



מבוא למדעי מחשב מ' / ח' (234117 / 234114)

סמסטר אביב תשע"ג

מבחן מסכם מועד א', 11 יולי 2013

| | | | | | |
|---|---|---|---|---|--|
| 2 | 3 | 4 | 1 | 1 | |
|---|---|---|---|---|--|

רשום/ה לקורס:

| | | | | | | | | | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|

מספר סטודנט:

משך המבחן: 3 שעות.

חומר עזר: אין להשתמש בכל חומר עזר.

הנחיות כלליות:

- מלאו את הפרטים בראש דף זה ובדף השער המצורף, בעט בלבד.
- בדקו שיש 14 עמודים (4 שאלות) במבחן, כולל עמוד זה.
- כתבו את התשובות על טופס המבחן בלבד, במקומות המיועדים לכך. שימו לב שהמקום המיועד לתשובה אינו מעיד בהכרח על אורך התשובה הנכונה.
- העמודים הזוגיים בבחינה ריקים. ניתן להשתמש בהם כדפי טיוטה וכן לכתיבת תשובותיכם. סמנו טיוטות באופן ברור על מנת שהן לא תבדקנה.
- יש לכתוב באופן ברור, נקי ומסודר. ניתן בהחלט להשתמש בעיפרון ומחק, פרט לדף השער אותו יש למלא בעט.
- בכל השאלות, הינכם רשאים להגדיר ולממש פונקציות עזר כרצונכם. לנוחיותכם, אין חשיבות לסדר מימוש הפונקציות בשאלה, ובפרט ניתן לממש פונקציה לאחר השימוש בה.
- אלא אם כן נאמר אחרת בשאלות, אין להשתמש בפונקציות ספריה או בפונקציות שמומשו בכיתה, למעט פונקציות קלט/פלט והקצאת זיכרון (`malloc`, `free`). ניתן להשתמש בטיפוס `bool` המוגדר ב-`stdbool.h`.
- אין להשתמש במשתנים סטטיים וגלובאליים אלא אם נדרשתם לכך מפורשות.
- ניתן להשתמש בהקצאות זכרון בסגנון C99 (מערכים בגודל משתנה), בכפוף לדרישות סיבוכיות זכרון.
- כשאתם נדרשים לכתוב קוד באילוצי סיבוכיות זמן/מקום נתונים, אם לא תעמדו באילוצים אלה תוכלו לקבל בחזרה מקצת הנקודות אם תחשבו נכון ותציינו את הסיבוכיות שהצלחתם להשיג.
- נוסחאות שימושיות:

$$1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} = \Theta(\log n) \quad 1 + \frac{1}{4} + \frac{1}{9} + \frac{1}{16} + \frac{1}{25} + \dots = \Theta(1)$$

$$1 + 2 + \dots + n = \Theta(n^2) \quad 1 + 4 + 9 + \dots + n^2 = \Theta(n^3) \quad 1 + 8 + 27 + \dots + n^3 = \Theta(n^4)$$

צוות הקורס 234114/7

מרצים: פרופ' ניר אילון (מרצה אחראי), ד"ר תמיר לוי, חביאר טורק.

מתרגלים: נחשון כהן, נדיה לבאי, עלי חמוד, אביב סגל, ירון קסנר, ורד כהן, אנסטסיה דוברובינה (מתרגלת אחראית).

בהצלחה!

[illegible]



שאלה 1 (25 נקודות):

א. (12 נקודות) חשבו את סיבוכיות הזמן והמקום של הפונקציה f המוגדרת בקטע הקוד הבא, כפונקציה של n . אין צורך לפרט שיקולים.

```
int f(int n)
{
    if (n == 1)
        return 1;

    return f(f(n-1));
}
```

סיבוכיות זמן: $\Theta(\quad)$ סיבוכיות מקום: $\Theta(\quad)$

ב. (13 נקודות) נשנה עתה את הגדרת הפונקציה מהסעיף הקודם:

```
int f(int n)
{
    if (n == 1)
        return 1;

    return 1 + f(f(n-1));
}
```

מה ערך החזרה של הפונקציה f עבור הקלט n ? _____

סיבוכיות זמן: $\Theta(\quad)$ סיבוכיות מקום: $\Theta(\quad)$

[illegible]



5

[illegible]



סעיף ב (13 נקודות)

בהינתן מערך a של מספרים חיוביים, נאמר שמספר z הוא **מיוחד ביחס ל- a** אם קיימים שני מספרים $x < y$ שנמצאים שניהם ב- a כך ש z מתחלק ב- $x \cdot y$. לדוגמא אם המערך a הוא

```
int a[7] = {4, 7, 5, 4, 3, 30, 201}
```

אז המספר 24 הוא מיוחד ביחס ל- a כי הוא מתחלק ב- $3 \cdot 4 = 12$ והמספרים 3,4 במערך. לעומת זאת, 25 אינו מיוחד ביחס ל- a כי הוא אינו מתחלק במכפלה של שני מספרים שונים ב- a .

ממשו פונקציה המקבלת מערך a , את גודלו n , ומספר נוסף $m > 1$. על הפונקציה להדפיס את **כל המספרים** בתחום $1, 2, \dots, m$, המיוחדים ביחס ל- a . יש להדפיס את המספרים בסדר עולה ואין להדפיס אף מספר פעמיים.

דרישות: סיבוכיות זמן $O(n+m \cdot \log^2 m)$ וסיבוכיות מקום נוסף $O(m)$

ניתן ומומלץ להשתמש בפונקציה מהסעיף הקודם, גם אם לא מימשתם אותה.

אם לפי חישוביכם לא עמדתם בדרישות הסיבוכיות, אנא ציינו כאן את הסיבוכיות שהגעתם אליה: זמן _____ מקום נוסף _____

```
void print_specials(int a[], int n, int m)
```

```
{
```

[illegible]



עליכם לממש את הפונקציה:

[illegible]



שאלה 4 (25 נקודות) :

קליקה היא קבוצה של אנשים שבה כל זוג אנשים הם חברים. נתון מערך דו מימדי של חברויות `int friends[N][N]`: עבור זוג אנשים `i` ו-`j`, הערך `friends[i][j]` הוא 1 אם `i` ו-`j` חברים, ו-0 אחרת. המערך סימטרי, כלומר מתקיים `friends[i][j]==friends[j][i]` לכל `i` ו-`j`.

עליכם לכתוב פונקציה

```
int hasClique(int friends[][N], int k)
```

המקבלת מערך חברויות `friends`, ומספר `k`. הפונקציה תחזיר 1 אם קיימת קליקה המכילה לפחות `k` אנשים, ו-0 אם לא קיימת קליקה כזו.

לדוגמא, עבור מטריצת החברויות הבאה:

| | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | | 1 | 1 | 0 |
| 1 | 1 | | 1 | 0 |
| 2 | 1 | 1 | | 1 |
| 3 | 0 | 0 | 1 | |

קבוצת החברים {0,1,2} היא קליקה בגודל 3, וקבוצת החברים {2,3} היא קליקה בגודל 2, אך לא קיימת קליקה בגודל 4 (כי לדוגמא 3 ו-0 לא חברים).

הערות:

- `N` מוגדר ע"י `#define`.
- יש להשתמש בשיטת `backtracking` כפי שנלמדה בכיתה.
- בשאלה זו אין דרישות סיבוכיות, אולם כמקובל ב-`backtracking` יש לוודא שלא מתבצעות קריאות רקורסיביות מיותרות עם פתרונות שאינם חוקיים.

```
int hasClique(int friends[][N], int k)
```

```
{
```



הטכניון, מכון טכנולוגי לישראל מבוא למדעי המחשב מ'/ח'

[illegible]

13



הטכניון, מכון טכנולוגי לישראל מבוא למדעי המחשב מ'/ח'

[illegible]