

מבחן בקורס CS 1001.py

סמסטר ב' התשע"א, מועד ב'

תאריך: 4.9.2011

מרצה: פרופ' בני שור

מתרגל: רני הוד

מומלץ לקרוא את כל ההנחיות והשאלות בתחילת המבחן, לפני תחילת כתיבת התשובות.

- משך הבחינה שלוש שעות.
- חומר עזר מותר: שני דפי A4, כתובים משני הצדדים.
- במבחן חמש שאלות שלחלקן סעיפי משנה. כדי לקבל ציון 100 בבחינה יש לענות נכונה על כל השאלות. ניקוד כל סעיף מצוין לידו. אין בהכרח קשר בין ניקוד הסעיף ובין קושי.
- את התשובות לשאלה הסגורה יש לסמן בעמוד הראשון של טופס המבחן (עמוד זה). לכל סעיף יש להקיף בעיגול את בודדת (א', ב', ג' או ד') ואין להוסיף הסבר.
- על התשובה לשאלות הפתוחות להופיע במחברת, כל אחת בדף נפרד. יש לענות תשובות ברורות ותמציתיות. תשובות מסורבלות או לא ניתנות לקריאה ע"י הבודקים יזכו לניקוד חלקי בלבד.
- על סעיף של שאלה פתוחה ניתן לענות "אינני יודע/ת" כתשובה; על סעיף זה יינתנו 20% מהנקודות. במקרה זה אין להוסיף שום הסבר.
- בפתרון סעיף בשאלה מותר להשתמש בתוצאות הסעיפים הקודמים, גם אם לא פתרתם אותם.

בהצלחה!

שאלה 1				
1	א	ב	ג	ד
2	א	ב	ג	ד

שאלה 1 (10 נק' לכל סעיף; סה"כ 20 נק')

1. להלן תמליל קצה.

```

Python 3.1.2 (r312:79147, Apr 15 2010, 15:35:48)
[GCC 4.4.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 1
>>> b = 2**100-1
>>> c = 2**100-a
>>> d = b/c
>>> id(b)+id(c)
22759233
>>> id(a)+id(d)
18661952

```

בחרו את האפשרות הנכונה.

- (א) מתקיים `(a is d) and (b is c)`.
- (ב) מתקיים `(a is not d) and (b is c)`.
- (ג) מתקיים `(a is d) and (b is not c)`.
- (ד) מתקיים `(a is not d) and (b is not c)`.

2. ד"ר צפניה הודיע לתקשורת על מחקרו האחרון: בידי אלגוריתם דחיסה חדשני שלוקח כל מחרוזת בינארית באורך n (פרט למחרוזות האפסים) ודוחס אותה (באופן ניתן לשחזור) למחרוזת בינארית קצרה יותר. בחרו את האפשרות הנכונה.

- (א) צפניה הוא רמאי ושקרן! האלגוריתם כפי שתואר אינו אפשרי כלל.
- (ב) צפניה הוא שרלטן ונוכל! האלגוריתם הוא טריוויאלי.
- (ג) צפניה הוא רברבן וגוזמאי! האלגוריתם דגן עובד רק עבור ערכים גדולים מאד של n , וממילא שם החיסכון קטן ולא משמעותי.
- (ד) צפניה הוא מדען פורץ דרך! האלגוריתם ישנה את פני מדעי המחשב ועולם התקשורת הדיגיטלית כפי שאנו מכירים אותם.

שאלה 2 (5 נק' לסעיפים 1,3 ו-10 נק' לסעיף 2; סה"כ 20 נק')

תמורה באורך n היא איברי $\text{range}(n)$ מסודרים בסדר כלשהו. להלן מימוש בסיסי של מחלקת תמורות בפיתון.

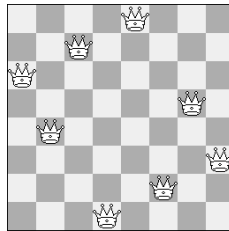
```
class Permutation:
    def __init__(self, perm):
        assert isinstance(perm, list)
        n = len(perm)
        assert sorted(perm) == list(range(n))
        self._perm = perm[:]
    def __getitem__(self, i):
        return self._perm[i]
    def __len__(self):
        return len(self._perm)
```

1. נגדיר את מטריצת התמורה $P = P(\pi)$ המתאימה לתמורה π באורך n כמטריצה ריבועית $n \times n$ שהאיבר במקום i, j בה

הוא $P_{ij} = \begin{cases} 1, & \text{if } \pi[i] = j \\ 0, & \text{if } \pi[i] \neq j \end{cases}$ כיתבו מתודה בשם `to_matrix` שמחזירה Matrix ובו מטריצת התמורה המתאימה ל-`self`. הסבירו במילים וממשו בקוד פייתון.

2. נאמר שתמורה היא מלכותית אם כשנציב n מלכות על-גבי לוח שחמט $n \times n$, בכל אחד מהתאים המסומנים 1 במטריצת התמורה המתאימה, אף מלכה לא מאיימת¹ על צרתה. כיתבו פונקציה יעילה ככל הניתן בשם `is_royal` שמקבלת כקלט תמורה π ובודקת האם היא מלכותית (מחזירה ערך בוליאני). הסבירו במילים וממשו בקוד פייתון.

דוגמא: התמורה $\pi = [4, 2, 0, 6, 1, 7, 5, 3]$ היא מלכותית.



3. נתון לכם גנרטור בשם `perms` שמקבל כקלט מספר טבעי n ומחזיר איטרטור שעובר (בסדר כלשהו) על כל התמורות האפשריות באורך n .

כיתבו פונקציה בשם `prob_royal` שמקבלת כקלט מספר טבעי n ומחזירה מספר ממשי $0 \leq p_n \leq 1$ שהוא ההסתברות שתמורה מקרית² באורך n היא מלכותית. הסבירו במילים וממשו בקוד פייתון.

שאלה 3 (20 נק')

יהופץ וצפניה משחקים במשחק הבא: צפניה בוחר זוג מספרים סודיים $x < y$ מתוך $\text{range}(N)$ (עבור N ידוע מראש) ויהופץ צריך לנחש אותם ע"י שאילת שאלות, כל אחת מהצורה "האם אחד מהמספרים הסודיים הוא $?k$ ". צפניה עונה אחת מבין התשובות הבאות: "שווה" אם $x = k$ או $y = k$; "גדולים" אם $x > k$; "קטנים" אם $y < k$; "באמצע" אם $x < k < y$.

כיתבו פונקציה בשם `guessing_game` שמקבלת כקלט את המספר הטבעי N ומחזירה את זוג המספרים הסודיים. על הפונקציה לשאול כמה שפחות שאלות (במקרה הגרוע ביותר). שאילת שאלה ממומשת ע"י קריאה לפונקציה `tzfania` שמקבלת כקלט את k ומחזירה אחת מבין המחרוזות `'small'`, `'big'`, `'middle'`, `'equal'`.

הסבירו ראשית את האלגוריתם במילים, כיתבו אותו בקוד פייתון ונתחו את מספר השאלות³ במקרה הגרוע ביותר.

¹למאותגרים אשקוקית: מלכה מאיימת על הכלים שנמצאים איתה באותה שורה, באותו טור, או באותו אלכסון.

²כלומר, תמורה שנבחרת מהתפלגות אחידה על כל התמורות האפשריות באורך n .

³לקבלת ניקוד מלא יש לציין את הקבועים ולא להסתפק בסימון $O(\dots)$.

שאלה 4 (5 נק' לסעיפים 1,3 ו-10 נק' לסעיף 2; סה"כ 20 נק')

נתון כרומוזום X המיוצג כמחרוזת באורך $n = 200,000,000$ תווים מעל הא"ב $\Sigma = \{A, C, G, T\}$. הכרומוזום מכיל מספר לא ידוע של צמדי חזרות ארוכים ($k \geq 100$).

שלשת מספרים (i, j, k) תיקרא צמד חזרה אם $0 \leq i < j \leq n - k$ ומתקיים $X[i:i+k] == X[j:j+k]$.
צמד חזרה ייקרא מקסימלי אם הוא לא מוכל בצמד חזרה אחר, כלומר אם $X[i-1] != X[j-1]$ and $X[i+k] != X[j+k]$.
דוגמא: צמדי החזרות המקסימליים מאורך $3 \leq$ עבור $X = \text{'ATCAAGTCAATCA'}$ הם $(0, 9, 4)$, $(1, 6, 4)$, $(6, 10, 3)$.

1. תכננו אלגוריתם יעיל שיחזיר את כל צמדי החזרות המקסימליים מאורך $k \geq 100$ בכרומוזום X והסבירו אותו במילים. מותר להשתמש באלגוריתם הסתברותי, אולם הפלט חייב להיות נכון (קרי: מוחזרים כל צמדי החזרות הארוכים ורק הם) לכל הטלת מטבע.

2. ממשו את האלגוריתם בקוד פייתון. הוסיפו הערות לחלקי הקוד שדורשים הסבר.

3. הניחו שיש m צמדי חזרות ושכולם באורך תלת-ספרתי ($100 \leq k < 1000$). תנו חסם על מספר הפעולות שמבצע האלגוריתם ועל כמות הזכרון הנצרכת (כפונקציה של n ו- m).

שאלה 5 (10 נק' לכל סעיף; סה"כ 20 נק')

1. כיתבו פונקציה בשם `create_paint_by_numbers` שמקבלת מטריצה (אובייקט מטיפוס Matrix) המייצגת תמונה ($l=0$ לבן, 1 =שחור) ומחזירה תיאור של חידת שחור ופתור המתאימה לה; במילים אחרות, הפלט הוא זוג רשימות. הראשונה מציינת את אורכי הרצפים השחורים בכל שורה והשנייה את אורכי הרצפים השחורים בכל טור.

דוגמא: נניח שהמטריצה A , בגודל 3×3 , היא .

```
>>> r, c = create_paint_by_numbers(A)
>>> r
[[], [3], [1, 1]]
>>> c
[[2], [1], [2]]
```

2. להלן שני קידודים אפשריים למטריצות בוליאניות בגודל 7×7 .

(א) `create_paint_by_numbers` מחזיר $14 = 7 + 7$ רשימות, שמקודדות כדלקמן: כל 3 ביטים מייצגים מספר בין 0 ל-7, כאשר 0 מייצג סוף שורה/טור. את קידודי הרשימות משרשרים יחדיו.
דוגמא: הרשימה `[5, 1]` תקודד ע"י `'101001000'`.

(ב) רצף ביטים באורך 49 שהוא שרשור של שבע השורות זו אחר זו.

תארו מטריצות בוליאניות בגודל 7×7 , A ו- B , כך שקידוד (א) של מטריצה A קצר יותר מקידוד (ב) שלה, ואילו קידוד (ב) של מטריצה B קצר יותר מקידוד (א) שלה.