



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:
 - a. פונקציות קלט/פלט (printf, scanf) מהספרייה stdio.h
 - b. פונקציות הקצאת זיכרון (malloc, free) מהספרייה stdlib.h
 - c. 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:
 - a. התוכנית יכולה להכיל תיעוד קל להבנה.
 - b. על התוכנית להיות כתובה באופן מסודר ומדורג.
 - c. יש לתת שמות פונקציות ושמות משתנים משמעותיים.
 - d. ערכים קבועים יש להגדיר בעזרת define.
 - e. 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:

RubberDuck



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.

```
void f1(int n) {
    int m = n / 2;
    int total_size = 0;
    for (int i = 1; i * i < m/4; i++) {
        for (int j = i; j < m * 2; j*=2) {
            total_size += i;
        }
    }
    int* a = malloc(total_size * sizeof(int));
    free(a);
}
```

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.

```
void aux2(int m) {
    if (m < 2) return;
    for (int i = 0; i < m; i++)
        printf("Good luck!");
    aux2(m/2);
    aux2(m/2);
}

void f2(int n) {
    aux2(n*n*n);
}
```



ג. בקובץ exam_q1c.c קיים קוד בשפת סי שמבצע חיפוש במערך.
ענו על 3 השאלות בגריידסקופ לגבי קוד זה.
שימו לב: אתם יכולים להיעזר ב sandbox כדי להריץ את הקוד עם קובץ טסטים שלכם.
הקוד לבדיקה:

```
#include <stdio.h>
#include <stdbool.h>
bool is_equal(char* a, char* b, int size);
int mystery_search(char* arr, int num_elements, char* x, int element_size, int idx);

int main()
{
    // example 1
    int arr_int[] = {-30, -16, -9, 12, 18, 21};
    int x_int = 12;
    int res1 = mystery_search((char*)arr_int, 6, (char*)&x_int, sizeof(int), 0);
    printf("index in int array: %d\n", res1);

    // example 2
    double arr_double[] = {-11.5, -8.0, -3.7, -.52, 5.2};
    double x_double = 5.2;
    int res2 = mystery_search((char*)arr_double, 5, (char*)&x_double, sizeof(double), 0);
    printf("index in double array: %d\n", res2);
}

bool is_equal(char* a, char* b, int size) {
    for (int i = 0; i < size; i++) {
        if (a[i] != b[i]) return false;
    }
    return true;
}

int mystery_search(char* arr, int num_elements, char* x, int element_size, int idx) {
    if (idx == num_elements) {
        return -1;
    }
    char* p = arr + idx * element_size;
    if (is_equal(p, x, element_size)) {
        return idx;
    }
    return mystery_search(arr, num_elements, x, element_size, idx + 1);
}
```



שאלה 2: [25 נקודות]

ממשו פונקציה שחתימתה:

```
int exam_q2(char ** arr, int n, int m);
```

הפונקציה מקבלת מערך של מחרוזות arr , את אורכו n , ופרמטר m . כל מחרוזת במערך מכילה לכל היותר 10 תווים.

רוצים ליצור מהמערך שיר קצר, על-ידי שרשור מחרוזות עוקבות במערך לשורות, כך שכל מחרוזת מופיעה בדיוק פעם אחת, ובסדר המקורי.

למשל עבור המערך $arr = ["I", "love", "computer", "science"]$ ניתן ליצור שיר עם 1,2,3 או 4 שורות. דוגמה לשיר עם 3 שורות:

```
I love
computer
science
```

דוגמה לשיר עם שורה אחת:

```
I love computer science
```

בכל מקרה, אין לפצל מחרוזת בין שורות, ואין לשנות את סדר המחרוזות.

הפרמטר m מייצג את מספר השורות המקסימלי המותר בשיר.

נגדיר אורך שורה להיות מספר התווים הכולל במחרוזות המרכיבות אותה (ללא הרווחים בין המחרוזות).

על הפונקציה להחזיר את אורך השורה המקסימלי הקטן ביותר האפשרי, כך שניתן יהיה לבנות את השיר באמצעות לכל היותר m שורות.

המשך השאלה בעמוד הבא

**דוגמאות:**

קלט הפונקציה	ערך מוחזר	הסבר
<pre>arr = ["I", "love", "computer", "science"] n = 4 m = 2</pre>	13	אפשר ליצור שיר עם שורה אחת באורך 20: I love computer science או שיר עם 2 שורות שהארוכה ביניהן באורך 19: I love computer science או שיר עם שתי שורות שהארוכה ביניהן באורך 15: I love computer science או שיר עם 2 שורות שהארוכה ביניהן באורך 13: I love computer science הערך הקטן ביותר הוא 13.

הערות:

- נתונה לכם תוכנית אשר קוראת מהמשתמש את המערך (אורך המערך ולאחר מכן לכל מחרוזת אורך ותוכן) ואת m .
- ניתן להניח שהקלט תקין, קיימת לפחות מחרוזת אחת, המחרוזות מכילות רק אותיות אנגליות ואורכן לכל היותר 10.
- מותר שלשורות שונות יהיו אורכים שונים.
- כל שורה חייבת להכיל לפחות מחרוזת אחת.
- מטרתכם היא למזער את אורך השורה הארוכה ביותר.
- מובטח כי ניתן ליצור את השיר על פי הכללים שהוגדרו.
- בשאלה זו בלבד מותר להשתמש בפונקציה `strlen` מהספרייה `string.h`. אסור להשתמש בפונקציות אחרות מספרייה זו.

דרישות סיבוכיות:

- סיבוכיות זמן: $O(n \log(n))$. פתרון בסיבוכיות $O(n^2)$ יקבל ניקוד חלקי.
- סיבוכיות מקום נוסף: $O(1)$.



שאלה 3 [25 נקודות]

עליכם לממש את הפונקציה הבאה:

```
int exam_q3(int a[], int n, int m, int b[]);
```

הפונקציה מקבלת מערך a בגודל n שכל האיברים בו הם מספרים שלמים בין 1 ל- m (כולל). על הפונקציה לכתוב למערך b את איברי a ללא חזרות, כך ש- b יהיה ממוין לפי מספר ההופעות של כל מספר במערך המקורי, מהכמות הקטנה ביותר לגדולה ביותר. יש להחזיר את מספר האיברים שנכתבו למערך b .

ניתן להניח כי במערך a לא קיימים שני מספרים שונים בעלי אותו מספר הופעות (סדר המיון מוגדר היטב).

דוגמאות:

קלט הפונקציה	המערך b לאחר הקריאה לפונקציה	ערך החזרה	הסבר
$n = 6$ $a = \{5, 2, 5, 5, 2, 8\}$ $m = 8$	8, 2, 5	3	המספר 8 מופיע פעם אחת, המספר 2 מופיע פעמיים, והמספר 5 מופיע 3 פעמים.
$n = 8$ $a = \{1, 3, 3, 3, 4, 4, 4, 4\}$ $m = 4$	1, 3, 4	3	המספר 1 מופיע פעם אחת, המספר 3 מופיע 3 פעמים, והמספר 4 מופיע 4 פעמים.
$n = 3$ $a = \{6, 6, 6\}$ $m = 7$	6	1	6 הוא המספר היחיד במערך.

הערות:

- נתונה לכם תוכנית אשר קוראת מהמשתמש את המערך a (אורך ואז תוכן) ואת המספר m .
- ניתן להניח שהמערך a חוקי, ומכיל רק מספרים שלמים בין 1 ל- m (כולל).
- ניתן להניח שבמערך b שמועבר לפונקציה יש מספיק מקום.

דרישות סיבוכיות:

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



שאלה 4: [25 נקודות]

עליכם לממש את הפונקציה הבאה:

```
int exam_q4(int mat[N][N]);
```

המקבלת מפת מבוך בגודל $N \times N$ מוגדר ב-`defined`, והפונקציה צריכה לעבוד גם אם נשנה אותו, ומחזירה את הזמן המינימלי הנדרש להגיע ממשבצת `mat[0][0]` למשבצת `mat[N-1][N-1]`, על פי הכללים הבאים:

- כל משבצת ב-`mat` היא אחד מהערכים הבאים:
 - 0 – קיר, לא ניתן להתקדם למשבצת זו.
 - 1 – חדר ריק, לוקח דקה לעבור אותו.
 - 2 – חדר המכיל אוצר, לוקח 2 דקות לאסוף את האוצר ולהמשיך.
 - 3 – חדר המכיל מפלצת איומה, לוקח 3 דקות להביס את המפלצת ולהמשיך.
- ניתן להתקדם במבוך רק דרך חדרים, ומכל חדר ניתן לעבור לחדר סמוך. לא ניתן לעבור לחדר באלכסון, אלא רק למעלה למטה ימינה או שמאלה.
- ידוע כי מתחילים את המסע בחדר `mat[0][0]`, ומסיימים אותו בחדר `mat[N-1][N-1]` (ניתן להניח שאלו חדרים ולא קיר), והזמן הנדרש עבור חדרי ההתחלה והסיום נכלל בזמן של המסלול.
- כיוון שהתחלנו את המסלול רעבים, והבטן מקרקרת, הזמן הוא המכריע העיקרי – בין שני מסלולים הדורשים זמנים שונים נעדיף את זה שלוקח הכי מעט זמן.
- אם יש שני מסלולים הלוקחים אותו זמן, אין חשיבות איזה נבחר.
- אם לא קיים מסלול חוקי יש להחזיר -1.

על הפונקציה להחזיר את הזמן שלוקח לעבור במסלול בעל הזמן המינימלי.

דוגמאות:

אוצר	חדר ריק	חדר ריק	חדר ריק	חדר ריק
חדר ריק	קיר	קיר	קיר	חדר ריק
חדר ריק	קיר	קיר	קיר	חדר ריק
חדר ריק	קיר	קיר	קיר	חדר ריק
חדר ריק	חדר ריק	אוצר	חדר ריק	מפלצת

קיימים שני מסלולים, הראשון עובר דרך 8 חדרים ריקים ואוצר אחד, לכן יקח 10 דקות, והשני עובר דרך 7 חדרים ריקים, אוצר אחד ומפלצת אחת, לכן יקח 12 דקות. הפונקציה תחזיר 10.

המשך השאלה בעמוד הבא



עבור המפה הבאה ($N=5$), המסלול המסומן הוא אחת האפשרויות למסלול בעל הזמן המינימלי.

עוברים דרך 11 חדרים ריקים, 3 אוצרות ומפלצת אחת, לכן הפונקציה תחזיר $11*1+3*2+1*3=20$.

חדר ריק	אוצר	חדר ריק	חדר ריק	חדר ריק
חדר ריק	קיר	קיר	מפלצת	חדר ריק
חדר ריק	חדר ריק	מפלצת	אוצר	חדר ריק
חדר ריק	חדר ריק	קיר	קיר	קיר
חדר ריק	חדר ריק	חדר ריק	חדר ריק	אוצר

ועבור המפה הבאה ($N=5$), לא קיים מסלול חוקי, לכן הפונקציה תחזיר 1-.

חדר ריק	אוצר	חדר ריק	חדר ריק	חדר ריק
חדר ריק	קיר	קיר	קיר	חדר ריק
חדר ריק	חדר ריק	מפלצת	קיר	חדר ריק
חדר ריק	חדר ריק	קיר	קיר	קיר
חדר ריק	חדר ריק	חדר ריק	חדר ריק	אוצר

הערות:

- נתונה לכם תוכנית אשר קוראת מהמשתמש את תוכן המבוך ומדפיסה למסך את הזמן של המסלול האופטימלי.
- יש להשתמש בשיטת `backtracking`.
- בשאלה זו אין דרישות סיבוכיות, אולם כמקובל ב-`backtracking` יש לוודא שלא מתבצעות קריאות רקורסיביות מיותרות עם פתרונות שאינם חוקיים.
- שמותר להשתמש בפונקציית מעטפת, אבל אין לשנות את פונקציית `main` הנתונה.
- ניתן ומומלץ להשתמש בפונקציות עזר (ויש לממש את כולן).
- ניתן להניח כי המטריצה מכילה את המספרים 0-3 בלבד, ושתאים `mat[0][0]` ו-`mat[N-1][N-1]` לא מכילים 0.
- מותר לשנות את ערכי המטריצה `mat` כרצונכם.

בהצלחה!