תרגיל כיתה מס' 2

מספרים בינאריים בעלי סימן ונקודה צפה

הגדרה: הספרה המשלימה 'd של הספרה d בבסיס b היא d'=b-1-d

דוגמא: הספרה המשלימה של 8 בבסיס 10 היא 1.

ישנן מספר שיטות ליצוג מספרים חיוביים ושליליים:

שיטת גודל וסימן, Sign and Magnitude

בשיטה זאת כל מספר מכיל סיבית סימן שמסמנת אם המספר הוא שלילי או חיובי.

b_{N-1}	b_{N-2} ... b_0
sign	Magnitude (ערך מוחלט של המספר)

$$sign = \begin{cases} 0 - positive \\ 1 - negative \end{cases}$$

באופן כללי: $$A = (1-2b_{N-1})\sum_{i=0}^{N-2} b_i 2^i$$

כאשר A הוא הערך במספרים עשרוניים.

דוגמאות: $1111 \to -7, \quad 0111 \to 7$

טבלת ערכים עבור 3 סיביות:

מספר בינארי	ערך (ללא סימן)	Sign/ Magnitude
000	0	0
001	1	1
010	2	2
011	3	3
100	4	0
101	5	-1
110	6	-2
111	7	-3

טווח הייצוג:

$$-(2^{N-1}-1) \le A \le 2^{N-1}-1$$

יתרונות השיטה:

- קלה לביצוע

חסרונות השיטה:

- סיבוך פעולות חשבון
- ייצוג כפול לאפס "-0"= 100…0, "+0"=000…0

שיטת המשלים לאחד, 1's Complement

גם בשיטה זאת כל מספר מכיל סיבית סימן שמסמנת אם המספר הוא שלילי או חיובי אך הפעם שאר הסיביות מייצגות את הערך הבינארי של המספר או את המשלים לערך הבינארי.

b_{N-1}	b_{N-2} ... b_0
sign	אם חיובי – הערך הבינארי של המספר אם שלילי – המשלים של הערך הבינארי של המספר

$$sign = \begin{cases} 0 - positive \\ 1 - negative \end{cases}$$

<u>באופן כללי</u>: $A = \sum_{i=0}^{N-2} b_i 2^i - b_{N-1}(2^{N-1} - 1)$

כאשר A הוא הערך במספרים עשרוניים.

<u>דוגמאות</u>: $1000 \rightarrow -7, \quad 0111 \rightarrow 7$

טבלת ערכים עבור 3 סיביות:

מספר בינארי	ערך (ללא סימן)	1's complement
000	0	0
001	1	1
010	2	2
011	3	3
100	4	-3
101	5	-2
110	6	-1
111	7	0

<u>טווח הייצוג</u>:

$$-(2^{N-1} - 1) \le A \le 2^{N-1} - 1$$

<u>יתרונות השיטה</u>: חיבור/חיסור פשוטים:

- <u>חיבור</u>: כמו חיבור בינארי רגיל, בשינוי הבא:
  אם יש carry out מהספרה האחרונה- נוסיף 1 לתוצאה ונזרוק את ה- carry out הזה.
  צריך להוסיף אחד כיוון שישנם שני ייצוגים לאפס.
- <u>חיסור</u>: חיבור של המספר הנגדי.

<u>חסרונות השיטה</u>:

- ייצוג כפול לאפס "-0"= 111…1, "+0"=000…0

<u>דוגמאות לחיבור וחיסור</u>:

$$\begin{array}{r} 0011 \\ + \\ 1010 \\ \hline 1101 \end{array} \; (\rightarrow 0010) \quad \Leftrightarrow \quad \begin{array}{r} 3 \\ + \\ -5 \\ \hline -2 \end{array}$$

היפוך

$$\begin{array}{r} 1100 \\ + \\ 1011 \\ \hline \cancel{1}0111 \\ + \\ 1 \\ \hline 1000 \end{array} \; (\rightarrow 0111) \quad \Leftrightarrow \quad \begin{array}{r} -3 \\ + \\ -4 \\ \hline -7 \end{array}$$

היפוך

<u>שיטת המשלים לשתיים, 2's Complement</u>

גם בשיטה זאת כל מספר מכיל סיבית סימן שמסמנת אם המספר הוא שלילי או חיובי אך הפעם שאר הסיביות מייצגות את הערך הבינארי של המספר או את המשלים לערך הבינארי ועוד אחד. התוספת של 1 נועדה למנוע את הייצוג הכפול של אפס ב- 1's Complement.

b_{N-1}	b_{N-2} … b_0
sign	אם חיובי – הערך הבינארי של המספר אם שלילי – המשלים של הערך הבינארי של המספר פלוס 1

$$sign = \begin{cases} 0 - positive \\ 1 - negative \end{cases}$$

באופן כללי: $$A = \sum_{i=0}^{N-2} b_i 2^i - b_{N-1} 2^{N-1}$$

כאשר A הוא הערך במספרים עשרוניים.

טבלת ערכים עבור 3 סיביות:

2's complement	ערך (ללא סימן)	מספר בינארי
0	0	000
1	1	001
2	2	010
3	3	011
-4	4	100
-3	5	101
-2	6	110
-1	7	111

<u>דוגמאות</u>: $1001 \rightarrow -7, \quad 0111 \rightarrow 7$

<u>טווח הייצוג</u>:

$$-2^{N-1} \le A \le 2^{N-1} - 1$$

יתרונות השיטה :

- חיבור/חיסור פשוטים : אם יש carry out פשוט מתעלמים ממנו.
- ייצוג בודד לאפס.

דוגמאות לחיבור וחיסור :

11011		-5	
+		+	
11010		-6	
$\cancel{1}$10101	⇔	-11	
-			
1			
10100	(→ 01011)		

00111		7	
+		+	
10111		-9	
11110	⇔	-2	
-			
1			
11101	(→ 00010)		

כללים לחיבור :

א) חיבור שני מספרים שליליים עלול להביא לתוצאה שגויה (גלישה).
לדוגמא :

$$(-4)+(-3)=100+101=\cancel{1}001=1 \neq -7$$

ב) אם מחברים שני מספרים חיוביים (סיבית סימן =0) ומקבלים תוצאה שלילית (סיבית סימן =1) – שגיאה.
לדוגמא :

$$3+3=011+011=110=-2 \neq 6$$

ג) חיבור מספר שלילי עם מספר חיובי נותן תמיד תוצאה נכונה.

ד) מקבלים carry רק אם לפחות אחד משני המספרים הוא שלילי.

<u>ייצוג מספרים שאינם שלמים</u> :

<u>טווח הייצוג</u> מגביל את התחום של הערכים שניתן לייצג בכל שיטה. כאשר פעולה חשבונית בין מספרים בתחום הייצוג נותנת תוצאה מחוץ לטווח הייצוג נגיד שיש גלישה.

<u>רזולוציה</u> של שיטת ייצוג היא ההפרש בין המספרים הקרובים ביותר באותה שיטת ייצוג. בשיטות הייצוג הבינאריות שראינו, תמיד קיבלנו רזולוציה אחידה 1 (מספרים שלמים).

<u>Fixed point</u> :

בשיטת ייצוג זאת משקל הסיבית הנמוכה ביותר יכול להיות קטן מאחד, כלומר ניתן לייצג גם שברים, אבל הנקודה העשרונית נמצאת תמיד במקום קבוע. הרזולוציה בשיטת ייצוג זאת שווה למשקל הסיבית הנמוכה ביותר (LSB).

לדוגמא :

1	0	1	1	0	1.	0	1
$x2^5$	$x2^4$	$x2^3$	$x2^2$	$x2^1$	$x2^0$	$x2^{-1}$	$x2^{-2}$

$$(101101.01)_2=45.25$$

בדוגמא זאת הרזולוציה היא 2^{-2} כיוון ששני המספרים הקרובים ביותר הם בהפרש של 2^{-2} אחד מהשני.

יש לשים לב שכיוון שעל ידי N ספרות בינאריות נוכל לייצג 2^N ערכים שונים לכל היותר, הרזולוציה באה על חשבון טווח הייצוג ולהיפך.

<u>נקודה צפה (floating point)</u>

S	E		M
Sign	Sign	Exponent	Mantisa

סימן המנטיסה

סימן האקספוננט

$$A = M \cdot 2^E$$

בשיטת ייצוג זאת הרזולוציה ניתנת לקביעה ע״י שינוי ערך האקספוננט.

לדוגמא, אם E= -1 אז שני המספרים הקרובים ביותר יכולים להיות

$$\underline{2^{-1}} = 1\cdot 2^{-1}, \underline{0} = 0\cdot 2^{-1}$$

כלומר הרזולוציה היא 2^{-1}.

ואם E= 0 אז שני המספרים הקרובים ביותר יכולים להיות

$$\underline{1} = 1\cdot 2^{0}, \underline{0} = 0\cdot 2^{0}$$

כלומר הרזולוציה היא $2^0=1$.

<u>יתרונות השיטה</u>:

1. תחום דינמי גבוה (טווח ייצוג גבוה)
2. פעולות כפל/חילוק פשוטות

<u>חסרונות השיטה</u>:

1. רזולוציה לא אחידה
2. פעולות חיבור/חיסור מסובכות (בגלל שהרזולוציה לא אחידה)

<u>קודים לקידוד ספרות</u>

<u>תזכורת מתחילת התרגול</u>:

הספרה המשלימה 'd של הספרה d בבסיס b היא d'=b-1-d.

לדוגמה, בבסיס 10 (עשרוני) הספרה המשלימה של 9 היא 0 ושל 3 היא 6.

הקודים משמשים לייצוג של ספרות. קודים שונים נוחים למטרות שונות.

<u>קודים בינאריים</u>: (נתייחס לקידוד ספרות בבסיס 10)

הגדרה: ייצוג בינארי מקיים:

א. לכל ספרה 9....0 קיימת מילת קוד שונה (רצוי שתהיה יחידה, אך לא הכרחי).

ב. קיימת נוסחה או טבלת מעבר: ספרה $\leftrightarrow$ קוד.

<u>קודים משוקללים</u>:

קודים משוקללים משתמשים ב-positional notation אך הפעם המשקול יכול להיות שונה מאשר למדנו בתרגולים קודמים. כל ספרה מיוצגת ע"י 4 סיביות x_4, x_3, x_2, x_1 ולכל סיבית יש משקל w_i, כלומר ערך הספרה המיוצגת הינו:

$$N = \sum_{i=1}^{4} w_i x_i = w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4$$

<u>דוגמאות לקודים משוקללים</u>:

ספרה / משקלים	8 4 2 1	2 4 2 1	6 4 2 -3
	קוד BCD		
0	0 0 0 0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1	0 1 0 1
2	0 0 1 0	0 0 1 0	0 0 1 0
3	0 0 1 1	0 0 1 1	1 0 0 1
4	0 1 0 0	0 1 0 0	0 1 0 0
5	0 1 0 1	1 0 1 1	1 0 1 1
6	0 1 1 0	1 1 0 0	0 1 1 0
7	0 1 1 1	1 1 0 1	1 1 0 1
8	1 0 0 0	1 1 1 0	1 0 1 0
9	1 0 0 1	1 1 1 1	1 1 1 1

קוד BCD בטבלה לעיל (Binary Coded Decimal) הוא הנפוץ ביותר ליצוג של מספרים עשרוניים. לכל ספרה עשרונית מותאם קוד של ארבע סיביות. ניתן בקלות לפרק מספר כלשהו לקבוצות סיביות כך שכל קבוצה מיצגת ספרה עשרונית אחרת. אילו היינו משתמשים בבסיס בינארי רגיל, לא היינו יכולים להפריד באופן נוח בין הספרות העשרוניות השונות.

<u>קודים לא משוקללים</u> :

למשל קוד Excess-3 מתקבל מ- BCD ע"י הוספת 0011 לכל מלת קוד :

ספרה	Excess-3
0	0 0 1 1
1	0 1 0 0
2	0 1 0 1
3	0 1 1 0
4	0 1 1 1
5	1 0 0 0
6	1 0 0 1
7	1 0 1 0
8	1 0 1 1
9	1 1 0 0

<u>קודים המשלימים את עצמם</u> :

קוד משלים את עצמו אם "המשלים ל-9" של <u>כל</u> ספרה מתקבל ע"י הפיכת אחדים לאפסים ולהיפך.

<u>דוגמאות</u> :

← קוד 2421 משלים את עצמו :

הספרה '0' היא ה"משלים ל-9" של הספרה '9' : $9+0=10-1$

ואכן : $9'=(1111)'=0000=0$

הספרה '3' היא ה"משלים ל-9" של הספרה '6' : $6+3=10-1$

ואכן : $6'=(1100)'=0011=3$

← אך קוד BCD <u>אינו</u> משלים את עצמו :

לדוגמה, $9'=(1001)'=0110=6$ אך : $9+6\neq 10-1$

<u>תנאי הכרחי</u> לכך שקוד משוקלל ישלים את עצמו הוא שסכום משקליו יהיה 9.

(הוכיחו משפט זה בבית. רמז : $x'=1-x$).

<u>קודים ציקליים</u> :

קוד שבו מילת קוד של ספרה כלשהי שונה ממילת הקוד של הספרה שלפניה ואחריה בסיבית אחת בלבד.

<u>לדוגמא קוד Gray</u> :

קוד באורך n=3

מילת קוד	ספרה
000	0
001	1
011	2
010	3
110	4
111	5
101	6
100	7

קוד באורך n=2

מילת קוד	ספרה
00	0
01	1
11	2
10	3

לקוד גריי שימושים רבים, וחלקם נראה בהמשך הקורס (לדוגמה, שימוש לצמצום פונקציות מיתוג, שימוש למניעת הבהובים סטטיים ועוד).

נניח ש- $g_n g_{n-1} \ldots g_0$ היא מילה בקוד Gray ו- $b_n b_{n-1} \ldots b_0$ הוא המספר הבינארי המתאים.

<u>המרה מבינארי ל- Gray</u>

<u>binary</u> <u>Gray</u>

$b_3 : 1 \xrightarrow{1} 1 : g_3$

$\downarrow$

$b_2 : 0 \overset{1\oplus 0}{\overline{\qquad}} 1 : g_2$

$\downarrow$

$b_1 : 0 \overset{0\oplus 0}{\overline{\qquad}} 0 : g_1$

$\downarrow$

$b_0 : 1 \overset{0\oplus 1}{\overline{\qquad}} 1 : g_0$

$$\begin{cases} g_n = b_n & , i = n \\ g_i = b_i \oplus b_{i+1} = \left(b_i + b_{i+1}\right) \bmod 2 & , i = (n-1),(n-2),....,1,0 \end{cases}$$

<u>המרה מ- Gray לבינארי</u> :

אם מספר ה- 1-ים משמאל לסיבית ה- i –ית זוגי, אזי $b_i=g_i$ אחרת $b_i=g_i'$

לדוגמא :

Gray 1110100 => 101$\underline{1}$000 binary

במקום המודגש שמנו אחד כיוון שמשמאלו בקוד Gray יש מספר אי-זוגי של אחדים (3) ובמקום הזהה לו בקוד Gray יש אפס, לכן הופכים את האפס לאחד.