

תרגיל מס' 5 – פוינטרים והקצאות דינמיות (ינן)

הוראות הגשה

- מועד אחרון להגשה :
- יש לשלוח את כל קבצי המקור (.h, c, makefile) באמצעות פקודת submitex לפני חלוף התאריך הנקוב לעיל. שם ההגשה: ptr5
- שמות הקבצים הנלווים לתוכנית יהיו utils.h, utils.c
- כל הודעות השגיאה (המוגדרות והלא מוגדרות) בתוכנית הם error.
- בתרגיל זה אינכם צריכים לבדוק את טיפוס הקלט, כלומר אם הנכם מצפים למספר או מתחייבים להכניס רק מספר.
- הנחיה כללית: כאשר אתם מתבקשים לקלוט מספר או אכן נכניס בטסטים מספר ולא אות.

התכנית

"מחסנית דו ראשית" היא מבנה נתונים אשר ניתן לאחסן בו נתונים. הצורה בה מנוהלת מחסנית זו היא: LIFO (Last In First Out) – אחרון שנכנס הוא הראשון שיוצא. הפעולות אשר נתמכות ע"י מחסנית הן:

- א. new – יצירת מחסנית.
- ב. push – הכנסת איבר למחסנית.
- ג. top – מיהו האיבר בראש המחסנית.
- ד. bottom – מיהו האיבר האחרון במחסנית.
- ה. pop – הוצאת איבר מהמחסנית (האיבר שבראש המחסנית).
- ו. popBottom – הוצאת איבר התחתון מהמחסנית והזזת כל האיברים מטה.
- ז. isEmpty – בדיקה אם המחסנית ריקה.

תרגיל 1 – מחסנית:

המחסנית תמומש ע"י מערך מסוג int בגודל קבוע 5 יש לממש את הפעולות הנתמכות ע"י המחסנית:

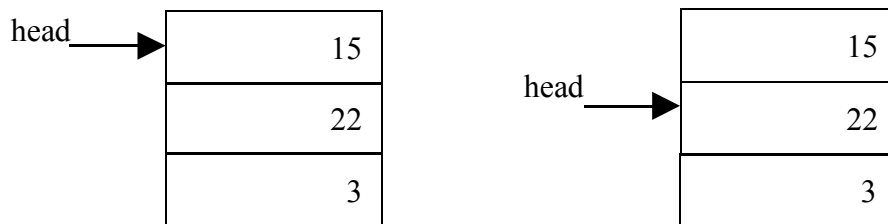
- א. new – יצירת מחסנית ריקה בגודל 5. // לא ניתן ליצור יותר ממחסנית אחת.
- ב. push(x) – הכנסת איבר x למחסנית.
- ג. top – החזר את ערך האיבר שבראש המחסנית (מבלי להוציאו מהמחסנית).
- ד. Bottom – החזר את ערך האיבר שבתחתית המחסנית (מבלי להוציאו מהמחסנית).
- ה. pop – הוצאת איבר מהמחסנית (האיבר שבראש המחסנית)--- האיבר האחרון שנכנס).
- ו. popBottom – הוצאת האיבר התחתון מהמחסנית (האיבר הראשון שהוכנס למחסנית) והזזת כל האיברים מקום אחד מטה.
- ז. isEmpty – מחזיר empty אם המחסנית ריקה ו-not empty אם לא ריקה.

חובה עליכם לבצע פעולות אילו ללא השימוש הישיר במיקום איברים במערך (כגון: array[3]) אלא באריתמטיקה של מצביעים (כגון: *(array+3) או ++arrayPointer) פרט לפעולה new בה אתם מגדירים את גודל המחסנית.

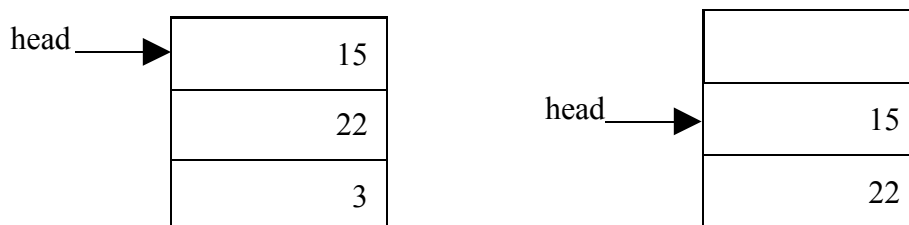
יש ליצור תפריט למשתמש אשר יענה לדרישות הבאות:

1. Create new stack
2. Push element to stack
3. What is in the head of the stack?
4. What is in the bottom of the stack?
5. Delete element from stack
6. Delete element from the bottom of the stack.
7. Stack status (empty/not empty)
8. exit

- פקודה 2 מקבלת ארגומנט (מקבלת איבר להוסיף למחסנית).
- במקרה שנבחרה פקודה 2-7 לפני 1 תהיה ההודעה error.
- במקרה ונבחרה פקודה 3-6 כשהמחסנית ריקה תהיה ההודעה error וכן כאשר נבחרה הפעולה 2 כאשר המחסנית מלאה.
- אם נבחרה אופציה מחוץ לטווח (1-8), תהיה הודעת error והתוכנית תמשיך להתבצע עד אשר תבחר האופציה 8 (יציאה).
- אם המשתמש בחר באופציה 1 כאשר כבר קיימת מחסנית תהיה ההודעה error (ולא תיווצר מחסנית נוספת).
- אפשרות 8 הינה יציאה מהתוכנית.
- לאחר כל הודעת error יש לחזור לתפריט הראשי.
- כאשר המשתמש בחר באופציה 5 (מחיקת איבר מהמחסנית), בפועל לא ימחק איבר מהמחסנית אלא המצביע של ראש המחסנית יצביע על איבר נמוך יותר. ראה תרשים לדוגמא:



- כאשר המשתמש בחר באופציה 6 (מחיקת איבר מתחתית המחסנית), האיבר התחתון ימחק לחלוטין והאיברים שמעליו כולל ראש המחסנית ירדו מקום אחד למטה לפי הסדר הנוכחי שלהם, ראה תרשים לדוגמא:



אין להדפיס את התפריט.

```

>> 3 // מיהו ראש המחסנית
error // לא קיימת עדיין מחסנית
>> 1 // (המשתמש בחר ליצור מחסנית (כמובן שגודל המחסנית 5
>> 2 3 // הוספת האיבר 3 למחסנית
>> 2 6 // הוספת האיבר 6 למחסנית
>> 2 8 // הוספת איבר 8 למחסנית
>> 2 9 // הוספת איבר 9 למחסנית
>> 2 10 // הוספת האיבר 10 למחסנית
>> 2 12 // ניסיון להוספת האיבר 12 למחסנית
error // האיבר האחרון (12) לא יתווסף--המחסנית מלאה
>> 1 // בקשה ליצירת מחסנית
error // לא ניתן ליצור יותר ממחסנית אחת
>> 3 // מהו האיבר שבראש המחסנית
10
>> 4 // מהו האיבר שבתחתית המחסנית
3
>> 7 // מהו מצב המחסנית
not empty
>> 5 // (מחיקת איבר מראש המחסנית (ז"א האיבר 10 נמחק)
>> 3 // מהו האיבר שבראש המחסנית
9
>> 7 // מהו מצב המחסנית
not empty
>> 2 98 // הוספת האיבר 98 למחסנית
>> 6 // הוצאת האיבר התחתון מהמחסנית
>> 4 // מהו האיבר שבתחתית המחסנית
6
>> 8 // יציאה מהתוכנית

```

תרגיל 2:

התוכנית הבאה גם משתמשת במחסנית אך בצורה משוכללת יותר. עליכם לממש מחסנית דינמית המחזיקה איברים מסוג `char *` (ראה למטה תפריט למשתמש). המתכנת יקבע את גודל המחסנית ל-2 בזמן יצירתה, ומכאן ולהבא יש לדאוג שתפוסתה לא תהיה יותר מ-90% ופחות מ-50% (פרט למקרה ההתחלתי בו המחסנית ריקה ותפוסתה 0%). לצורך כך, אם היא הגיעה לגבולות אלו (גדול מ-90% או קטן מ-50%) יש להקצות או לשחרר מקום כך שתהיה בתפוסה של **(קרוב ביותר** ל-70%). תפוסת המחסנית מתבטאת במספר התאים המלאים במחסנית ביחס למספר התאים המוקצים במחסנית.

בנוסף: בתוכנית הקודמת לא ניתן היה ליצור יותר ממחסנית אחת (במהלך התוכנית היה ניתן לבחור פעם אחת את אפשרות 1), כאן יהיה ניתן ליצור מספר לא מוגבל של מחסניות (בכל פעם שהמשתמש יבחר באפשרות מספר 1 תיווצר מחסנית חדשה.

נוסף את האפשרויות הבאות לתוכנית החדשה :

- אפשרות מספר 8 - מחיקת מחסנית מהזיכרון ע"י ציון מספר המחסנית אשר ברצוננו למחוק. אם המשתמש בחר למחוק את מחסנית מספר 3 (הפעולה תהיה 3 8 כפי שתראה

- בדוגמאות בהמשך) במידה והייתה קיימת מחסנית מספר 4 - אז לאחר פעולת המחיקה מחסנית מספר 4 תהפוך להיות מספר 3, 5 תהפוך 4 וכן הלאה. (אם המשתמש בחר למחוק מחסנית אשר לא קיימת, התוכנית תדפיס error ותחכה לקלט נוסף מהמשתמש – בחירת אופציה וכו'--)
- אפשרות מספר 9 - החזרת מספר המחסניות הקיימות.
 - אפשרות מספר 10 - החזרת תפוסת המחסנית באחוזים (מספר שלם אשר יעוגל תמיד כלפי מעלה ז"א אם תפוסתה 55.34 אחוזים יעוגל ל 56).
 - אפשרות מספר 11 – קליטת תו והחזרת סך כל התווים בכל המחסניות (תווי \0\ לא נספרים) ששויים לתו הזה. לדוגמא: בהנתן קלט 11 ואחריו את האות d, התוכנית תחזיר את מספר המופעים של האות d בכל האיברים שבכל המחסניות.
 - אפשרות מספר 12 – הדפסת כל המחסניות באותה שורה כאשר קיימת הפרדת רווח אחד בדיוק בין מחרוזות באותה המחסנית ורווח של שורה אחת בין מחסנית למחסנית (מחסנית ריקה תודפס כשורה ריקה).
 - אפשרות מספר 13 – יציאה מהתוכנית.

עבור אפשרויות 2,3,4,5,6,7,8,9,10 נצטרך לציין על איזה מחסנית (המספור של המחסניות מתחיל ב 1) ברצוננו לבצע את הפעולה (מספר המחסנית יהיה הקלט הראשון כמובן לאחר בחירת האופציה).

הדוגמה שלפניכם מפרטת את הפורמט הרצוי לפעולות המוגדרות. במקרה והמשתמש מבקש פעולה אשר לא תיתכן יש להדפיס error. התוכנית תמשיך להתבצע גם לאחר ה-error ותחכה לקלט חדש מהמשתמש (ז"א בחירה מחדש של אופציה מהתפריט). אין להדפיס את התפריט. כל השגיאות אשר הוגדרו בתרגיל הקודם תקפות גם כאן.

1. Create new stack.
2. Push element to stack.
3. Show the head of the stack.
4. Show the bottom of the stack
5. Deleting element from stack
6. Deleting element from bottom of the stack
7. Stack status
8. Delete stack
9. Number of existing stacks
10. Stack capacity
11. num of occurrences of a given letter
12. print stacks elements
13. exit

אם נבחרה אופציה מחוץ לטווח, תהיה הודעת error ותהיה אפשרות נוספת לבחירה עד ליציאה. בבחירה של exit ישוחרר כל המקום שהוקצה.

הבהרה: מחיקת איבר (אופציה מספר 4) אינה בהכרח מלווה במחיקתו מהזיכרון אלא בהזזת מצביע של ראש הרשימה. מחיקה מהזיכרון תיעשה אם ישנה חריגה מגבולות 50%-90%.

```

>> 1 //2 (הגודל יהיה 2)
>> 1 // (יצירת מחסנית נוספת (כעת, מחסנית מספר שתיים)
>> 2 2 aba // הכנסת הטקסט למחסנית והקצאת מקום בהתאם
>> 2 3 red // (כאן המשתמש מנסה להכניס את הטקסט למחסנית מספר 3 (מחסנית שלא קיימת
error // המערכת מדפיסה)
>> 1 // (יצירת מחסנית (נוצרה מחסנית מספר 3)
>> 8 2 // מחיקת מחסנית מספר 2 – כך גם נמחקים כל האיברים שהיו בה (לאחר הפעולה מחסנית//
מספר 3 הופכת להיות מספר 2 וכן אם הייתה מחסנית מספר 4 הייתה הופכת להיות מספר 3
(וכיו)
>> 2 3 a // (ניסיון להכניס את התו למחסנית 3 (אולם מחסנית 3 כבר לא קיימת)
error
>> 2 2 b // הכנסת התו למחסנית מספר 2
>> 2 2 father // הכנסת הטקסט למחסנית מספר 2. כאן התוכנית תצטרך להקצות מקום כך
שתפוסתה תהיה קרובה ל 70% (מאחר והמחסנית חורגת מגבול 90%-50%) לכן תצטרך להקצות
עוד איבר אחד ותפוסתה תהיה 66.66%
>> 2 2 mother // הכנסת הטקסט למחסנית מספר 2. גם כאן גודל המחסנית חורג מהגבולות לכן
התוכנית תצטרך להקצות מקום כך שתפוסתה תהיה קרובה ל 70% לכן תצטרך להקצות עוד
איבר אחד ותפוסתה תהיה 75%
>> 2 2 brother // הכנסת הטקסט למחסנית מספר 2. כאן התוכנית תצטרך להקצות מקום כך
שתפוסתה תהיה קרובה ל 70%. כאן תצטרך להקצות עוד שני איברים ותפוסתה: 66.66%
>> 10 2 // מהי תכולת המחסנית//
67%
>> 3 2 // (ללא הוצאתו) מיהו האיבר בראש מחסנית
brother
>> 5 2 // מחיקת איבר שבראש מחסנית 2 (כאן גם צריך לעדכן את גודל המחסנית עפ"י//
האחוזים---מאחר ובמחסנית זו לאחר מחיקה יש 3 איברים וגודלה הוא 6, לכן נעדכן את גודלה
(ל- 4 וכך תפוסתה תהיה 75%)
>> 7 2 // מהו מצב המחסנית
not empty
>> 7 1 // מהו מצב מחסנית
empty
>> 3 1 // מהו האיבר בראש מחסנית
error // המחסנית ריקה)
>> 3 2 // מהו האיבר בראש מחסנית
mother
>> 9 // מחזיר מספר מחסניות קיימות//
2
>> 10 1 // מהי תכולת מחסנית 1
0% // אחרת התכולה תהיה בין 50%-90%
>> 10 2 // מהי תכולת מחסנית 2
75%
>> 11 r // מהו מספר המופעים של האות בכל המחסניות//
2
>> 12

```

b father mother

>> 13 // סיום התוכנית