



מכון טכנולוגי חולון
Holon Institute of Technology

מבחן סדנה מתקדמת בתכנות - 61108

מרצים: ד"ר מרק קורנבליט, מר טוביה דב רוזנברג

סמסטר ב', תשפ"ג

מועד א', 8.06.2023

מדבקת הבחינה

- משך הבחינה: 3 שעות.
- אין להשתמש בחומרי עזר, פרט לדף המצורף לשאלון.
- לא יחולקו מחברות. יש לכתוב הכל בשאלון הבחינה בלבד, כולל טיוטה.
- **יש לכתוב תשובות על גבי טופס המבחן במקום המוקצה לכך.**
- 6 דפים אחרונים מיועדים לטייטה.
- בזמן הבחינה אסור להפריד או לתלוש את דפי הבחינה. בנוסף, בזמן הבחינה אסור לפרק ולשדרך מחדש את דפי הבחינה בעזרת שדכן סיכות. אי עמידה בהוראות אלו תגרור את פסילת הבחינה כולה.
- הבמבחן מורכב מ-4 חלקים:
 - בחלק 1 יש שאלה מס' 1.
 - בחלק 2 יש שאלות מס' 2 ומס' 3. יש לענות על שאלה אחת מהן.
 - בחלק 3 יש שאלות מס' 4 ומס' 5.
 - בחלק 4 יש שאלה מס' 6 – בונוס.

בהצלחה!

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel
www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences
Department of Computer Science

חלק 1.

שאלה 1 (40 נק')

בפולינום $P(x) = \sum_{i=0}^{n-1} a_i x^i$ כל איבר $a_i x^i$ (a_i הוא מקדם ממשי) נקרא **מונום** i .

נגדיר מבנה הבא המייצג מונום:

```
typedef struct monom {  
    double coefficient;  
    int power;  
} Monom;
```

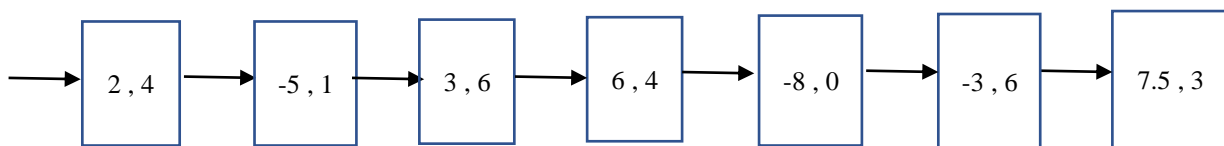
וגם נגדיר מבנה המייצג איבר ברשימה מקושרת:

```
typedef struct item {  
    Monom mon;  
    struct item * next;  
} Item;
```

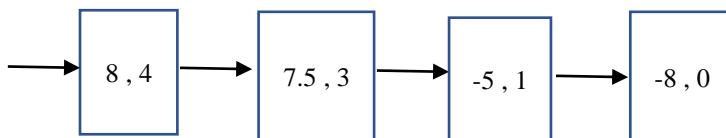
כתבו פונקציה ***Item * createPolynom(Item * L)*** המקבלת רשימה מקושרת חד-כיוונית של איברים מטיפוס **Item**.

על הפונקציה ליצור ולהחזיר רשימה חדשה **P** (של איברים מטיפוס **Item**) המייצגת פולינום כך שאיברי הרשימה יהיו ממוינים לפי חזקות המונומים שהם מכילים (עבור $i < j$, מונום j יהיה יותר קרוב לראש הרשימה מאשר מונום i). רשימת **P** תיווצר ע"י **סיכום המונומים** המרכיבים את **L**. ב-**P** לא יהיו מונומים עם מקדמים אפס. הרשימה הריקה (המקורית והחדשה) תייצג מספר 0.

לדוגמא, עבור הרשימה **L** הבאה (המספר השמאלי הוא מקדם והמספר הימני הוא חזקה):



הפונקציה תיצור את הרשימה **P** הבאה:



אשר מייצגת את הפולינום $8x^4 + 7.5x^3 - 5x - 8$.

שימו לב. כדי לייעל את העיבוד של הרשימה המקורית יש להשתמש במערך עזר שגודלו תלוי בחזקה המקסימלית ברשימה הזאת. זוהי דרישה מחייבת!

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```
Item* createPolynom(Item* L)
{
    Item* curr = L, * P = NULL;
    int max_power = 0, power;
    double *arr;

    /* Search of maximum power */
    while (curr)
    {
        if (curr->mon.power > max_power)
            max_power = curr->mon.power;
        curr = curr->next;
    }

    /* Calculation of coefficients */
    arr = (double*)calloc(max_power + 1, sizeof(double));
    curr = L;
    while (curr)
    {
        arr[curr->mon.power] += curr->mon.coefficient;
        curr = curr->next;
    }

    /* Creation of polynomial list */
    curr = L;
    for (power = 0; power <= max_power; power++)
        if (arr[power] != 0)
        {
            curr = (Item*)malloc(sizeof(Item));
            curr->next = P;
            P = curr;
            P->mon.coefficient = arr[power];
            P->mon.power = power;
        }

    free(arr);
    return P;
}
```


חלק 2. בחלק זה יש לענות על שאלה אחת מתוך שתיים.

שאלה 2 (40 נק')

כתבו פונקציה יעילה בשם `sortedRowsCols` המקבלת מערך דו-ממדי דינאמי `ar` של מספרים שלמים בעל `m` שורות ו-`n` עמודות. על הפונקציה ליצור שני מערכים חד-ממדיים. אחד מהם (בשם `ar_s_rows`) יהיה מורכב ממספרי השורות במערך הדו-ממדי **ממוינות בסדר עולה ממש** (בשם `ar_s_cols`) יהיה מורכב ממספרי העמודות **ממוינות בסדר יורד ממש**. הפונקציה תעביר (by reference) את המערכים החדשים ואת הגדלים שלהם. במידה ומערך נוצר ריק, הפונקציה תעביר NULL ככתבתו ו-0 כגודלו.

לדוגמא:

עבור המערך `ar` (משמאל לימין, מלמעלה למטה)

10	13	56	99
23	57	59	93
36	19	74	89
30	75	80	85
25	73	84	84

הפונקציה תעביר מערך `ar_s_rows`: {0,1,3} עם גודלו 3 ומערך `ar_s_cols`: {3} עם גודלו 1.

החתימה של הפונקציה `sortedRowsCols` תהיה:

```
void sortedRowsCols(int **ar, int m, int n,  
                    int **ar_s_rows, int *num_s_rows, int **ar_s_cols, int *num_s_cols)
```

לבדיקה האם השורה ממוינת יש להשתמש בפונקציית עזר `int arrSorted(int *a, int n)` המקבלת את השורה כמערך `a` בגודל `n`. הפונקציה תחזיר 1 אם המערך ממוין בסדר עולה ממש ו-0 אחרת.

לבדיקה האם העמודה ממוינת יש להשתמש בפונקציית עזר `columnSorted`. הפונקציה תקבל מערך דו-ממדי `ar`, מספר שורותיו `m`, ומספר העמודה הנבדקת `col`. על הפונקציה להחזיר 1 אם העמודה ממוינת בסדר יורד ממש ו-0 אחרת.

חתימת הפונקציה תהיה `int columnSorted(int **ar, int m, int col)`.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```
int arrSorted(int* a, int n)
{
    int i;
    for (i = 1; i < n; i++)
        if (a[i] <= a[i - 1])
            return 0;
    return 1;
}

int columnSorted(int** ar, int m, int col)
{
    int i;
    for (i = 1; i < m; i++)
        if (ar[i][col] >= ar[i - 1][col])
            return 0;
    return 1;
}

void sortedRowsCols(int** ar, int m, int n, int** ar_s_rows, int* num_s_rows, int** ar_s_cols, int* num_s_cols)
{
    int i, j;

    *num_s_rows = *num_s_cols = 0;
    *ar_s_rows = (int*)calloc(m, sizeof(int)); // max size - number of rows
    *ar_s_cols = (int*)calloc(n, sizeof(int)); // max size - number of columns

    for (i = 0; i < m; i++)
        if (arrSorted(ar[i], n))
            (*ar_s_rows)[(*num_s_rows)++] = i;

    for (j = 0; j < n; j++)
        if (columnSorted(ar, m, j))
            (*ar_s_cols)[(*num_s_cols)++] = j;

    *ar_s_rows = (int*)realloc(*ar_s_rows, *num_s_rows * sizeof(int));
    *ar_s_cols = (int*)realloc(*ar_s_cols, *num_s_cols * sizeof(int));
}
```


שאלה 3 (40 נק')

נגדיר כמה תווים מסוימים כ**מפרידים**. במיוחד, תווי פסיק, נקודה, רווח, סימן קריאה, סימן שאלה, נקודתיים וכו' ניתן להגדיר כמפרידים.
מילה היא רצף של תווים שאינם מפרידים.
משפט הוא מחרוזת המכילה שרשור של מילים ומפרידים. המילים יכולות להיות מופרדות ע"י מפריד אחד או יותר. משפט לא ריק תמיד מתחיל במילה.

בכל השאלות הבאות **אסור להשתמש** בפונקציה סטנדרטית strtok.

סעיף א (15 נק')

כתבו פונקציה עם החתימה

char * nextWord(char * sentence, char * separators, char * word)

אשר מקבלת מחרוזת sentence המהווה את המשפט ומחרוזת separators המורכבת מכל המפרידים שעשויים להופיע במשפט.
על הפונקציה לעדכן את word להיות המחרוזת המכילה העתק של המילה הראשונה ב-sentence המוגדרת עד למפריד הראשון (ניתן להניח שגודל של מערך word יהיה מספיק).
הפונקציה תחזיר מצביע לתחילת המילה הבאה ב-sentence או NULL אם המשפט לא מכיל מילים נוספות.

לדוגמא:

עבור sentence = "Why?! That's why." ו-separators = "? !."
תוכן של word יהיה "Why" והפונקציה תחזיר מצביע לתת-מחרוזת "That's why." מתוך sentence.


```
char* nextWord(char* sentence, char* separators, char* word)
{
    int i = 0;
    while ( sentence[i] && !strchr(separators, sentence[i]) ) // scan word
        word[i++] = sentence[i];
    word[i] = '\0';
    while (sentence[i] && strchr(separators, sentence[i])) // scan separators
        i++;
    return sentence[i] ? sentence + i : NULL;
}
```

סעיף ב (15 נק')

כתבו פונקציה עם החתימה

int breakSentence(char * sentence, char * separators, char * file_name)

אשר מקבלת אותן מחרוזות sentence ו- separators שבסעיף א' וגם מחרוזת file_name המהווה שם של קובץ טקסט. על הפונקציה לשמור את המילים של המשפט בקובץ שנוצר (כל מילה בשורה נפרדת) ולהחזיר את כמות המילים במשפט.

לדוגמא:

עבור sentence = "But why?! That's why." ו- separators = "? !." הפונקציה תחזיר 4 ותוכן הקובץ יהיה

But
why
That's
why

יש להשתמש בפונקציה nextWord כפונקציית עזר. אפשר להשתמש במערך עזר. במקרה זה, ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

תשובה:

```
int breakSentence(char* sentence, char* separators, char* file_name)
{
    FILE* fptr = fopen(file_name, "w");
    char* word;
    int count = 0;
    if (!fptr)
        return -1;

    word = (char*)malloc(strlen(sentence) + 1);
    while ( (sentence = nextWord(sentence, separators, word)) != NULL )
    {
        fprintf(fptr, "%s\n", word);
        count++;
    }
    if (*word) // sentence isn't empty
    {
        fprintf(fptr, "%s\n", word); // last word
        count++;
    }
    free(word);

    fclose(fptr);
    return count;
}
```

סעיף ג (10 נק')

כתבו פונקציה עם החתימה

int countWordOccur(char * sentence, char * separators, char * query)

אשר מקבלת אותן מחרוזות sentence ו- separators שבסעיף א' וגם מחרוזת query המהווה מילה כלשהיא.
על הפונקציה להחזיר את כמות המופעים של המילה query ב- sentence.

לדוגמא, עבור

separators = "? !.", sentence = "But **why**?! That's **why**. And don't ask me **why**."
ו- query = "why" תחזיר 3.

יש להשתמש בפונקציה nextWord כפונקציית עזר.
אפשר להשתמש במערך עזר. במקרה זה, ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

תשובה:

```
int countWordOccur(char* sentence, char* separators, char* query)
{
    char* word;
    int count = 0;

    word = (char*)malloc(strlen(sentence) + 1);
    while ( (sentence = nextWord(sentence, separators, word)) != NULL )
        if (strcmp(word, query) == 0)
            count++;
    if (strcmp(word, query) == 0) // last word
        count++;
    free(word);

    return count;
}
```


חלק 3.

שאלה 4 (10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>

int x = 30, y = 0;

void f1(int x)
{
    printf("x=%d,y=%d\n", x, y);
    x = y = 10;
}

void f2(int y)
{
    printf("x=%d,y=%d\n", x, y);
    x = y = 15;
}

int main()
{
    y = 5;
    f1(20);
    f2(y);
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

יש לנמק את התשובה.

תשובה ונימוק:

$x=20, y=5$
 $x=30, y=10$
 $x=15, y=10$

אין משתנים x, y ב- `main`. לכן ב- `main` עובדים עם משתנים גלובליים x ו- y שמאותחל ב- 0 יקבל ערך חדש 5.
אחרי קריאה ל- $f1(20)$ פרמטר פורמלי x של $f1$ יקבל ערך 20. אין משתנה y ב- $f1$ ולכן הפונקציה עובדת עם y גלובלי אשר שווה ל- 5.
לכן `printf` ב- $f1$ מדפיסה $x=20, y=5$.
לאחר מכן x פורמלי y גלובלי משתנים ב- 10.
אחרי קריאה ל- $f2(y)$ פרמטר פורמלי y של $f2$ יקבל ערך 10 של y גלובלי. אין משתנה x ב- $f2$ ולכן הפונקציה עובדת עם x גלובלי אשר מאותחל ב- 30.
לכן `printf` ב- $f2$ מדפיסה $x=30, y=10$.
לאחר מכן x גלובלי y פורמלי משתנים ב- 15.
ואז לאחר חזרה ל- `main` x גלובלי שווה ל- 15 ו- y גלובלי שווה ל- 10.
לכן `printf` ב- `main` מדפיסה $x=15, y=10$.

שאלה 5 (10 נק')

בחרו את התשובה הנכונה ונמקו בכל אחד מהסעיפים הבאים:

א. מהי הטענה הנכונה?

- (1) הקריאות puts("abc") ו- fputs("abc",stdout) הן שקולות
- (2) הקריאות puts("abc") ו- fputs("abc\n",stdout) הן שקולות
- (3) הקריאות puts("abc\n") ו- fputs("abc",stdout) הן שקולות
- (4) הקריאות puts("abc") ו- fputs("\nabc",stdout) הן שקולות
- (5) אף טענה אינה נכונה

נימוק:

לעומת puts, fputs לא מוסיפה 'נ' בסוף המחרוזת.
לכן fputs("abc",stdout) מציגה במסך abc ללא הורדת שורה.
בזאת fputs("abc\n",stdout) מציגה במסך abc ומורידה סמן לשורה הבאה, בדיוק כמו puts("abc").

ב. איזה פעולה היא בלתי חוקית?

- (1) של מספר שלם למצביע
- (2) חיסור של מספר שלם ממצביע
- (3) חיבור של מצביע למצביע
- (4) חיסור של מצביע ממצביע
- (5) השוואת מצביעים
- (6) כל הפעולות לעיל הן חוקיות

נימוק:

בשפת C מוגדרות הפעולות הבאות על מצביעים:
חיבור (חיסור) מספר שלם למצביע (התוצאה היא כתובת),
חיסור מצביעים (התוצאה היא מספר שלם),
השוואת מצביעים.
פעולת חיבור מצביעים לא מוגדרת.

חלק 4.

שאלה 6 (בנוס – 10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>
#include <string.h>

void main()
{
    puts("puts"-strlen("strlen")+sizeof"sizeof");
}
```

יש לנמק את התשובה.

תשובה ונימוק:

uts

פונקציה strlen מחזירה מספר תווים במחרוזת עד לתו אפסי לא כולל תו אפסי. לכן strlen("strlen") מחזירה 6. אופרטור sizeof מחזיר את גודל האופרנד בבתים. מחרוזת "sizeof" מהווה 7 תווים כולל תו אפסי. לכן sizeof"sizeof" מחזיר 7. ואז המשפט שקול ל- puts("puts"-6+7) ז"א ל- puts("puts"+1). למחרוזת קבועה במשפט צריך להתייחס כמו לכתובת התו הראשון שלה. בזאת מקבלים למעשה puts("uts"). לכן הפלט יהיה uts.



מכון טכנולוגי חולון
Holon Institute of Technology

- דף טיוטה -

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences
Department of Computer Science



מכון טכנולוגי חולון
Holon Institute of Technology

- דף טיוטה -

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences
Department of Computer Science



מכון טכנולוגי חולון
Holon Institute of Technology

- דף טיוטה -

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences

Department of Computer Science



מכון טכנולוגי חולון
Holon Institute of Technology

- דף טיוטה -

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences

Department of Computer Science



מכון טכנולוגי חולון
Holon Institute of Technology

- דף טיוטה -

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences
Department of Computer Science



מכון טכנולוגי חולון
Holon Institute of Technology

- דף טיוטה -

גולומב 52, ת.ד. 305, חולון 5810201
טלפקס: 03-5026528

52 Golomb St., Holon 5810201 Israel

www.hit.ac.il Tel/Fax: 972-3-502-6528

הפקולטה למדעים
המחלקה למדעי המחשב

Faculty of Sciences
Department of Computer Science