחת הדגשים הבאים:
 - מניעת שכפול קוד – באמצעות עבודה עם פונקציות עזר וכו'
 - אורך הקוד – מספר השורות בכל פונקציה
 - שימוש יתר במשתנים (זיכרון מיותר) ושמות משתנים משמעותיים
 - מימוש יעיל של קוד
 - יש לכתוב הערות לקוד, בפרט בתחילת כל פונקציה יש להוסיף הערות אשר מתארות את מטרת הפונקציה, הארגומנטים ומה היא מחזירה. בנוסף, יש להקפיד על הערות אשר מתארות הקוד באלגוריתם והמשמעות של משתנים.

הוראות הגשה:

- הגשת המטלה הינה **בזוגות בלבד** בתוך הקבוצה
- יש להגיש את התוכנית במלואה לקובץ WORD - יש לציין את **שמות המגישים ות.ז. (במודגש!) בתחילת קובץ זה.**
- יש להגיש את קובץ המקור (הקובץ עם סיומת c.) מתוך הפרוייקט ב VS .
- את הקבצים הללו (קובץ ה WORD וקובץ ה c.) ולהעלות לאזור המשימה באתר הקורס- **אין להעלות את הקבצים בפורמט של zip או rar.**
- מועד אחרון להגשה: 17.2.2021 עד לסוף היום (23:59) - לא ניתן לאחר בהגשה: **בתאריך זה, בשעה הנקובה, המערכת תיסגר באופן אוטומטי להגשות.**
- שאלות לצוות הקורס יש להפנות באמצעות קובץ השאלות **בלבד** ולא באופן פרטי – השאלות תתרכזנה בהבנת המשימה. שאלות הקשורות לאופן המימוש או יישום הסטודנט לא יקבלו מענה- לינק לקובץ השאלות יפורסם באתר המודל.

מימוש מערכת לניהול מידע במרשם האוכלוסין

נדרש לממש מערכת לניהול מידע של תושבים במרשם האוכלוסין. המערכת תמומש בשפת C ויוחזק בה מידע אישי לכל תושב במבנה הנתון על פי השדות הבאים:

- **Id**: מספר תעודת זהות - מספר שלם חיובי בגודל של 8 בתיים.
- **Name**: שם פרטי - מחרוזת המנוהלת כמערך דינאמי בהתאם לאורך השם.
- **Family**: שם משפחה - מחרוזת המנוהלת כמערך דינאמי בהתאם לאורך שם המשפחה.
- **DateOfBirth**: תאריך לידה - מבנה בעל 3 שדות כמתואר להלן:
 - **Year**: שנה - (מספר שלם חיובי בן 4 ספרות) משתנה בגודל של 4 בתיים.
 - **Month**: חודש - (מספר שלם חיובי) משתנה בגודל של בית אחד.
 - **Day**: יום - (מספר שלם חיובי) משתנה בגודל של בית אחד (לשם פשטות הניחו שמספר הימים המקסימלי בחודש הוא 30).
- **PartnerId**: מס תעודת זהות של בן/בת הזוג - מספר שלם חיובי בגודל של 8 בתיים. אם לא קיים בן/בת זוג שדה זה מקבל ערך 0.
- **MotherId**: מס תעודת זהות של האם - מספר שלם חיובי בגודל של 8 בתיים. אם האם לא קיימת שדה זה מקבל ערך 0.
- **FatherId**: מס תעודת זהות של האב - מספר שלם חיובי בגודל של 8 בתיים. אם האב לא קיים שדה זה מקבל ערך 0.
- **NumOfChildren**: מספר הילדים - (מספר שלם חיובי) משתנה בגודל של בית אחד.
- **ChildrenPtr**: מערך דינאמי של תעודות הזהות של הילדים. אם אין ילדים מצביע זה מקבל NULL.

עליכם לכתוב תכנית ראשית אשר תבקש מהמשתמש להקליד את מספר התושבים בבסיס הנתונים ותקצה לבסיס הנתונים מערך דינאמי של תושבים מהמבנה המתואר לעיל. נדרש לייצר תפריט ניווט עבור הפעולות השונות אשר יתוארו בהמשך. יש להריץ את התפריט עד אשר המשתמש יבחר לסיים את פעילותו.

א. (4 נק') כתבו מבנה person המכיל את כל נתוני המידע האישי של תושב. שימו לב לטיפוסי הנתונים של כל שדה.

ב. (4 נק') כתבו פונקציית עזר `print_person`. הפונקציה מקבלת מצביע למבנה מסוג `person` ומדפיסה למסך את כל פרטי הרשומה (כולל מספר תעודות הזהות של ילדיו במידה והם קיימים).

ג. (4 נק') כתבו:

1. מבנה `db_mgr` שישמש כמבנה מנהל לבסיס הנתונים. המבנה יכול שדה של מצביע למערך דינאמי של רשומות מסוג `person` ושדה נוסף המתאר את גודלו של המערך.

2. פונקציה `init_db` אשר מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה לקרוא מהקלט את מספר התושבים במדינה ולבדוק שהמספר חיובי. על הפונקציה ליצור מערך דינמי בגודל מתאים לכלל התושבים ולעדכן את שדות המבנה המנהל בהתאם.

ד. (5 נק') כתבו פונקציה `menu` המדפיסה את תפריט הפעולות האפשרי על בסיס הנתונים. התפריט מתואר להלן:

Database System Menu:

1. Add person
2. Search a person
3. Search Parents
4. Delete a person
5. Get generation
6. Print database
7. Search by name
8. Quit

על הפונקציה להחזיר את התו שהוקלד. במידה והוקלד תו לא נכון על הפונקציה להדפיס הודעת שגיאה בהתאם ולהציג מחדש את התפריט ולהמתין לקלט מהמשתמש.

כעת נממש בכל סעיף נפרד כל אחת מהפעולות האפשריות על בסיס הנתונים.

ה. (12 נק') כתבו פונקציה `add_person` המממשת את פעולה 1 בתפריט. הפונקציה מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה:

- לקרוא מהקלט פרטים של תושב אחד. הפונקציה תדפיס הודעה מתאימה למסך ותבקש מהמשתמש להקליד כל אחד מהשדות של פרטי התושב בשורה נפרדת. על הפונקציה לדאוג להקצאת זיכרון דינאמי עבור השם הפרטי ושם המשפחה. בנוסף, על הפונקציה לקרוא מהמקלדת את מספר הילדים של התושב. אם המספר גדול מ 0, הפונקציה תקצה מערך דינאמי עבור תעודות הזהות של ילדי התושב, תדרוש משמשתמש להקליד את מספרי תעודות הזהות של כל אחד מילדי התושב ותאתחל אותם במערך הדינאמי בהתאמה.
- על הפונקציה לבדוק לכל קלט שהחזן שהוא תקין (לדוגמא מספר החודש בשנה). במקרה של קלט לא תקין יש להדפיס הודעת שגיאה ולחזור על בקשת הקריאה של הנתון.
- כל פרטי הרשומה של התושב שנקלט יוכנסו למערך התושבים הכללי. **על הפונקציה לשמור על סידור התושבים בסדר עולה במערך על פי מספר תעודת הזהות.** למשל באינדקס 0 יהיה מספר תעודת הזהות הנמוך ביותר ובאינדקס הגבוהה מספר תעודת הזהות הגבוהה ביותר.
- בנוסף, אם נעשה ניסיון להוסיף תושב כך שמספר התושבים חורג מגודל מערך התושבים הכללי, על הפונקציה להגדיל את המערך הדינאמי בהתאם ולעדכן את המבנה המנהל. **שימו לב שאין להשתמש בפונקציה realloc בפעולת הוספת התושב למערך.**

ו. (5 נק') כתבו פונקציה עזר `search_id`. הפונקציה מקבלת מצביע למבנה מנהל מסוג `db_mgr` ואת מספר הזהות של התושב. על הפונקציה:

- לחפש את מספר הזהות בבסיס הנתונים הכללי.
- להחזיר מצביע למבנה שנמצא. אם לא נמצא מבנה יוחזר `NULL`

במימוש הפונקציה יש לשים דגש על יעילות הקוד.

ז. (3 נק') כתבו פונקציה `search_person` המממשת את **פעולה 2 בתפריט**. הפונקציה מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה:

- להדפיס למסך בקשה לקריאת מספר הזהות של תושב ולקרוא מהמקלדת את מספר תעודת הזהות.
- על הפונקציה לקרוא לפונקציה עזר `search_id`.
- הפונקציה תדפיס למסך את כל פרטי התושב מבסיס הנתונים. במידה ותעודת הזהות לא נמצאה תודפס הודעה מתאימה.

ח. (12 נק') כתבו פונקציה **search_parents** המממשת את **פעולה 3 בתפריט**. הפונקציה

מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה:

- להדפיס למסך בקשה לקריאת מספר הזהות של תושב ולקרוא מהמקלדת את מספר תעודת הזהות. אם התושב לא נמצא בבסיס הנתונים על הפונקציה להדפיס הודעת שגיאה.

- להדפיס למסך את כל פרטי ההורים של התושב מבסיס הנתונים. אם רק אחד מההורים נמצא אז יש להדפיס את פרטיו. אם שני ההורים לא נמצאו אז יש להדפיס הודעת מתאימה.

אין לשכפל קוד. יש לעשות שימוש בפונקציות עזר שהוגדרו במסגרת השאלה.

ט. (12 נק') כתבו פונקציה **delete_person** המממשת את **פעולה 4 בתפריט**. הפונקציה

מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה:

- להדפיס למסך בקשה לקריאת מספר הזהות של התושב שימחק מבסיס הנתונים ולקרוא מהקלט את מספר תעודת הזהות. אם התושב לא נמצא על הפונקציה להדפיס הודעת שגיאה.

- למחוק את כל פרטי התושב מבסיס הנתונים, לשנות את גודל המערך הדינמי של בסיס הנתונים הכללי ולעדכן את המבנה המנהל בהתאם. למשל, אם לפני מחיקת התושב גודל המערך הוא 100 איברים לאחר המחיקה יש לשנות את גודלו ל-99.

- לשמור על הסדר העולה של מספרי תעודות הזהות בבסיס הנתונים.

- על הפונקציה לבדוק טרם המחיקה:

1. אם קיים לתושב בן/בת זוג, ואם כן אז לשנות בפרטי הרשומה של בן/בת הזוג את שדה ה- `PartnerId` לערך 0.

2. לחפש את ההורים של התושב שנמחק, ובמידה ונמצאו אז נדרש לעדכן את השדה של מספר הילדים של הוריו בהתאמה וכן למחוק את מספר תעודת הזהות שלו מהמערך הדינמי של תעודות הזהות של הילדים (המוצבע על ידי הרשומה של ההורה). שימו לב שפעולה זו תשנה את גודלו של המערך הדינמי.

3. אם קיימים לתושב ילדים בבסיס הנתונים, אז לשנות בפרטי הרשומה של הילדים את תעודת הזהות של ההורה המתאים (`FatherId` או `MotherId`) לערך 0.

הנחיות:

1. אין להשתמש בפונקציה `realloc`.

2. בביצוע חיפוש במערך יש להשתמש בפונקציות עזר שהוגדרו במסגרת השאלה.

י. (12 נק') כתבו פונקציה `get_gen` המממשת את **פעולה 5 בתפריט**. הפונקציה מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה:

- להדפיס למסך בקשה לקריאת מספר תעודת הזהות של תושב. אם התושב לא נמצא בבסיס הנתונים תודפס הודעת שגיאה.
- הפונקציה תחשב את מספר הדורות של צאצאי התושב ותדפיס אותו למסך. מספר הדור יחושב באופן הבא:

- אם אין לתושב ילדים בבסיס הנתונים מספר הדור יהיה 1.
- אם קיימים רק ילדים מספר הדור יהיה 2.
- אם קיימים רק ילדים ונכדים מספר הדור יהיה 3.
- אם קיימים רק ילדים, נכדים ונינים מספר הדור יהיה 4.
- וכו'

הנחיות:

1. אין להשתמש ברקורסיה.
2. ניתן להשתמש בהקצאות עזר דינאמיות

יא. (5 נק') כתבו פונקציה `print_db` המממשת את **פעולה 6 בתפריט**. הפונקציה מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה להדפיס את כל הרשומות במערך של בסיס הנתונים. עבור כל רשומה יש לתת כותרת מתאימה לכל שדה.

יב. (8 נק') כתבו פונקציה `search_by_name` המממשת את **פעולה 7 בתפריט**. הפונקציה מקבלת מצביע למבנה מנהל מסוג `db_mgr`. על הפונקציה:

- להדפיס למסך בקשה לקריאת שם פרטי ושם משפחה של תושב ולקרוא את המידע מהמקלדת.
- לחפש את כל התושבים בעלי שם פרטי ושם משפחה מתאימים להדפיס למסך את כל פרטי התושבים הללו. אם לא נמצאו תושבים מתאימים על הפונקציה להדפיס הודעת שגיאה.

יג. (8 נק') כתבו פונקציה `quit` המממשת את **פעולה 8 בתפריט**. על הפונקציה להדפיס הודעת על סיום הריצה. על הפונקציה לשחרר זיכרון שהוקצה דינאמית ולסיים את ריצת התכנית.

יד. (6 נק') כתבו תכנית ראשית המנהלת את בסיס הנתונים של כלל התושבים במדינה במערך של מבנים המוקצה דינאמית. כל מבנה מתייחס לתושב אחד. על התכנית לעשות שימוש בפונקציות שתוארו בסעיפים קודמים ולבצע את הפעולות הבאות:

- להדפיס למסך בקשת קלט ולקרוא מהמקלדת את מספר התושבים.
- להקצות מערך דינאמי בהתאמה.
- להציג הודעת תפריט ולקרוא לכל פונקציה בהתאם לפעולה שנבחרה.
- בסיום כל פונקציה, למעט פעולת Quit, יש להציג את התפריט מחדש ולחכות לקלט עד לבחירת פעולת סיום מהתפריט.
- לשחרר כל זיכרון דינאמי שלא יעשה בו שימוש. כמו כן לפני סיום התכנית יש לשחרר כל זיכרון דינאמי שהוקצה.

*** אזהרה:**

המרכז אקדמי רופין רכש לאחרונה מערכת ממוחשבת לזיהוי העתקות. כל הגשותיכם ייבדקו ע"י המערכת הזו. כל הגשה, לגבייה תהיה התראת העתקה – ולו הקטנה ביותר – תיפסל באופן אוטומטי וכל הסטודנטים הנוגעים בדבר יועלו באופן מיידי לוועדת משמעת!

בהצלחה!