

המכללה הטכנולוגית להנדסה

הדסה נעורים

המחלקה לתוכנה

פרויקט 2 בתכנות מערכות בשפת C

לתלמידי כיתת תוכנה – יג

מרצים: ד"ר ראובן חוטובלי
גב' יונה סעדיה

Polynoms (in C)

תרגיל 1

בתרגיל זה, עליכם לממש סכום פולינומים, מכפלת פולינומים וגזירה של פולינום שהתקבל ממכפלת פולינומים.
פולינום ייוצג כאוסף של זוגות: (חזקה, מקדם). למשל, הפולינום: $3x^{22}+7x^3-99x^{100}+101x^0$, ייוצג כ: 3 22 7 3 -99 100 101 0 . יש לייצג את הפולינום כרשימה מקושרת ממוינת לפי חזקות בסדר יורד.

יש להדפיס למשתמש הודעה:

Please enter polynomial

לקלוט את המחרוזת הראשונה, ושוב, יש להדפיס למשתמש הודעה:

Please enter polynomial

לקלוט את המחרוזת השנייה.

כפלט, יש להדפיס את סכום ואת מכפלת הפולינומים המצומצמת (המונום $0x^k$, עבור k כלשהוא, שווה 0 ולכן לא יודפס) מן החזקה הגבוהה לנמוכה.

לאחר מכן, יש להדפיס למשתמש:

Please enter derivative depth

כפלט, יש להדפיס את כל הנגזרות, עד למספר המבוקש.

דוגמה:

הפולינום הראשון: $10x^7-9x^5+3x^1$

הפולינום השני: $1x^2+2x^0$

תוצאת הסכום היא: $10x^7-9x^5+1x^2+3x^1+2x^0$

תוצאת המכפלה היא: $10x^9+11x^7-18x^5+3x^3+6x^1$

תוצאת הגזירה הראשונה: $90x^8+77x^6-90x^4+9x^2+6x^0$

תוצאת הגזירה השנייה: $720x^7+462x^5-360x^3+18x^1$

ניתן להניח כי:

Please enter polynomial

10 7 -9 5 3 1

Please enter polynomial

1 2 2 0

10 7 -9 5 1 2 3 1 2 0

10 9 11 7 -18 5 3 3 6 1

Please enter derivative depth

2

90 8 77 6 -90 4 9 2 6 0

720 7 462 5 -360 3 18 1

- הקלטים יהיו מספרים שלמים (יתכן מקדמים שלילים).
- הקלט יהיה תקין, כלומר מספר זוגי של מספרים (כל מונם מצריך זוג מספרים שיתאר אותו).
- כל פולינום יכיל לפחות מונם אחד (מונם = זוג מספרים = מקדם וחזקה).
- החזקה המינימלית האפשרית תהיה 0
- מספר הגזירות המבוקש יהיה מספר שלם, גדול מ-0 אך קטן מ-3.

תרגיל 2

Bulk (in C)

עליכם לממש משחק תפזורת.

יש לקלוט מהמשתמש מטריצה בגודל 10X10 המכילה תווים ומילה ולהדפיס את מיקום האות הראשונה במילה. מילה יכולה להימצא בתפזורת בכל כיוון וגם כשהיא הפוכה. הקלט יתקבל באופן הבא: יש להדפיס למשתמש הודעה לקבלת המטריצה:

Please enter the Matrix

כעת יש לקלוט את המטריצה ולהדפיס למשתמש בקשה לקבלת המילה:

Please enter the word to search

בסיום החיפוש יש להדפיס למשתמש את המיקום שבו נמצאה האות הראשונה של המילה.

לדוגמא:

Please enter the Matrix

a	b	c	o	d	e	f	g	h	i
j	k	l	m	l	n	o	p	q	r
s	t	r	z	a	l	q	w	s	x
c	d	e	r	f	v	e	b	g	t
y	h	y	u	j	m	k	h	i	l
o	p	p	l	o	i	k	m	j	z
x	c	v	b	n	m	a	s	f	g
h	j	k	l	q	w	r	t	y	u
o	p	q	z	z	b	n	h	y	t
r	e	k	l	j	g	h	r	s	a

Please enter the word to search

Hello

[4][7]

שימו לב:

- במידה והמילה לא קיימת בתפזורת יש להדפיס: Not Found.
- במידה והמילה קיימת יותר מפעם אחת יש להדפיס את המיקום הקטן יותר, ראשית לפי מספר השורה ואז לפי מספר העמודה.
- אין משמעות לאותיות גדולות וקטנות.
- אנא התבוננו בקבצי הקלט-פלט לדוגמא בכדי לראות את תצורת הקלט.
- כפלט, אין להדפיס את המטריצה עם העיגול על המילה שנמצאה אלא רק את שורת ההדפסה: [4][7], המסמלים את מיקום האות הראשונה של המילה במטריצה.
- נקודה למחשבה: האם יש טעם להתחיל בחיפוש אם המילה לחיפוש אורכה גדול מ-10?

תרגיל 3

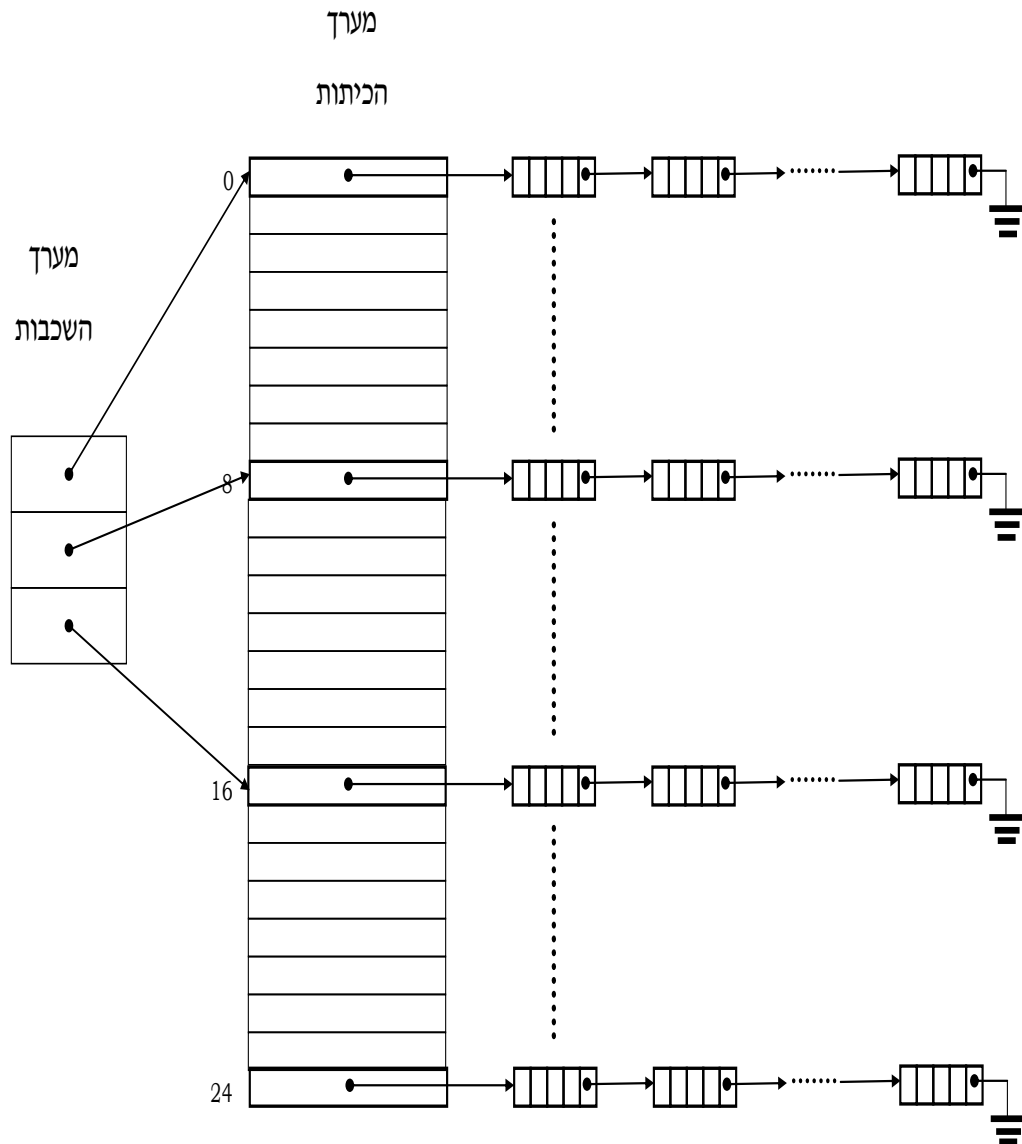
מטרתנו לכתוב תכנית אשר מטפלת בבית ספר נתון.
בית הספר מכיל 24 כיתות המחולקות ל 3 שכבות גיל :

- שכבה 0 : המכילה 8 כיתות י
- שכבה 1 : המכילה 8 כיתות יא
- שכבה 2 : המכילה 8 כיתות יב .

נתון קובץ בינארי Grades.dat המכיל רשומות תלמידים במבנה הבא :

- מספר שכבה
- כיתה
- שם תלמיד
- טלפון
- ציון ממוצע של תלמיד

לפניך מבנה הנתונים בו מאחסנים את נתוני התלמידים הלומדים בבית הספר הנתון.



איור 1 – מבנה הנתונים

המבנה המתואר לעיל מכיל :

- מעריך השכבות של 3 איברים המיצג את 3 השכבות בבית הספר.
- כל תא במעריך השכבות מכיל מצביע למעריך הכיתות של 8 איברים המיצג את הכיתות באותה השכבה.
- כל תא במעריך הכיתות מכיל אף הוא מצביע לרשימה מקושרת המייצגת את קבוצת התלמידים הלומדים באותה כיתה.
- כל צומת ברשימת התלמידים מכיל את הפרטים הבאים :
 - שם תלמיד
 - טלפון
 - ציון ממוצע של תלמיד
 - מצביע לרשימת התלמיד הבא (אם קיים תלמיד נוסף שנמצא אחריו ברשימה) , אחרת מצביע ל- NULL .

הערות :

- שלושת השכבות בבית ספר הנתון מיוצגות באמצעות מערך חד ממדי בשם school .
- כל הכיתות בבית ספר זה מיוצגות באמצעות מערך חד ממדי אחד בשם classes .
- במערך זה האינדקסים מ-0 עד 7 (כולל) מייצגים את השכבה 0 , האינדקסים מ-8 עד 15 (כולל) מייצגים את השכבה 1 , והאינדקסים מ-16 עד 23 (כולל) מייצגים את השכבה 2 .
- בכל אחת מן הרשימות המקושרות רשומת התלמיד הראשון ברשימה מייצגת תלמיד בעל הציון הממוצע הגבוה ביותר מבין כל הלומדים בכיתתו. יתר הרשומות ברשימה אינן מסודרות (לפי סדר עולה או יורד) בהכרח.

הנח כי אין שני תלמידים באותה כיתה שיש להם ציון הכי גבוה.

נניח שרשומת התלמיד הראשון ברשימה מסוימת מייצגת תלמיד בעל ציון ממוצע – X .
אם נוסיף לרשימה זו רשומת תלמיד חדש בעל ציון ממוצע- Y אז נבצע את הבדיקה הבאה :

אם $(X < Y)$ אז נוסיף את רשומת התלמיד החדש לראש רשימה זו (חשוב מדוע!)
אחרת נוסיף את רשומת התלמיד החדש כאיבר שני ברשימה זו.

התוכנית מבצעת את הצעדים הבאים :

צעד 1: אתחול מבנה הנתונים שתואר לעיל.

צעד 2: קריאת נתונים מהקובץ הבינארי Grades.dat, ואחסון במבנה הנתונים שתואר לעיל.
קריאת הרשומות (מבנים) מהקובץ הבינארי נעשית באופן סידרתי , החל מהמבנה הראשון עד וכולל המבנה האחרון.

צעד 3:

התוכנית מציגה את התפריט הבא :

1. מציאת הציון השכיח בבית הספר.
2. יצירה ובניית מערך דינאמי ממוין של תלמידים מצטיינים בשכבה מסוימת נתונה. מיון המערך הזה נעשה באמצעות הפונקציה qsort (פונקציה שקיימת בשפת C). **תלמיד נקרא מצטיין אם הציון הממוצע שלו 90 לפחות.**
3. חיפוש בינארי במערך הדינאמי של התלמידים המצטיינים בשכבה מסוימת נתונה. החיפוש נעשה באמצעות הפונקציה bsearch (פונקציה שקיימת בשפת C).
4. מציאת התלמיד המצטיין בשכבה מסוימת ונתונה.
5. יצירת קובץ תלמידים מצטיינים בשכבה מסוימת אשר ממוין לפי שם התלמיד.
6. יציאה .

לפניך הכרזות והגדרות בשפת C בקובץ Grades.h

```

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#define LEVELS 3 /* מספר שכבות */
#define CLASSES_IN_LEVEL 8 /* מספר כיתות לשכבה */
typedef struct pupilType /* טיפוס צומת בעבור רשומת תלמיד בשימה */
{
    char lname[24]; /* שם משפחה */
    char fname[12]; /* שם פרטי */
    char tel[12]; /* מספר טלפון */
    int grade ; /* ציון ממוצע */
    struct pupilType *next; /* מצביע למבנה הבא */
} pupil,*pupilPtr;

typedef struct bufferType /* טיפוס של רשומת תלמיד בקובץ בינארי */
{
    int level; /* מספר שכבה */
    int clas; /* מספר כיתה */
    char lname[24]; /* שם משפחה */
    char fname[12]; /* שם פרטי */
    char tel[12]; /* מספר טלפון */
    int grade ; /* ציון ממוצע */
} buffer,*bufferPtr;

typedef struct classType /* טיפוס נתון כיתה */
{
    struct pupilType *first; /* מצביע לראש רשימת התלמידים הלומדים בכיתה זו */
} class,*classPtr;

```

לפניך הכרזות והגדרות בשפת C בקובץ Grades.c

```

#include "Grades.h"
int n ;
pupilPtr *classes; /* מערך כל הכיתות בבית הספר הנתון */
pupilPtr **school; /* מערך של שכבות */
pupilPtr pupils; /* לצורך מערך דינאמי */

```

א. כתוב שגרה שכותרתה `void createStructure(void)` אשר יוצרת ומאתחלת את מבנה הנתונים כמתואר **בצעד 1**.

ב. כתוב שגרה שכותרתה `void insertGrade(bufferPtr buff)` אשר מקבלת כפרמטר מצביע (`buff`) על **רשומת תלמיד מהקובץ הבינארי** `Grades.dat` ומאחסנת את פרטי התלמיד במבנה הנתונים, כמפורט בפתיח השאלה **בצעד 2**, **כרשומת תלמיד ברשימה**.

ג. כתוב שגרה שכותרתה `void populateStructure(void)` הקוראת נתונים מהקובץ **הבינארי** `Grades.dat` ומאחסנת אותם כמפורט בפתיח השאלה **בצעד 2**.

ד. כתוב שגרה שכותרתה `void diligentPupils(int level)` אשר מקבלת כפרמטר שכבה – `level`. השגרה יוצרת ובונה **מערך דינאמי** של תלמידים מצטיינים בשכבה הנתונה. בנוסף השגרה ממיינת את המערך **הדינאמי הזה**, באמצעות הפונקציה `qsort` (פונקציה שקיימת בשפת C) כנדרש **בצעד 3.2**.

ה. כתוב שגרה שכותרתה `void electedPupil(int level)` אשר מקבלת כפרמטר שכבה – `level`. השגרה מוצאת ומדפיסה את פרטי התלמיד המצטיין בשכבה נתונה.
ו. כתוב שגרה שכותרתה `int frequentGrade(void)` אשר מוצאת ומחזירה את הציון השכיח בבית הספר.

ז. כתוב שגרה שכותרתה `void saveArray()` אשר יוצרת קובץ `pupils.dat` של תלמידים מצטיינים בשכבה מסוימת כאשר הקובץ ממוין לפי שם התלמיד.
שים לב: הינך רשאי להשתמש בפונקציות שבסעיפים הקודמים או במשתנה גלובלי שנבנה בסעיפים הקודמים.

ח. לפניך שגרה `mainProcess` אשר מציגה את התפריט כמפורט ב**צעד 2** בפתיח השאלה. כל משימה תבוצע בהתאם לבחירת האופציה שבתפריט.
שים לב: הינך רשאי להשתמש בפונקציות שבסעיפים הקודמים או במשתנה גלובלי שנבנה בסעיפים הקודמים.

```
void mainProcess()
{
    int reply = 1;
    int level;
    int maxfreq;
    char buf[80];
    pupilPtr ptr;

    while (reply != 6)
    { /* main Menu */
        puts("\n\n 1. מציאת הציון השכיח ");
```

```

puts("\n    2. מציאת תלמידים מצטיינים בשכבה מסוימת והצגתם בצורה ממוינת ");
puts("\n    3. חיפוש בינארי בקרב תלמידים מצטיינים ");
puts("\n    4. מציאת התלמיד המצטיין בשכבה ");
puts("\n    5. יצירת קובץ ממוין של תלמידים מצטיינים בשכבה ");
puts("\n    6. יציאה ");
printf("\n\n\n Your choice: ");
scanf("%d", &reply);
switch(reply)
{
    case 1:
        maxfreq = ____ (1) ____ ;
        printf("\nGrade with Max frequency is %d ",maxfreq);
        break ;
    case 2: .....
    case 3:.....
    case 4:.....
    case 5: .....
    case 6: .....
    default:.....
}
}
getchar();
return 0;
}

```