



מבוא למדעי המחשב מ' (02340114)

מבחן פטור

מבחן לדוגמה

--	--	--	--	--	--	--	--	--	--

מספר סטודנט:

משך המבחן: שלוש שעות.

הנחיות כלליות:

- השימוש בכל חומר עזר אסור.
- בדקו שיש 10 עמודים (4 שאלות) במבחן, כולל עמוד זה.
- אלא אם כן נאמר אחרת בשאלות, אין להשתמש בפונקציות ספרייה למעט הבאים:
 - פונקציות קלט/פלט (printf, scanf) מהספרייה stdio.h
 - פונקציות הקצאת זיכרון (malloc, free) מהספרייה stdlib.h
 - הטיפוס bool המוגדר ב-stdbool.h
- אין להשתמש במשתנים סטטיים וגלובליים אלא אם נדרשתם לכך מפורשות.
- הקפידו על סגנון כתיבה כזה שקורא התוכנית ידע לפענח את רעיונותיכם:
 - התוכנית יכולה להכיל תיעוד קל להבנה.
 - על התוכנית להיות כתובה באופן מסודר ומדורג.
 - יש לתת שמות פונקציות ושמות משתנים משמעותיים.
 - ערכים קבועים יש להגדיר בעזרת define.
 - ניתן להוסיף פונקציות עזר כרצונכם.
- נוהל "לא יודע": אם תורידו את ההערה מהקריאה לפונקציה printIDontKnow() על שאלה שבה אתם נדרשים לקודד, תקבלו 20% מהניקוד. דבר זה מומלץ אם אתם יודעים שאתם לא יודעים את התשובה.
- שימו לב שקיבלתם את קבצי ה-C ובהם שלד של התוכנית. עליכם רק להשלים את המימוש של הפונקציה הנדרשת. אין לשנות את ה-main.
- לכל שאלה קיבלתם מספר טסטים. מומלץ להוסיף טסטים שלכם.

המשך ההנחיות בעמוד הבא.



הנחיות הגשה: חשוב לקרוא בעיון!

- (1) בחצי שעה הראשונה של המבחן, יש למלא במודל הצהרה על טוהר הבחינות. סיסמה להצהרה במודל נמצאת בעמוד 3. ניתן למלא את ההצהרה רק בחצי השעה הראשונה.
- (2) הגשת כל שאלה מתבצעת במטלה נפרדת במערכת Gradescope רק לאחר ההצהרה על טוהר הבחינות, ולא בחצי השעה הראשונה.
- (3) יש למלא הצהרה על סיום הבחינה במודל לאחר הגשת המטלה האחרונה שפתרתם. מילוי הצהרה זו יפתח לכם גישה לתמונה במודל. יש להראות את התמונה למשגיחים לפני היציאה מהכיתה.
- (4) במהלך המבחן מותר לעבוד אך ורק בסביבת CLion.
- (5) אין להשתמש בשום כלי בינה מלאכותית יוצרת כגון: CoPilot, ChatGPT וכדומה. שימוש בכלים אלו יגרור הליכים משמעתיים.
- (6) יש להתחבר מהמחשב הנייד **אך ורק לרשת ה-wifi הטכניונית**, ובפרט חל איסור חמור להתחבר מ hotspot שמופעל על ידי מכשיר אישי (טלפון נייד, טבלט, לפטופ וכו'). שימוש ברשת שאינה רשת טכניונית יגרור הליכים משמעתיים.
- (7) הגשה לא לפי השלבים שפורטו לעיל, ובפרט: אי הצהרה על טוהר הבחינות/ אי הצהרה על סיום הבחינה, הגשת מטלה לאחר הצהרה על סיום הבחינה, שימוש בכלי בינה מלאכותית, התחברות לרשת חיצונית, **תוביל לציון 0** והליכים משמעתיים.

נוסחאות שימושיות:

$$1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} = \Theta(\log n) \quad \log 1 + \log 2 + \dots + \log n = \Theta(n \log n)$$

$$1^k + 2^k + 3^k + \dots + n^k = \Theta(n^{k+1}) \quad 1 + k + k^2 + k^3 + \dots + k^n = \Theta(k^n), \quad k > 1$$

$$S_n = \frac{a_1}{1-q} \text{ אינסופית: } S_n = a_1 * \frac{q^n - 1}{q - 1}$$

$$S_n = n * \frac{a_1 + a_n}{2} \text{ סכום סדרה חשבונית:}$$

$$\log_a(m \cdot n) = \log_a(m) + \log_a(n)$$

$$\log_a\left(\frac{m}{n}\right) = \log_a(m) - \log_a(n)$$

$$\log_a(m^n) = n \log_a(m)$$

$$n^{\log_n m} = m$$



סיסמה להצהרה על טוהר הבחינות:

BluePanther



שאלה 1: [25 נקודות]

ענו על השאלה במטלה המתאימה ב-Gradescope. לכל סעיף עם קטע קוד יופיעו שני תתי-סעיפים, אחד לסיבוכיות זמן ואחד לסיבוכיות מקום. בכל תת-סעיף עליכם לבחור אפשרות אחת ולסמן אותה. בכל הסעיפים יש להניח כי סיבוכיות הזמן והמקום של פונקציות הספרייה המוכרות *printf*, *free* היא $\Theta(1)$. בנוסף יש להניח כי סיבוכיות הזמן של *malloc(n)* היא $\Theta(1)$ וסיבוכיות מקום נוסף היא $\Theta(n)$.

א. חשבו את סיבוכיות הזמן והמקום של הפונקציה *f1* המוגדרת בקטע הקוד הבא, כפונקציה של *n*. אין צורך לפרט את שיקוליכם. חובה לפשט את הביטוי ככל שניתן.

```
int aux1(int a, int b) {
    return (5 * a + b % 8) / (a + b);
}

void f1(int n) {
    if (n <= 0) return;

    int *p = (int*) malloc(n * sizeof(int));
    for (int i = n; i > 0; i /= 2) {
        for (int j = 1; j <= i; j++) {
            int r = aux1(i, j);
            if (r % 2 != 0) {
                void *q = malloc(sizeof(char));
                if (q) free(q);
            }
        }
    }
    if (p) free(p);
}
```

ב. חשבו את סיבוכיות הזמן והמקום של הפונקציה *f2* המוגדרת בקטע הקוד הבא, כפונקציה של *n*. אין צורך לפרט את שיקוליכם. חובה לפשט את הביטוי ככל שניתן.

```
int f2(int n) {
    if (n < 3) return 1;
    int* arr = (int*) malloc(sizeof(int) * n);
    f2(f2(n - 2));
    if (arr) free(arr);
    return n;
}
```



ג. בקובץ exam_q1c.c קיימת פונקציה find_in_dict בשפת סי שמבצעת חיפוש מילה במערך מחרוזות. בקובץ יש גם לרשותכם תוכנית אשר קולטת את מערך המחרוזות מהמשתמש (אורכו ולאחר מכן לכל מחרוזת אורך ותוכן), ואת המילה לחיפוש (אורך ותוכן), קוראת לפונקציה ומדפיסה את הערך שהוחזר. ענו על 3 השאלות בגריידיסקופ לגבי קוד זה. שימו לב:

- אתם יכולים להיעזר ב sandbox כדי להריץ את הקוד עם קובץ טסטים שלכם.
- הפונקציה strcmp בספריה string.h מחזירה 0 אם המחרוזות זהות, מספר חיובי אם s1 תופיע בסדר מילוני לאחר s2, מספר שלילי אם s1 תופיע בסדר מילוני לפני s2.

הפונקציה לבדיקה:

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

int find_in_dict(char* dict[], int words, char* target) {
    int left = 0, right = words - 1;
    while (left <= right) {
        int mid = left + (right - left) / 2;
        int temp = strcmp(dict[mid], target);
        if (temp == 0) {
            return mid;
        }
        if (temp < 0) {
            left = mid + 1;
        }
        else {
            right = mid - 1;
        }
    }
    return -1;
}
```



שאלה 2: [25 נקודות]

נגדיר משקל של תו כמרחקו מהתו 'a'. למשל המשקל של התו 'b' הוא 1 והמשקל של התו 'z' הוא 25.

נגדיר משקל מחרוזת כסכום משקלות התווים בה. למשל המשקל של המחרוזת "ab" הוא 1, המשקל של המחרוזת "cd" הוא 5 והמשקל של המחרוזת "zzz" הוא 75.

ממשו פונקציה שחתימתה:

```
int exam_q2(char * arr[], int n, int x);
```

הפונקציה מקבלת מערך של n מחרוזות, ממוינות לפי משקלן, מהכי קלה להכי כבדה, וערך שלם x . על הפונקציה להחזיר 1 אם ניתן למצוא 3 מחרוזות שונות במערך שסכום משקלן הוא x , ו-0 אחרת.

אורך המחרוזת המקסימלית אינו חסום.

דוגמאות:

קלט הפונקציה	פלט	הסבר
$arr = \{"ab", "cd", "zzz"\}$, $n = 3, x = 81$	1	סכום המשקלות של 3 המחרוזות הוא $81 = 1 + 5 + 75$.
$arr = \{"c", "cb", "cia", "ciao"\}$, $n = 4, x = 2$	0	המחרוזת הראשונה במשקל 2, וכל האחרות בעלות משקל גבוה יותר, לכן לא ניתן להגיע לסכום משקלות שהוא 2.

דרישות סיבוכיות:

- סיבוכיות זמן: $O(m \cdot n + n^2)$, כאשר n אורך המערך, ו- m אורך המחרוזת המקסימלית. פתרון בסיבוכיות $O(m \cdot n + n^2 \log n)$ יקבל ניקוד חלקי.
- סיבוכיות מקום: $O(n)$

רמז לעמידה בסיבוכיות:

חשבו איך הייתם פותרים את השאלה בסיבוכיות $O(n)$ אם הייתה נתונה לכם המחרוזת האמצעית מבין השלוש.

המשך השאלה בעמוד הבא



הערות:

- לרשותכם תוכנית אשר קולטת את מערך המחרוזות מהמשתמש (אורכו ולאחר מכן לכל מחרוזת אורך ותוכן), ואת הערך x , קוראת לפונקציה ומדפיסה את הערך שהוחזר.
- מובטח כי המערך אכן ממוין על פי המשקל וכל המחרוזות חוקיות, מכילות אותיות אנגליות קטנות בלבד ואינן ריקות.
- מובטח כי לכל המחרוזות משקל שונה.
- מובטח כי במערך יש לפחות 3 מחרוזות.
- בשאלה זו בלבד ניתן להיעזר בפונקציות $strlen$, $strcmp$ מהספרייה $string.h$.

שם הפונקציה	קלט הפונקציה	פלט הפונקציה	סיבוכיות זמן	סיבוכיות מקום
$strlen$	s מחרוזת	אורך המחרוזת (ללא התו $'\backslash 0'$)	$\theta(m)$, כאשר m אורך המחרוזת	$\theta(1)$
$strcmp$	$s1, s2$ שתי מחרוזות להשוואה	0 אם המחרוזות זהות מספר חיובי אם $s1$ תופיע במילון לאחר $s2$ מספר שלילי אם $s1$ תופיע במילון לפני $s2$	$\theta(m)$, כאשר m אורך המחרוזת הקצרה	$\theta(1)$



שאלה 3 [25 נקודות]

עליכם לממש את הפונקציה הבאה:

```
void exam_q3(int arr[ ], int n, int res[ ]);
```

הפונקציה מקבלת מערך המתאר ציוני n סטודנטים, ומערך res אותו עליה למלא. הפונקציה תרשום ב $res[i]$ את הדירוג של הסטודנט ה i על פי הציון שלו. כלומר הסטודנט עם הציון הנמוך ביותר ידורג 1, והסטודנט עם הציון הגבוה ביותר ידורג במקום הגבוה ביותר.

שימו לב:

- אם לשני סטודנטים אותו ציון, הם ידורגו באותו מקום.
- מובטח כי כל הציונים בין 0 לבין 100.

להלן מספר דוגמאות:

קלט הפונקציה	מערך הדירוגים
$arr = \{100, 40\}$ $n = 2$	$res = \{2, 1\}$ כיוון שלסטודנט הראשון ציון גבוה יותר.
$arr = \{0, 100, 0\}$ $n = 3$	$res = \{1, 2, 1\}$ כיוון שלסטודנט הראשון והשלישי אותו ציון והוא הנמוך ביותר שניהם ידורגו 1, והסטודנט השני שלו הציון הגבוה ביותר ידורג 2.

הערות:

- נתונה לכם תוכנית אשר קוראת מהמשתמש את מספר הסטודנטים ואת מערך הציונים, קוראת לפונקציה ומדפיסה את המערך של הדירוגים.

דרישות סיבוכיות:

- סיבוכיות זמן: $O(n)$
- סיבוכיות מקום נוסף: $O(1)$.



שאלה 4: [25 נקודות]

בגן החיות יש N חיות, נמספר אותם מ 0 עד $N-1$.
לאחר שגן החיות נסגר, השומר בגן מוציא לטיול את כל החיות בגן.

ישנן חיות שלא יכולות לצאת יחד לטיול, מחשש שיטרפו זו את זו. יחס זה מתואר במטריצת אילוצים בגודל $N \times N$. תא בשורה i עמודה j במטריצה זו יהיה 1 אם חיות i ו j יכולות לטייל יחד, ו 0 אחרת. ניתן להניח שהמטריצה סימטרית וכי באיברי האלכסון רשום 1 .

כיוון שגן החיות משלם לשומר על כל טיול, ההנהלה מעוניינת במספר הטיולים המינימלי הנדרש.

עליכם לממש את הפונקציה הבאה: `int exam_q4(int mat[N][N]);`

הפונקציה מקבלת את מטריצת האילוצים (`mat`), ומחזירה את מספר הטיולים המינימלי שיש לעשות.

שימו לב שמספר החיות N מוגדר ב-`#define`, והפונקציה צריכה לעבוד גם אם נשנה אותו.

לדוגמה עבור $N = 3$, ועבור מטריצת האילוצים הבאה:

1	0	0
0	1	1
0	1	1

השומר יוציא את חיה 0 לבד, ויעשה טיול נוסף עם חיות 1 ו 2 . סה"כ השומר יעשה 2 טיולים. הפונקציה תחזיר 2 .

עבור $N = 4$, ועבור מטריצת האילוצים הבאה:

1	0	0	0
0	1	0	1
0	0	1	0
0	1	0	1

השומר יוציא את חיה 0 לבד, ויעשה טיול נוסף עם חיות 1 ו 3 , וטיול נוסף עם חיה 2 לבד. סה"כ השומר יעשה 3 טיולים. הפונקציה תחזיר 3 .



הערות:

- נתונה לכם תוכנית אשר מקבלת מהמשתמש את מטריצת האילוצים, קוראת לפונקציה, ומדפיסה למסך את הערך החוזר מהפונקציה.
- יש להשתמש בשיטת `backtracking`.
- בשאלה זו אין דרישות סיבוכיות, אולם כמקובל ב-`backtracking` יש לוודא שלא מתבצעות קריאות רקורסיביות מיותרות עם פתרונות שאינם חוקיים.
- מותר להשתמש בפונקציית מעטפת, אבל אין לשנות את פונקציית `main` הנתונה.
- ניתן ומומלץ להשתמש בפונקציות עזר (ויש לממש את כולן).

בהצלחה!