# logo

# מבני נתונים 61104

סמסטר קיץ תשע"ח

**תרגיל הגשה מס' 2 (להרצה)**

**שאלה 1 (50 נקודות)**

יש לכתוב פונקציה המשתמשת בטיפוס הנתונים המופשט **מחסנית** כדי לקמפל ביטוי אריתמטי פשוט ללא סוגריים. לדוגמא, ביטוי הקלט

A + B \* C – D ^ 2

יתקמפל לטבלה

Operation Operand1 Operand2 Result

\* B C Z

+ A Z Y

^ D 2 X

– Y X W

הטבלה מראה את הסדר שעל-פיו הפעולות מתבצעות לצורך חישוב הביטוי, (\* , + , ^ , , – מימין לשמאל) והאופרנדים לכל הפעולה. עמוד התוצאה מגדיר שם המזהה (החל מ- Z) שנבחר לשמור על כל התוצאה. ניתן להניח כי האופרנדים המשתתפים הם הספרות 0 עד 9 והאותיות A עד H, והפעולות הן / , ^, \* , – ,+. הקדימות של ^ היא הגבוהה ביותר, אחר כך הקדימות של \*,/ והקדימות של +,– היא הנמוכה ביותר.

האלגוריתם יהיה מאורגן כדלקמן. אנו יוצרים שתי מחסניות: מחסנית האופרנדים s1 ומחסנית הפעולות s2. יש לקלוט את תווי הביטוי אחד אחרי השני (או להגדיר את הביטוי כמחרוזת קלט של הפונקציה). אם התו הוא אופרנד יש להכניס אותו ל-s1. אם התו הוא פעולה יש להשוות הקדימות שלה עם קדימות הפעולה הנמצאת בראש s2.

אם הפעולה החדשה היא בעלת קדימות גבוהה מזו הנמצאת בראש s2 או ש-s2 ריקה ,אז הפעולה החדשה תיכנס ל-s2.

אם הפעולה החדשה היא בעלת קדימות נמוכה או שווה לזו שנמצאת בראש s2, יש להוציא את הפעולה שנמצאת בראש s2, להוציא שני אופרנדים מ-s1 ולכתוב השורה החדשה בטבלה. התו הנבחר לשמירת התוצאה ייכנס אחרי כן ל-s1. לאחר מכן יש להמשיך לבדוק את קדימות הפעולה שלנו מול קדימות הפעולה הנמצאת בראש s2. יש להמשיך ליצור שורות בטבלה עד שקדימות הפעולה בראש s2 תהיה נמוכה יותר מקדימות הפעולה שברשותנו, או עד ש-s2 תתרוקן. בשלב זה (עבור שני המצבים האלו) יש להכניס את הפעולה שלנו ל-s2 ולטפל בתו הבא. לאחר שסיימנו לעבור על כל סדרת התווים יש להוציא את כל הפעולות שנשארו ב-s2 יחד עם זוגות האופרנדים שלהן מ-s1 כמו הוסבר למעלה. לא לשכוח להכניס את תו התוצאה ל-s1 אחרי יצירת שורה בטבלה!

יש להתעלם מרווחים. אם התו הוא לא אופרנד, לא פעולה, ולא רווח יש להציג את הודעת שגיאה ולסיים את ביצוע האלגוריתם.

למשל, לביטוי הנ"ל האלגוריתם יעבוד כדלקמן.

1. תו A נכנס ל-s1.
2. תו + נכנס ל-s2.
3. תו B נכנס ל-s1.
4. תו \* נכנס ל-s2.
5. תו C נכנס ל-s1.
6. תו – מחכה. תו \* יוצא מ-s2. תווי C ו-B יוצאים מ-s1. B\*C=Z (השורה החדשה בטבלה). תו Z נכנס ל-s1.
7. תו – מחכה. תו + יוצא מ-s2. תווי Z ו-A יוצאים מ-s1. A+Z=Y (השורה החדשה בטבלה). תו Y נכנס ל-s1.
8. תו – נכנס ל-s2.
9. תו D נכנס ל-s1.
10. תו ^ נכנס ל-s2.
11. תו 2 נכנס ל-s1.
12. תו ^ יוצא מ-s2. תווי 2 ו-D יוצאים מ-s1. D^2=X (השורה החדשה בטבלה). תו X נכנס ל-s1.
13. תו – יוצא מ-s2. תווי X ו-Y יוצאים מ-s1. Y–X=W (השורה החדשה בטבלה). תו W נכנס ל-s1.
14. תו W יוצא מ-s1.

ניתן להניח שפעולת חזקה שמאלית מתבצעת לפני פעולת חזקה ימנית ז"א A^B^C מתקמפלת כי (A^B)^C למרות שלפי כללי המתמטיקה .

**בונוס (עד 10 נקודות נוספות):** יש לבצע פעולת חזקה ימנית לפני פעולת חזקה שמאלית ז"א לפרש A^B^C כ- A^(B^C).

יש להשתמש בספרייה למימוש המחסנית (נא לצרף את קבצי המימוש).

יש לכתוב פונקציה ראשית המפעילה את הפונקציה הנ"ל.

**שימו לב:** הפונקציה לא מחשבת כלום. היא רק מציגה את הטבלה.

**שאלה 2 (50 נקודות)**

יש לממש טיפוס מופשט Polynomial כסוג של **רשימה מקושרת דו-כיוונית**. כל איבר של רשימה זו כולל שני נתונים: מקדם (ממשי שונה מ-0) וחזקה (שלם אי-שלילי). על איברי הרשימה להיות ממוינים בסדר עולה ממש לפי ערכי החזקות .

למשל, הפולינום 5-x2+4x3+8x6 יהיה ייוצג ע"י הרשימה הבאה:

5

0

-1

2

4

3

8

6

יש להגדיר את הפונקציות (הפעולות) הבאות ל- Polynomial:

1. פונקציה ליצירת פולינום ריק.

הפונקציה תממש פולינום ריק כרשימה ריקה. יש לפרש את הפולינום כמספר 0.

1. פונקציה להכנסת איבר חדש לפולינום.

הפונקציה תקבל פרמטרים מסוג Polynomial וזוג מספרים (מקדם וחזקה) ותכניס את האיבר החדש במקום המתאים (האיברים יהיו ממוינים לפי ערכי החזקות גם אחרי ההכנסה). אם נכנס איבר בעל חזקה ששייכת כבר לפולינום, לאחד את האיברים בעלי אותה חזקה ע"י חיבור המקדמים שלהם.

לדוגמא, הכנסת איבר 2x3 לפולינום

5-x2+4x3+8x6

הופכת את הפולינום ל-

5-x2+6x3+8x6.

1. פונקציה לחיבור שני פולינומים.

הפונקציה תקבל שני פרמטרים P1 ו-P2 מסוג Polynomial ותיצור פולינום חדש המהווה סכום של שני הפולינומים שהועברו כפרמטרים.

לדוגמא, הסכום של הפולינומים

5-x2+4x3+8x6

ו-

3x+5x2-4x3+x5-2x6+7x8

הוא

5+3x+4x2+x5+6x6+7x8.

אם P2=-P1 הפונקציה תיצור פולינום ריק.

1. פונקציה לחיסור שני פולינומים (באופן דומה). אם P2=P1 הפונקציה תיצור פולינום ריק.
2. פונקציה להכפלת פולינום במספר.

הפונקציה תקבל פרמטרים מסוג Polynomial ומספר ממשי C ותיצור פולינום חדש המהווה מכפלת הפולינום המקורי ב-C.

לדוגמא, לפולינום

5-x2+4x3+8x6

ו- C=3

הפולינום החדש יהיה

15-3x2+12x3+24x6

אם C=0 הפונקציה תיצור פולינום ריק.

1. פונקציה שמחזירה סדר גודל הפולינום.

לדוגמא, לפולינום

5-x2+4x3+8x6

יוחזר 6.

לפולינום ריק הפונקציה תחזיר -1.

1. פונקציה לאיפוס פולינום.

הפונקציה תקבל פולינום ותהפוך אותו לפולינום ריק.

1. פונקציה להדפסת פולינום.

הפונקציה תקבל פולינום ותדפיס את האיברים שלו החל מחזקה המשמעותית ביותר בצורה הרגילה (לא תציג מקדם 1, תציג מקדם -1 רק באמצעות הסימן "-", לא תציג סימן "+" אם לפני האיבר הראשון יש מקדם חיובי, לאיבר בעל חזקת 0 הפונקציה תציג רק את המקדם, לאיבר בעל חזקת 1 הפונקציה לא תציג את החזקה).

לדוגמא, הפולינום שבציור לעיל יוצג כ-

8x^6+4x^3-x^2+5

הפונקציה תציג פולינום ריק כ-

**Zero (Empty Polynomial)**

1. **בונוס (עד 10 נקודות נוספות):** פונקציה להכפלת שני פולינומים.

הפונקציה תקבל שני פרמטרים מסוג Polynomial ותיצור פולינום חדש המהווה מכפלה של שני הפולינומים שהועברו כפרמטרים.

לדוגמא, המכפלה של הפולינומים

5-x2+4x3

ו-

3x+5x2-4x3

היא

15x+25x2-23x3+7x4+24x5-16x6.

על הפונקציות שיוצרות פולינום חדש להחזיר (להעביר) את התוצאה.

יש להשתמש בפעולות בסיסיות על רשימות מקושרות.

טיפוס Polynomial צריך להיות מיושם בקבצים (c ו-h) מיוחדים.

יש לכתוב פונקציה ראשית בעלת **ממשק ידידותי** **ככל האפשר** להפעלה ובדיקה של כל הפעולות הנ"ל.

על הפונקציה הראשית לכלול קלט פרמטרים, בניית פולינומים, הצגת תוצאות וכו'. ניתן להשתמש בפונקציות עזר.

**יש לאחד את השאלות 1 ו-2 לתכנית אחת באמצעות תפריט. בשאלה 2 אפשר לארגן תפריט פנימי.**

**יש להציג את כל התוצאות.**

**יש להשתמש בשמות משמעותיים וגם בהערות.**

**יש להקפיד לכתוב בצורה מבנית.**

**יש להקפיד על שימוש בממשק ידידותי ככל האפשר.**

**תכנית שלא עוברת קומפילציה לא תתקבל!**