

## הנחיות כלליות

יש לשלוח את הקבצים באמצעות [מערכת ההגשה](#) לפני חלוף התאריך **24/12/19**.

ניתן להגיש את התרגיל באיחור עם קנס אוטומטי על פי הפירוט הבא:

- יום איחור - קנס של **5 נקודות** (ציון מקסימלי - 95).
- יומיים איחור - קנס של **15 נקודות** (ציון מקסימלי - 85).
- שלושה ימי איחור - קנס של **30 נקודות** (ציון מקסימלי - 70).

לאחר מכן לא יהיה ניתן להגיש את התרגיל (ציון 0).

המתרגל האחראי על התרגיל הוא אייל.

שאלות בנוגע לתרגיל יש לפרסם **באופן ציבורי בפורום הקורס** בלבד! רק אם לא התקבלה תשובה לאחר 24 שעות, יש לשלוח מייל לכתובת [eyal.dayan@biu.ac.il](mailto:eyal.dayan@biu.ac.il) עם קישור לדיון הרלוונטי.

בקשות להארכה מסיבות מוצדקות (מילואים, לידה וכו') יש לפרסם **באופן פרטי בפורום הקורס** בלבד (יש למען את הפוסט ל-instructors). בכל בקשה יש לציין שם מלא, שם משתמש במערכת ההגשה, תעודת זהות והאם אתם ממדעי המחשב או מתמטיקה.

יש להקפיד מאוד על הוראות עיצוב הקלט והפלט, בדיוק על פי הדוגמאות המצורפות. **שימו לב** להנחיות במסמך ה-Coding Style המפורסם באתר הקורס.

עליכם לכתוב קוד על פי ההנחיות ולוודא שקיבלתם 100 בבדיקה האוטומטית הראשונית, וכן שהתרגיל מתקמפל ורץ על שרתי המחלקה (u2) ללא **שגיאות** או **אזהרות**. תרגיל שלא עומד בסטנדרטים הבסיסיים הללו יגרור **ירידה משמעותית בציון התרגיל**, בשל הטרחה שהוא מייצר בתהליך הבדיקה שלו.

להזכירכם העבודה היא אישית. "עבודה משותפת" דינה כהעתקה. התרגיל נבדק על ידי מערכת ההגשה האוטומטית גם מהבחינה הזו, ותרגיל שהועתק יגרור ציון 0 **לכל הגורמים** השותפים בהעתקה. אתם יכולים לדון בגישות לפתרון התרגיל באופן תיאורטי, אך אין לשתף קוד בשום צורה.

בפיתוח הקוד ניתן להשתמש בכל סביבת עבודה, העיקר הוא שתדעו איך לקחת את קבצי הקוד מתוך הסביבה הזו, לבדוק אותם על שרתי האוניברסיטה ולהגיש אותם באמצעות מערכת ההגשה. דוגמאות לחלק מהסביבות האפשריות:

IDEs (Integrated Development Environment):

- Visual studio
- Clion
- Eclipse
- Xcode

Text Editors:

- Atom
- Sublime
- Notepad++
- Vim

בהצלחה!

## תרגיל 4 – ass4

בתרגיל זה עליכם לממש פונקציות בקובץ יחיד בשם `ass4.c`

משקל התרגיל מתוך ציון התרגול: 20%.

- לתרגיל זה מצורף קובץ בשם `ass4.h` המכיל הצהרות, וקובץ בשם `main.c` המכיל פונקציית `main`. אין להגדיר פונקציית `main` בקובץ `ass4.c` שאתם מגישים. בנוסף, אין צורך להגיש את הקבצים המצורפים. הקבצים הנוספים מיועדים לבדיקות אישיות שלכם.
- בתרגיל זה עליכם להשתמש בקבוע `define` חיצוני בשם `SIZE`, ערכו יכול להיות חד ספרתי.
- בתרגיל זה מותר להשתמש בספריות `stdio.h`, `stdlib.h`, `string.h`, `ctype.h`, `math.h`, `assert.h`. יש להוסיף לפקודת הקימפול את הדגל `-lm` לצורך קישור הספרייה `math.h`.

פקודת הידור לדוגמה (כי ערכו של `SIZE` יכול להיות אחר) עבור התרגיל:

```
gcc ass4.c main.c -std=c99 -lm -DSIZE=8
```

בקובץ `output.txt` מצורף הפלט עבור הקוד המופיע בקובץ `main.c` (כאשר הפונקציות ממומשות).

הערה: בקובץ `public_code.c` מופיעות כמה פונקציות נוספות, קבועים והגדרות. אין חובה להשתמש בקוד הזה, אך אפשר לעשות זאת אם אתם רוצים.

בתרגיל זה עליכם לממש מערכת המאפשרת להציג באופן ויזואלי משחק שחמט, המקודד בעזרת שתי שיטות קידוד מקובלות.

קישורים שימושיים:

- [מדריך למשחק שחמט](#) סעיפים 1-7 (מומלץ למי שלא מכיר את המשחק בכלל). אין צורך לדעת לשחק שחמט, אך נדרשת הבנה של החוקים הבסיסיים - תנועת הכלים, הגדרת שח. בתרגיל זה אין צורך לתמוך בהצרח או ב"הכאה דרך הילוכו".
- [אתר](#) המאפשר לנתח ולהציג משחקי שחמט, על פי הקידודים הטקסטואליים המוצגים במסמך זה.

**שיטת FEN**

**בשיטה זו** המטרה היא לתאר מצב לוח בעזרת מחרוזת תווים.  
לכל כלי שחמט תוגדר אות אנגלית על פי הטבלה הבאה:

שחור (lowercase)	לבן (UPPERCASE)	כלי
k	K	מלך (King)
q	Q	מלכה (Queen)
r	R	צריח (Rook)
b	B	רץ (Bishop)
n	N	פרש (Knight)
p	P	רגלי (Pawn)

כדי לייצג משבצת ריקה נשתמש בספרה 1.  
אם יש כמה משבצות ריקות צמודות באותה שורה נייצג את כולן יחד בעזרת ספרה אחת המייצגת את סכומן.  
ייצוג של שורה יורכב מרצף של אותיות וספרות משמאל לימין.  
על מנת לתאר את כל הלוח:

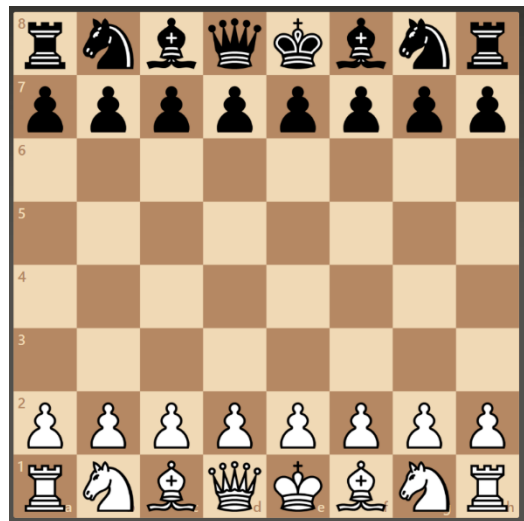
1. נסתכל על הלוח כך ששורה 1 היא התחתונה, ושורה 8 העליונה.  
עמודה a היא השמאלית ועמודה h היא הימנית.
2. נבנה את המחרוזת המתארת כל שורה, משורה 8 עד שורה 1.
3. נשרשר אותן עם התו '/' (לכסן ימני) ביניהן.

**לדוגמה:**

המחרוזת הראשונה מתארת את עמדת הפתיחה המקובלת (תמונה שמאלית).  
המחרוזת השנייה מתארת את מצב המשחק כמה מהלכים לאחר מכן (תמונה ימנית).

rnbgkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR

r1bqkb1r/pppp1ppp/2n2n2/1B2p3/4P3/2N2N2/PPPP1PPP/R1BQK2R



**שיטת PGN**

**בשיטה זו** המטרה היא לתאר **משחק שלם**, בעזרת תיאור **כל מהלך** לפי מספר מאפיינים.

קידוד המהלכים נעשה באופן מצומצם כדי להיות קצר ככל האפשר ולא להעמיס במידע מיותר, לכן הוא מכיל ברובו רכיבים אופציונליים, שנעשה בהם שימוש רק כאשר הדבר הכרחי.

מאפיין	אופן הקידוד	סימון מקוצר	אופציונלי
איזה כלי מבצע את המהלך	אות גדולה יחידה מתוך הקבוצה [KQRBN]. אם לא צוין כלי, מדובר ברגלי	[KQBNR]	כן
מאיזו עמודה מגיע הכלי שמבצע את המהלך	אות קטנה יחידה מתוך הקבוצה [abcdefgh]	[a-h]	כן
מאיזו שורה מגיע הכלי שמבצע את המהלך	ספרה יחידה מתוך הקבוצה [12345678]	[1-8]	כן
האם מדובר במהלך שכולל הכאה (Capture) של כלי יריב	האות הקטנה x	[x]	כן
לאיזו עמודה מגיע הכלי שמבצע את המהלך	אות קטנה יחידה מתוך הקבוצה [abcdefgh]	(a-h)	לא
לאיזו שורה מגיע הכלי שמבצע את המהלך	ספרה יחידה מתוך הקבוצה [12345678]	(1-8)	לא
אם מדובר ברגלי (Pawn) שהגיע אל השורה האחרונה שלו (שורה 8 ללבן ושורה 1 לשחור), ולאיזה כלי הוא מוכתר	סימן '=' (שיוויון) ואות יחידה מתוך הקבוצה [QRBN] (לא ניתן להכתיר מלך נוסף או להימנע מהכתרה)	[=(QBNR)]	כן
האם המהלך כולל איום של "שח" או של "מט"	איום שח יסומן על ידי סימן '+' (חיבור). איום מט יסומן על ידי סימן '#' (סולמית)	[+ #]	כן

כללים:

- תנועה של רגלי תסומן על ידי נקודת היעד (למשל e4).
- תנועה של כלי תסומן על ידי אות גדולה ונקודת היעד (למשל Qd2)
- כאשר לא ברור איזה משני כלים זהים (צריח, פרש, רץ או מלכה) ביצע את התנועה, יופיע שם העמודה או מספר השורה המבחין ביניהם (או שניהם) לאחר שם הכלי (למשל Rad1, R1a3).
- כאשר כלי מכה כלי אחר, תופיע האות x לפני שם המשבצת אליה הוא נע. כאשר הכלי המכה הוא רגלי, תופיע עמודת המקור באות קטנה בתור "שם הכלי" (למשל Bxd3, exd5).
- הכתרה של רגלי תסומן בתוספת = והאות המציינת את הכלי שאליו מוכתר הרגלי (למשל b8=Q).
- שח יסומן בתוספת + (למשל Nf6+).
- מט יסומן בתוספת # (למשל c8=N#).

ניתן לתאר את כל המחרוזות האפשריות באופן הבא:

[KQBNR][a-h][1-8][x](a-h)(1-8)[=(QBNR)][+ #]

רכיבים אופציונליים מוקפים בסוגריים מרובעים, רכיבי חובה בסוגריים עגולים.  
רכיב ההכתרה הוא "חצי חובה" – אם מופיע סימן '=', אז חייבים לכתוב אחריו את אחת האותיות.

ההבחנה בין מהלך של השחקן הלבן למהלך של השחקן השחור נעשית לפי הסדר – ראשון משחק הלבן ואז השחור וכן הלאה.

דוגמה למהלכים שהתרחשו בתמונות למעלה (מספרי המהלכים מצורפים לצורך נוחות, אך אינם הכרחיים):

1. e4 e5
2. Nf3 Nc6
3. Nc3 Nf6
4. Bb5

התרגיל

בקובץ `public_code.c` מופיעה פונקציה המקבלת מחרוזת FEN ומדפיסה את הלוח.

```
void printBoardFromFEN(char fen[]);
```

לדוגמה, הפונקציה מדפיסה את הלוח המיוצג על ידי המחרוזת שהובאו למעלה.

*		A	B	C	D	E	F	G	H		*							
*	-----										*							
8		r		n		b		q		k		b		n		r		8
7		p		p		p		p		p		p		p		p		7
6																		6
5																		5
4																		4
3																		3
2		P		P		P		P		P		P		P		P		2
1		R		N		B		Q		K		B		N		R		1
*	-----										*							
*		A	B	C	D	E	F	G	H		*							

*		A	B	C	D	E	F	G	H		*							
*	-----										*							
8		r				b		q		k		b				r		8
7		p		p		p		p				p		p		p		7
6						n				n								6
5				B				p										5
4								P										4
3						N				N								3
2		P		P		P		P				P		P		P		2
1		R				B		Q		K						R		1
*	-----										*							
*		A	B	C	D	E	F	G	H		*							

בתרגיל עליכם לממש שלוש פונקציות (חתימותיהן מופיעות בקובץ `ass4.h`):

```
void createBoard(char board[][SIZE], char fen[]);
```

```
void printBoard(char board[][SIZE]);
```

```
void makeMove(char board[][SIZE], char pgn[], int isWhiteTurn);
```

הפונקציה `createBoard` מקבלת מערך תווים לא מאותחל בגודל `SIZE*SIZE` ומחרוזת FEN. על הפונקציה להשתמש במערך כדי לשמור את הנתונים המקודדים במחרוזת.

הפונקציה `printBoard` מקבלת מערך תווים שאותחל בעזרת הפונקציה `createBoard`. על הפונקציה להדפיס אל המסך בדיוק את מה שמדפיסה הפונקציה `printBoardFromFEN` עבור אותה מחרוזת FEN שבעזרתה אותחל המערך.

הפונקציה `makeMove` מקבלת מערך תווים שאותחל בעזרת הפונקציה `createBoard`, מחרוזת PGN, ומספר המצוין האם זהו התור של הלבן (1 אם כן, 0 אם לא). במידה ומדובר במהלך חוקי, על הפונקציה לשנות את מצב הלוח בהתאם. במידה ומדובר במהלך שאינו חוקי, על הפונקציה להדפיס הודעה ולא לשנות את הלוח. ייתכן שהפונקציה תיקרא עם מערך שנשלח לה כמה פעמים ברצף (כמו שאפשר לראות בפונקציית `main`).