אלגוריתמים 2 – משימה 1

1. (a)

$$\mathbf{2^{340}} \equiv 2^{10*34} \equiv (2^{10})^{34} \equiv 1024^{34} \equiv 1^{34} \equiv \mathbf{1(mod341)}$$

(b)

$$4^{66} \equiv 4^{3*22} \equiv (4^3)^{22} \equiv 64^{22} \equiv 1^{22} \equiv 1(mod21)$$
$$8^{1132} \equiv 8^{2*566} \equiv (8^2)^{566} \equiv 64^{566} \equiv 1^{566} \equiv 1(mod21)$$
$$\Downarrow$$
$$\mathbf{4^{66} - 8^{1132}} \equiv 1 - 1 \equiv \mathbf{0(mod21)}$$

(c) ע״פ האלגוריתם הרקורסיבי שלמדנו בכיתה:

$5^{3001} \equiv$	$5 * 1^2 \equiv \mathbf{5(mod31)}$
$5^{1500} \equiv$	$1^2 \equiv 1(mod31)$
$5^{750} \equiv$	$1^2 \equiv 1(mod31)$
$5^{375} \equiv$	$5 * 5^2 \equiv 1(mod31)$
$5^{187} \equiv$	$5 * 1^2 \equiv 5(mod31)$
$5^{93} \equiv$	$5 * 5^2 \equiv 1(mod31)$
$5^{46} \equiv$	$25^2 \equiv 5(mod31)$
$5^{23} \equiv$	$5 * 25^2 \equiv 25(mod31)$
$5^{11} \equiv$	$5 * 25^2 \equiv 25(mod31)$
$5^{5} \equiv$	$5 * 25^2 \equiv 25(mod31)$
$5^{2} \equiv$	$5^2 \equiv 25(mod31)$
$5^{1} \equiv$	$5 * 1^2 \equiv 5(mod31)$
$5^{0} \equiv$	$1(mod31)$

$12^{301} \equiv$	$12 * 1^2 \equiv \mathbf{12(mod31)}$
$12^{150} \equiv$	$30^2 \equiv 1(mod31)$
$12^{75} \equiv$	$12 * 24^2 \equiv 30(mod31)$
$12^{37} \equiv$	$12 * 8^2 \equiv 24(mod31)$
$12^{18} \equiv$	$15^2 \equiv 8(mod31)$
$12^{9} \equiv$	$12 * 28^2 \equiv 15(mod31)$
$12^{4} \equiv$	$20^2 \equiv 28(mod31)$
$12^{2} \equiv$	$12^2 \equiv 20(mod31)$
$12^{1} \equiv$	$12 * 1^2 \equiv 12(mod31)$
$12^{0} \equiv$	$1(mod31)$

$$\Downarrow$$
$$\mathbf{5^{3001} \not\equiv 12^{301}(mod31)}$$

.2

1. $for\ k\ in\ range(2, \lfloor \log N \rfloor):$
2. $\quad l = 1, h = N$
3. $\quad while(l + 1 < h):$
4. $\quad\quad a = \left\lfloor \frac{l+h}{2} \right\rfloor$
5. $\quad\quad if(a^k == N): return\ True$
6. $\quad\quad else\ if(a^k > N): h = a$
7. $\quad\quad else: l = a$
8. $return\ False$

<u>טענה ראשית:</u>

האלגוריתם מחזיר "כן" ⇔ קיימים טבעיים $a, k > 1$ כך ש $a^k = N$.

<u>טענת עזר:</u>

אם קיימים טבעיים $a, k > 1$ כך ש $a^k = N$ אזי $k \leq \lfloor \log N \rfloor$.

<u>הוכחת טענה ראשית:</u>

⇐ טריויאלי.

⇒ קיימים טבעיים $a, k > 1$ כך ש $a^k = N$, אזי לפי טענת העזר $k \leq \lfloor \log N \rfloor$. מכאן, הלולאה בשורה 1 תגיע לחזקה k. בנוסף, הלולאה בשורה 3 מבצעת חיפוש בינארי בטווח האפשרי למציאת הבסיס, ומנכונות אלגוריתם החיפוש הבינארי תגיע לבסיס a. מכיוון ש $a^k = N$, האלגוריתם מחזיר בשורה 5 "כן".

<u>הוכחת טענת העזר:</u>

נניח בשלילה כי $k > \lfloor \log N \rfloor$. מכיוון ש k שלם ניתן להסיק כי $k > logN$. מכאן ש $2^k > 2^{logN} = N$, ולכן לכל טבעי $a > 1$ מתקיים $a^k > N$ בסתירה לכך שקיימים טבעיים $a, k > 1$ כך ש $a^k = N$.

<u>ניתוח זמן ריצה:</u>

הלולאה בשורה 1 רצה בזמן $O(logN) = O(n)$,
הלולאה בשורה 3 רצה בזמן $O(logN) = O(n)$ (חיפוש בינארי),
הפעולות בשורות 4-7 רצות בזמן $O(logN) = O(n)$ (חיבור\השוואה בין 2 מספרים בני $O(n)$ ביטים).

בסה"כ זמן ריצה האלגוריתם הינו $\boldsymbol{O(n^3)}$.

3. <u>תיאור האלגוריתם:</u>

נריץ את אלגוריתם *Miller-Rabin* על הקלט $C = 2^s * t + 1$ (*):

a. אם בשלב הראשון קבלנו כי $a^{2^s*t} \not\equiv 1(modC)$ נחזיר $\gcd(a, C)$.

b. אחרת, אם בשלב $0 \le r < s$ קבלנו כי $a^{2^s*t} \equiv x(modC)$:

b.1. אם $x \ne 1, C - 1$ נחזיר $\gcd(x - 1, C)$.

b.2. אחרת, אם $x = C - 1$ נחזיר "לא יודע".

c. אחרת, נחזיר "לא יודע".

(*) ע"פ קריאה באינטרנט ניתן להסיק מקריטריון קורסלט שכל מספרי קרמייקל הם אי זוגיים.

<u>טענה ראשית:</u>

בהינתן מספר קרמייקל C האלגו' מחזיר בהסתברות לפחות $\frac{3}{4}$ גורם לא טריויאלי של C.

<u>טענת עזר:</u>

בהינתן C פריק ו $x < C - 1$ שלם,
אם $(x-1)(x+1) \equiv 0(modC)$ אזי C, $(x-1)$ אינם זרים.

<u>הוכחת טענה ראשית:</u>

יהי מספר קרמייקל C. ע"פ משפט שנלמד בכיתה לפחות $\frac{3}{4}$ מהמספרים בטווח שממנו נבחר a הם עדים לפריקות של C, עבורם האלגוריתם מחזיר תשובה. נניח ונבחר a כזה.
אם בשלב הראשון קבלנו כי $a^{2^s*t} \not\equiv 1(modC)$, ע"פ הגדרת מספרי קרמייקל נסיק כי a, C אינם זרים ולכן האלגוריתם מחזיר בשורה *a*. גורם לא טריויאלי של C.
אחרת, ע"פ הגדרת "עד לפריקות", בשלב $0 \le r < s$ כלשהו קבלנו כי $a^{2^s*t} \equiv x(modC)$ כאשר $x \ne 1, C - 1$. ע"פ תיאור האלגוריתם, ניתן להסיק כי $x^2 \equiv 1(modC)$, מכאן ש $x^2 - 1 \equiv 0(modC)$ וע"פ פירוק לגורמים נקבל כי $(x-1)(x+1) \equiv 0(modC)$. ע"פ טענת העזר, $C, (x-1)$ אינם זרים ולכן האלגוריתם מחזיר בשורה *b.1* גורם לא טריויאלי של C.

בסה"כ הראינו כי האלגוריתם מחזיר בהסתברות לפחות $\frac{3}{4}$ גורם לא טריויאלי של C.

<u>הוכחת טענת עזר:</u>

נסתכל על הפירוק של C לגורמים ראשוניים $C = p_1 p_2 *** p_l$.
$(x-1)(x+1) \equiv 0(modC)$ לכן קיים k טבעי כך ש $(x-1)(x+1) = k p_1 p_2 *** p_l$.
נניח בשלילה כי $\gcd(x-1, C) = 1$, לכן $p_1, p_2, \dots, p_l$ גורמים של $(x+1)$ ומכאן ש $(x+1) \ge C$ בסתירה לכך ש $x < C - 1$.

<u>ניתוח זמן ריצה:</u>

ע"פ תיאור האלגוריתם זמן ריצתו הינו זמן הריצה של אלגוריתם מילר רובין $O(n^3)$ ועוד זמן הריצה של אלגוריתם אוקלידיס $O(n^3)$, בסה"כ $\boldsymbol{O(n^3)}$ (כאשר n הינו מספר הביטים של C).

4. <u>תיאור האלגוריתם:</u>

a. הרץ $ext-GCD(e,p-1)$ למציאת d, ההפכי של e מודולו $p-1$.
b. חשב $(m^e)^d(modp)$.

<u>טענה ראשית:</u>

עבור p ראשוני, e זר ל $p-1$ ו d ההפכי של e מוד $p-1$, מתקיים לכל $m\in\{0,\dots,p-1\}$ כי $(m^e)^d\equiv m(modp)$.

<u>הוכחת טענה ראשית:</u>

d הינו ההפכי של e מוד $p-1$, לכן $e*d=1(mod(p-1))$, משמע קיים k טבעי כך ש $e*d=1+k(p-1)$.

יהי $m\in\{0,\dots,p-1\}$.

$m^{e*d}\equiv m^{1+k(p-1)}\equiv m*(m^{p-1})^k\equiv m(modp)$,

כשהמעבר האחרון ע״פ משפט פרמה הקטן.

<u>ניתוח זמן ריצה:</u>

נסמן n מספר הביטים של e,p.

a. ע״פ ניתוח שנראה בכיתה $O(n^3)$.
b. ע״פ ניתוח שנראה בכיתה $O(n^3)$.

סה״כ זמן ריצה של $\boldsymbol{O(n^3)}$.