



מכון טכנולוגי חולון
Holon Institute of Technology

מבחן סדנה מתקדמת בתכנות - 61108

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

סמסטר ב', תשפ"ה

מועד א', 11.06.2025

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.**
- 5 דפים אחרונים מיועדים לטייטה.
- בזמן הבחינה אסור להפריד או לתלוש את דפי הבחינה. בנוסף, בזמן הבחינה אסור לפרק ולשדך 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 – בONUS.

בהצלחה!

חלק 1.

שאלה 1 (40 נק')

נגדיר מבנה המייצג איבר ברשימה מקושרת:

```
typedef struct node {  
    char value;  
    struct node* next;  
} node;
```

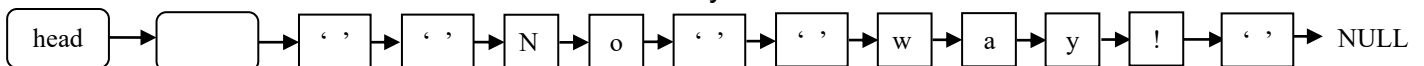
כתבו פונקציה **void removeExtraSpaces(node** head)** אשר מקבלת פוינטר **head** המצביע על ראש רשימה מקושרת חד כיוונית של איברים מסוג **node** המייצגת משפט כלשהו. על הפונקציה להסיר איברים מתוך הרשימה המהווים תווי רווח (' ') **מיותרים**. תו רווח נחשב **מיותר** אם הוא מופיע **מיד אחרי תו רווח אחר**, או נמצא **בתחילת הרשימה**, או נמצא **בסוף הרשימה**.

הערות:

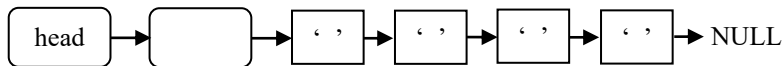
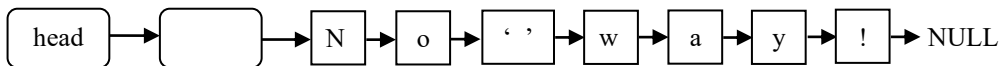
- הפונקציה לא אמורה לייצר איברים חדשים.
- אם הרשימה מכילה רק רווחים, הפונקציה תהפוך אותה לרשימה ריקה.
- במידה והרשימה ריקה, היא תישאר ריקה לאחר סיום הפונקציה.

דוגמאות:

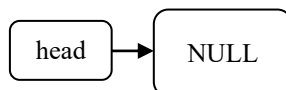
עבור הרשימה הבאה המייצגת את המשפט "No way!":



הפונקציה תעדכן את הרשימה בהתאם:



עבור הרשימה הבאה:



ישוחררו כל איברי הרשימה ויתקבל:

```
void deleteHead(node** head)
{
    node* temp = *head;
    *head = temp->next;
    free(temp);
}

void deleteNext(node* pos)
{
    node* temp = pos->next;
    pos->next = temp->next;
    free(temp);
}

void removeExtraSpaces(node** head)
{
    node* curr, *prev;
    while (*head && (*head)->value == ' ')
        deleteHead(head); //remove first spaces
    if (*head == NULL)
        return;
    prev = *head;
    curr = prev->next;
    while (curr)
    {
        if (curr->value == ' ' && (!curr->next || curr->next->value == ' '))
            deleteNext(prev); //remove space if no next or the next is space too
        else
            prev = curr;
        curr = prev->next;
    }
}
```



מכון טכנולוגי חולון
Holon Institute of Technology

תשובה (המשך):

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

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

שאלה 2 (40 נק')

האלכסון הראשי של מטריצה ריבועית הוא האלכסון שמתחיל מהפינה השמאלית העליונה של המטריצה עד לפינה הימנית התחתונה.
האלכסון המשני של מטריצה ריבועית הוא האלכסון שמתחיל מהפינה השמאלית התחתונה של המטריצה עד לפינה הימנית העליונה.
כל איברי המטריצה הריבועית הנמצאים באלכסון הראשי שלה ומתחתיו מרכיבים את **המשולש השמאלי התחתון** של המטריצה.
כל איברי המטריצה הריבועית הנמצאים באלכסון המשני שלה ומעליו מרכיבים את **המשולש השמאלי העליון** של המטריצה.

דוגמא:

המשולש השמאלי העליון	המשולש השמאלי התחתון	המטריצה
$\begin{bmatrix} 5 & 7 & -3 & 13 \\ 76 & 16 & 8 & \\ 98 & 4 & & \\ 11 & & & \end{bmatrix}$	$\begin{bmatrix} 5 & & & \\ 76 & 16 & & \\ 98 & 4 & 43 & \\ 11 & 19 & 21 & 2 \end{bmatrix}$	$\begin{bmatrix} 5 & 7 & -3 & 13 \\ 76 & 16 & 8 & 0 \\ 98 & 4 & 43 & 6 \\ 11 & 19 & 21 & 2 \end{bmatrix}$

כתבו פונקציה `void triangles(int A[][N], int n, int ***tr_low, int ***tr_up)` אשר מקבלת מטריצה ריבועית של מספרים שלמים המיושמת כ**מערך דו-ממדי סטטי** (לא דינאמי) **A** בגודל $n \times n$. המספר הפיסי של השורות (העמודות) יינתן ע"י קבוע בשם **N**. יש להגדיר אותו לפני הפונקציה.

על הפונקציה ליצור שני **מערכים דו-ממדיים דינאמיים** (בעלי שורות בגדלים שונים) המיישמים את המשולש השמאלי התחתון ואת המשולש השמאלי העליון של המטריצה. גדלי השורות במערכים יהיו שווים לכמויות האיברים בשורות המתאימות של המשולשים. יש להעביר את המערכים by reference באמצעות מצביעים **tr_low** ו- **tr_up** בהתאמה.

ניתן להניח שיש בזיכרון מספיק מקום להקצאה.

```
#define N 4

void triangles(int A[][N], int n, int*** tr_low, int*** tr_up)
{
    int i, j, ** low, ** up;

    low = (int**)calloc(n, sizeof(int*));
    for (i = 0; i < n; i++)
    {
        low[i] = (int*)calloc(i + 1, sizeof(int));
        for (j = 0; j <= i; j++)
            low[i][j] = A[i][j];
    }

    up = (int**)calloc(n, sizeof(int*));
    for (i = 0; i < n; i++)
    {
        up[i] = (int*)calloc(n - i, sizeof(int));
        for (j = 0; j < n - i; j++)
            up[i][j] = A[i][j];
    }

    *tr_low = low;
    *tr_up = up;
}
```




מכון טכנולוגי חולון
Holon Institute of Technology

תשובה (המשך):

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

שאלה 3 (40 נק')

כתבו פונקציה `int wordAppearanceNum(char *file_name, char *word)` אשר מקבלת מחרוזת `file_name` המהווה שם של קובץ טקסט ומחרוזת `word`. תוכן הקובץ מורכב משורות. ידוע שאורך השורה (כולל 'ח') לא עולה על 100. השורה האחרונה גם מסתיימת ב-'ח'.
על הפונקציה להחזיר את **כמות המופעים** של המילה `word` בקובץ (**מילה** היא רצף תווים שאינם רווח או 'ח') המופרדים ממילים אחרות ע"י רווחים או תווי 'ח'). אם הקובץ לא נפתח, הפונקציה תחזיר -1.

שימו לב:

יש להתייחס רק למילה שלמה. אין להתייחס לרצף שהוא תת-מילה במילה אחרת.

דוגמא:

עבור הקובץ הבא והמילה **live**

I believe that they **live** here and who does not **live** here
Mike lives in this house
And who does **live** here ? We **live**
10 alive creatures **live** in this lived place

הפונקציה תחזיר 5.

אפשר (אבל לא חובה) להשתמש במערך עזר של תווים בגודל שאינו עולה על 101.

רמז:

ניתן להשתמש בפונקציות סטנדרטיות.

גרסה 1:

```
int wordAppearanceNum(char* file_name, char* word)
{
    FILE* fptr = fopen(file_name, "r");
    char str[101], *sub_str, *p;
    int count = 0, len = strlen(word);
    if (!fptr)
        return -1;

    while (fgets(str, 101, fptr))
    {
        sub_str = str;
        while ((p = strstr(sub_str, word)) != NULL) //p - the first occurrence of word in
substring
        {
            //checking edges: is the found word complete or subword
            if ((p == str || *(p - 1) == ' ') && (*(p + len) == ' ' || *(p + len) == '\n'))
                count++;
            sub_str = p + len + 1; //the next substring
        }
    }
    fclose(fptr);
    return count;
}
```

גרסה 2:

```
int wordAppearanceNum(char* file_name, char* word)
{
    FILE* fptr = fopen(file_name, "r");
    char str[100], *separators = " \n", *token;
    int count = 0;
    if (!fptr)
        return -1;

    while (fgets(str, 100, fptr))
    {
        token = strtok(str, separators); //the first token
        while (token != NULL)
        {
            if (strcmp(token, word) == 0) //comparison of token with word
                count++;
            token = strtok(NULL, separators); //the next token
        }
    }
    fclose(fptr);
    return count;
}
```


חלק 3.

שאלה 4 (10 נק')

מהו הפלט של התוכנית הבאה?

```
#include <stdio.h>

typedef struct
{
    int x;
    int y;
} type1;

typedef union
{
    int x;
    int y;
} type2;

void fun(type2 b) {
    b.x++;
    printf("%d %d\n", b.x, b.y);
}

void main() {
    type1 a;
    type2 b;

    a.x = b.x = 3;
    a.y = b.y = 5;
    printf("%d %d\n", a.x, a.y);
    printf("%d %d\n", b.x, b.y);

    b.x = 8;
    b.y = 4;
    a.x = b.x;
    a.y = b.y;
    printf("%d %d\n", a.x, a.y);

    b.x = 15;
    b.y = 18;
    fun(b);
    printf("%d %d\n", b.x, b.y);
}
```

יש לנמק את התשובה.

תשובה:

3 5
 5 5
 4 4
 19 19
 18 18

נימוק:

שדות x, y של המבנה מקבלים ערכים 3,5 בהתאמה.
 שדות x, y של האיגוד מקבלים ערכים 3,5 בהתאמה. השדות תופסים אותו מקום בזיכרון ולכן שניהם שווים לערך 5 של העדכון האחרון.
 לכן שני הפלטים הראשונים:

3 5
 5 5

שדות x, y של האיגוד מקבלים ערכים 8,4 בהתאמה. השדות תופסים אותו מקום בזיכרון ולכן שניהם שווים לערך 4 של העדכון האחרון.
 שדות x, y של המבנה מקבלים ערכי שדות x, y של האיגוד בהתאמה. לכן שניהם גם שווים ל-4.
 לכן הפלט השלישי הוא

4 4

שדות x, y של האיגוד מקבלים ערכים 15,18 בהתאמה. השדות תופסים אותו מקום בזיכרון ולכן שניהם שווים לערך 18 של העדכון האחרון. האיגוד מועבר לפונקציה f והאיגוד מועתק לאיגוד שהוא פרמטר פורמאלי של f . שדה x שלו (ובהתאם שדה y) מתקדם ב-1 וכך שני השדות שלו יהיו שווים ל-19.
 לכן הפלט הרביעי הוא

19 19

אחרי יציאה מהפונקציה חוזרים לאיגוד המקורי בו שני השדות שווים ל-18.
 לכן הפלט החמישי הוא

18 18



מכון טכנולוגי חולון
Holon Institute of Technology

נימוק (המשך):

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

שאלה 5 (10 נק')

בחרו את התשובה הנכונה ונמקו בכל אחד מהסעיפים הבאים:

א. מהו פלט (משמאל לימין) של התוכנית הבאה?

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

```
void main()
```

```
{
```

```
    printf("%d%d%d", sizeof(char*), sizeof(double*), sizeof(void*));
```

```
}
```

184 (1)

444 (2)

180 (3)

440 (4)

לא יהיה פלט כי גודל של מצביע ל- void לא מוגדר (5)

אף תשובה אינה נכונה (6)

נימוק:

גדלי כל המצביעים ללא קשר לטיפוס (גם של מצביע כללי) הם 4 בתים.

ב. בחרו מהו ערך של הביטוי $1 < 24$

- | | |
|---------------------|-----|
| 240 | (1) |
| 2 | (2) |
| 2.4 | (3) |
| 48 | (4) |
| 12 | (5) |
| אף תשובה אינה נכונה | (6) |

נימוק

הזזת סיבית אחת שמאלה במספר בה הסיבית בשמאלית היא 0 שווה ערך להוספת סיבית 0 לימין, כלומר הכפלת המספר ב-2.
 $24 \times 2 = 48$

חלק 4.

שאלה 6 (בונוס – 10 נק')

בהינתן קובץ הטקסט file.dat (שנפתח בהצלחה) בעל התוכן

abc
↓
סוף הקובץ

מהו הפלט למסך של התוכנית הבאה?

```
#include <stdio.h>
void main()
{
    char c;
    FILE* fptr = fopen("file.dat", "r");
    if (!fptr) return;
    while (c = fgetc(fptr) != EOF)
        printf("%d", c);
    fclose(fptr);
}
```

יש לנמק את התשובה.

תשובה:

111

נימוק

בהיעדר סוגריים, לפי סדר הקדימות, פעולת השוואה מבוצעת לפני פעולת השמה. עבור תווי 'c', 'b', 'a' התנאי `fgetc(fptr) != EOF` מתקיים ולכן ערך הביטוי יהיה 1. כלומר c בשלוש האיטרציות יהיה 1, תנאי המשך הלולאה יתקיים ובהתאם, 1 יודפס שלוש פעמים. עבור EOF התנאי `fgetc(fptr) != EOF` לא מתקיים ולכן ערך הביטוי יהיה 0. כלומר c יהיה 0 ותנאי המשך הלולאה לא יתקיים.



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