

מבוא לתכנות מערכות 10010 סמסטר א' – תשפ"ג

תרגיל בית מס' 3

נושא התרגיל:

קבצים, void* מצביעים לפונקציות ורשימות מקושרות

יש להגיש אך ורק דרך תפריט המטלות שבאתר הקורס, כפי שהוסבר בתרגול.
הנחיות הגשה כלליות:

• **התרגיל ייבדק בסביבת Window בסביבת Visual Studio בלבד. יש**

לוודא לפני הגשה שהקוד והפרויקט תואם VS

- הקוד חייב לעבור קומפילציה, קוד שאינו מתקמפל לא ייבדק.

- **יש להגיש קוד ללא הערות קומפילציה, warnings, קוד בו יהיו הערות יגורר הורדה של 10 נקודות.**

- יש להגיש את תיקיית הפרויקט כולל קבצי `vcxproj` ו `sln`, מקובצת לקובץ אחד ששמו כשם הסטודנט (לא מספר ת"ז!!!). שם פרטי ומשפחה.

- ניתן לעבוד בזוגות - במידה ומגישים בזוג קובץ ההגשה יהיה מורכב משמות (לא מספרי ת"ז!!!) 2 המגישים ושני המגישים צריכים לעלות את העבודה למודל.

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. מבנה התכנית (הזחות) ותיעוד במידת הצורך.
4. חובה להשתמש בקבועים במקומות המתאימים.
5. יש להשתמש בפונקציות קצרות, כלליות, קריאות ושימושיות.
6. יש להקפיד על בדיקת תקינות קלט, אפשר להניח שאם ביקשו מהמשתמש מספר הוא הכניס מספר אך יתכן ולא בטווח הנכון.
7. הפלט צריך להיות כפי שניתן בתרגיל.
8. קוד קצר, לא מסורבל ויעיל הן מבחינת כתיבתו והן מבחינת ריצת התוכנית.

[Type here]

פירוט תרגיל בית 3

בתרגיל זה נמשיך לממש את מערכת ניהול חנות המזון מתרגיל 2.

חובה להמשיך מפתרון תרגיל בית 2 שפורסם במודל

רשימות מקושרות:

1. נתון קוד של רשימה מקושרת חד כיוונית השומר נתון int. עדכן אותה כך שתהיה כללית, כלומר ה Node שומר ערך מסוג void*. (חובה להשתמש בקוד המצורף!) הפונקציות הנדרשות:

1. איתחול רשימה
2. הוספת איבר לרשימה
3. מחיקת איבר מהרשימה – צור פונקציה כללית שתומכת במצב בו יש לשחרר את הערך ב Node ובמצב שאין צורך לשחרר את ערך זה.
4. שחרור כל הרשימה
5. הדפסת כל הרשימה

2. נשנה את המבנה של SuperMarket, כך שיחזיק את המוצרים השונים כרשימה מקושרת השומרת Product*. יש לדאוג שהמוצרים יהיו ממוינים לפי ברקוד. יש למחוק את המערך הקיים ואת המשתנה productCount!!!

אין למיין את הרשימה בשום שלב אלא להכניס את האיבר למקום הנכון עבורו, אין להשתמש במבנה עזר מכל סוג שהוא!

המבנה Customer

נוסיף למבנה Customer את השדות הבאים, כמה פעמים קנה בחנות המזון וסכום הקניות המצטבר שלו.

```
typedef struct
```

```
{
    char*      name;
    int         shopTimes;
    float       totalSpend;
    ShoppingCart* pCart;
}Customer;
```

יש לעדכן את פונקציות הבאות בהתאמה:

1. initCustomer (בהתחלה אין קניות כלל)
2. printCustomer
3. pay – יש לעדכן את השדות בהתאם לקניה.

חברת המזון

חברת המזון מאפשרת למיין את מערך הלקוחות שלה לפי השדות הבאים:

- שם
- מספר הפעמים שכל לקוח קנה בחנות המזון
- סך כל הקניות

על מנת שנדע לפי איזה שדה הלקוחות ממוינים נוסיף תכונה מתאימה לחברת המזון (enum שיכיל גם את האופציה שהמערך אינו ממוין). בהתחלה מערך הלקוחות לא ממוין כלל.

יש להשתמש בפונקציה qsort מ-stdlib על מנת לבצע את המיון.

יש להוסיף יכולת חיפוש לקוח במערך הלקוחות של חברת המזון ע"פ הפרמטר שלפיו הם ממוינים, יש להשתמש בחיפוש בינארי פונקציית הספרייה: bsearch, במידה והמערך אינו ממוין יש להודיע למשתמש שלא ניתן לבצע חיפוש.

[Type here]

קובץ binary:

חברת המזון שומרת את כל נתוניה למעט רשימת הלקוחות בקובץ binary הקובץ יישמר תחת השם "SuperMarket.bin"

כל מחרוזת דינאמיות char* ישמרו בפורמט הבא:

- מספר שמייצג את מספר התווים במחרוזת **כולל** '0\'
- המחרוזת עצמה כולל '0\'

פורמט כתיבת המוצרים בקובץ:

[שם חברת המזון]

[כתובת: מספר בית, שם רחוב, שם עיר]

[כמות מוצרים]

[מוצר 1]

[מוצר 2]

...

[מוצר n]

כאשר כל מוצר ישמר בפורמט הבא: (רמז – אין מצביעים)

[שם מוצר]

[ברקוד]

[סוג מוצר]

[מחיר]

[כמות]

לא נשמור את פרמטר המיין לקובץ, נאתחל את חברת המזון תמיד בהנחה שמערך הלקוחות לא ממזין.

קובץ text:

את נתוני הלקוחות חברת המזון שומרת בקובץ text בשם "Customers.txt"

פורמט כתיבת הלקוחות בקובץ text:

[כמות הלקוחות]

[לקוח 1]

[לקוח 2]

...

[לקוח n]

כאשר כל לקוח ישמר בפורמט הבא:

[שם לקוח]

[מספר קניות, סך כל ההוצאות המצטברות]

[Type here]

מצביעים לפונקציה

יש להכין פונקציה גנרית המטפלת במערך, `generalArrayFunction`. הפונקציה תקבל מערך, מספר אברים במערך, גודל כל איבר בבתיים וכן מצביע לפונקציה

```
void f(void* element);
```

הפונקציה `generalArrayFunction` עוברת על כל איברי המערך אחד אחר השני ושולחת את כתובת האיבר לפונקציה `f`.

יש להפעיל את הפונקציה הזו במקרים הבאים:

1. הדפסת מערך הלקוחות בפונקציה `printAllCustomers`
2. שיחרור הלקוחות בפונקציה `freeMarket`
3. שחרור המוצרים שנמצאים בעגלת הקניות בפונקציה `freeShoppingCart`

שינויים ב main

בתחילת התוכנית נאתחל את המערכת ע"י טעינת חברת המזון מהקבצים המתאימים. במידה והטעינה של נתונים אילו מהקובץ לא הצליחה נאתחל את חברת המזון מהמשתמש כפי שנעשה בתרגיל 2.

נוסף מספר אפשרויות לתפריט הראשי:

1. מיון הלקוחות לפי תכונה נתונה בסדר עולה (תפריט פנימי ע"פ ערכי ה-enum).
2. חיפוש לקוח לפי התכונה בה המערך ממוין

ביציאה מהתוכנית יש לשמור את נתוני חברת המזון לקבצים המתאימים.

בהצלחה!