

## תרגיל 3-מערכים דו מימדיים – להגשה עד ה-27.12 בשעה

23:50

הוראות ההגשה ודגשים מיוחדים **מופיעים בסוף התרגיל! חובה לקרוא!**  
בתרגיל זה שלושה חלקים. עליכם לממש את כל מה שכתוב בתרגיל. שימו לב כי מדובר בשלוש תוכנות שונות (אחד לכל חלק) וכל חלק ייכתב בקובץ נפרד עם main נפרד. כל שלושת הקבצים יועלו למערכת.

שימו לב כי חלק שלוש (צוללות) הוא כולו תוכנית אחת(!) - פונקציה main אחת – שמפעילה הרבה פונקציות אחרות! **חישבו היטב כיצד לחלק את התוכנית לפונקציות כדי לסייע לקלות המימוש והקריאות של הקוד (כל פעולה סמנטית היא פונקציה נפרדת!)**

### חלק א – מערך (10%):

כתבו פונקציה `int equal(int arr1[], int arr2[], int size)`. הפונקציה מקבלת כפרמטרים שני מערכים של שלמים ואת גודלם המשותף. היא מחזירה את כמות האינדקסים בהם המערכים שווים.

הפונקציה לא משנה את הפרמטרים שלה.  
לדוגמא, אם המערכים הם:  $\{1,2,300,4,50,6\}$ ,  $\{1,20,300,40,5,6\}$  אז הפונקציה תחזיר 3.  
כתבו את הפונקציה הראשית בתכנית. הפונקציה קולטת מהמשתמש 2 מערכים בגודל 8, מעבירה אותם לפונקציה `equal`, ומדפיסה את הערך המוחזר.  
**שימו לב:** מותר ואף רצוי לחלק את הקוד לפונקציות נוספות.

### חלק ב – מחרוזת (10%):

כתבו פונקציה `int words(char str[], int length)`. הפונקציה מקבלת כפרמטרים מחרוזת ומספר שלם. היא מדפיסה את כל המילים במחרוזת שאורכן שווה לפרמטר השני. רווח הוא תו מפריד בין מילים במחרוזת. הפונקציה מחזירה כערך את מספר מילים שהדפיסה, אך לא משנה את המחרוזת.

לדוגמא, עבור הפרמטרים `length=4`, `str[]="one two three four five six";` יודפס:  
four five

והפונקציה תחזיר 2.

כתבו את הפונקציה הראשית בתכנית. הפונקציה קולטת מהמשתמש מחרוזת בגודל של עד 20 תווים, מעבירה אותה לפונקציה `words`, ומדפיסה את הערך המוחזר.  
**שימו לב:** מותר ואף רצוי לחלק את הקוד לפונקציות נוספות.

## חלק ג – משחק צוללות (80%):

בחלק זה בתרגיל אתם מתבקשים לממש את המשחק "צוללות".

### הגדרת המשחק מויקיפדיה:

- לכל משתתף במשחק יש לוח עם שני ריבועים בגודל 10X10 משבצות. הריבוע האחד מסמל את השטח ה"ימי" שלו, בו הוא מציב את הצוללות שלו כאוות נפשו. ואילו הריבוע השני מסמל את השטח ה"ימי" של צוללות יריביו, ומשמש מעקב לגילויים.
- הצבת הצוללות נעשית לפני תחילת המשחק **בלבד**, ואין להזיז אותן מרגע תחילתו. ישנם כמה חוקים לפיהם מציבים את הצוללות:
  1. יש להציב את הצוללות במאונך או במאוזן, ולא באלכסון.
  2. אין לחפוף את הצוללות (כלומר, שאחת תהיה מוצבת על השנייה, או צמודה לה).
- לכל שחקן יש 5 צוללות:
  1. צוללת אחת בגודל 5 משבצות
  2. צוללת אחת בגודל 4 משבצות
  3. שתי צוללות בגודל 3 משבצות כל אחת
  4. צוללת אחת בגודל 2 משבצות

### המימוש שלנו

אתם נדרשים לממש חלקית את המשחק.

הרעיון בגדול:

1. למחשב יהיה לוח ובו מוצבות הצוללות.
2. בכל תור המשתמש יצטרך להכניס קואורדינטות (קודם שורה ואח"כ תור) שאותם הוא מבקש "לתקוף". במידה ולא היה שם כלום המחשב ידפיס "Miss", במידה והייתה שם צוללות המחשב ידפיס "Hit".
3. כמו כן בסיום כל שלב יודפס הלוח עם תוצאות התקיפות עד כה למסך.
4. המשחק ימשך עד שמשתמש יצליח להטביע את כל הצוללות או עד שהוא יכניס את הערכים 1-1-1 (כדי לסמן רצון לצאת).
4. במידה והמשתמש הצליח להטביע הכל יודפס למסך:

Congratulation you win!

ואז התוכנה תסתיים. אם המשתמש ביקש לצאת יודפס למסך:

See you next time!

והתוכנה תסתיים.

## פירוט על כל סעיף:

### חלק ראשון - בניית הלוח:

הלוח בגודל 10X10. הצוללות (כל ה-5) יוצבו בו בצורה רנדומית (להוראות איך להשתמש בפונקציות רנדומית – עיינו בתרגיל השני) **לפי הכללים**.

אתם צריכים לממש זאת כך:

1. לכל צוללת המחשב יגריל:

a. ערך למספר השורה (בין 0 ל-9)

b. וערך למספר העמודה (בין 0 ל-9)

c. ואז יגריל עוד ערך האם הצוללת תוצב במאוזן (כך שהמקום שנבחר הוא הקצה

השמאלי ביותר) או במאונך (כך שהמקום שנבחר הוא הקצה העליון ביותר).

2. כעת יש לבדוק האם ניתן להציב את הצוללת במיקום ובצורה שנבחרו ועדיין לשמור

על הכללים. אם כן הצוללת תוצב שם, במידה ולא נחזור ל-1 ונחפש לאותה צוללת

מיקום אחר (בצורה רנדומית כמובן).

לאחר שסיימנו להציב את כל הצוללות – הלוח מוכן וניתן להתחיל במשחק עצמו.

### חלק שני – מהלך המשחק:

במהלך המשחק יש תורות. בכל תור המשתמש צריך להכניס שני מספרים שלמים – מספר שורה ומספר עמודה.

אם הערכים הם 1- ו-1- המשחק מסתיים (כפי שמוזכר למעלה). אחרת:

אם הערכים לא חוקיים (שלילים או חורגים מגודל המטריצה) – יש להדפיס למשתמש:

Illegal values, please try again.

ולחזור שוב לקליטת ערכים.

כאשר הערכים חוקיים יש לבדוק האם יש לנו צוללת מוצבת באותו מיקום או לא. אם כן יש להדפיס Hit (ולשמור לנו שהמיקום הנ"ל כבר נפגע), אחרת יש להדפיס Miss (ולסמן לנו שהמיקום הנ"ל נבדק).

בסיום התור תודפס למסך מטריצת התוצאות עד כה כפי שהיא ידועה למשתמש, זאת אומרת:

- מקום שהמשתמש טרם תקף (ולכן לא יודע מה יש בו) - יודפס התו רווח.
- מקום שהמשתמש תקף ואין בו כלום (ים) – יודפס התו ~
- מקום שהמשתמש תקף והייתה בו צוללת – יודפס התו X.

### חלק שלוש – סיום המשחק:

המשחק מסתיים כאשר המשתמש הצליח להטביע את כל הצוללות או כאשר הוא ביקש במפורש לצאת (הקליד 1 -1). ובהתאם לזה תודפס ההודעה המתאימה והתוכנה תסתיים.

### סעיף בונוס (10%)

הגדרה:

ה-Hit שפוגע בתא האחרון שעדיין לא נפגע מאותה צוללת (כל שאר התאים כבר נפגעו, וזה האחרון שנשאר) הוא ה-Hit שגורם לטביעת הצוללת.

בסעיף זה יש להוסיף לתוכנית הקודמת את התמיכה בזיהוי הטבעה של כל צוללת בנפרד. זאת אומרת שכאשר יש Hit שגורם להטבעה של צוללת יש להדפיס למסך:

Congratulation you just sink a submarine with # cells.

(כמובן שיש להחליף את # במספר התאים המתאים שהיה לאותה צוללת).

למשל להטבעה של הצוללת עם 5 התאים יודפס:

Congratulation you just sink a submarine with 5 cells.

כל שאר המערכת אמורה לעבוד בדיוק כמו שהוגדר בשאר התרגיל.

לצורך סעיף זה **בלבד** מותר להיעזר המצביעים (שימו לב כי כל שימוש נוסף במצביעים מעבר לזה בתרגיל הוא אסור!).

## הערות:

1. שימו לב שהקלט מהמשתמש הוא מספרים שלמים. אתם יכולים להניח שאנו אכן נכניס מספרים שלמים. אך לא בהכרח ערכים חוקיים (אינדקסים חוקיים) – באחריותכם לוודא את חוקיות הקלט!
2. אחרי כל הדפסה יש לבצע ירידת שורה (`std::endl` בסוף ההדפסה).
3. **אין להשתמש בתרגיל בחומר שנלמד לאחר נושא התרגיל אלא אם נכתב במפורש בתרגיל שמותר!**
4. יש להקפיד על תכנות נכון:
  - a. כל הערכים שהם קבועים, זאת אומרת שמבחינה לוגית הם לא אמורים להשתנות, חייבים להיות מוגדרים כ: `const`, `define` או `enum`, בהתאם לצורך, אין זה נכון להשתמש בקבועים מספריים ("magic numbers").
  - b. יש לרשום הערות בשפע! לכל תוכנית מה היא מבצעת (אפשר לראות דוגמא בפתרון תרגיל 2), כל פעולה לא טריוויאלית להסביר, כל לולאה ותנאי להסביר מה המטרה שלהם.
  - c. יש לנסות ולייעל את הקוד והתוכנית ככל שניתן (לא להוסיף פקודות או לולאות מיותרות, לא לרוץ בלולאה מעבר למה שצריך, וכו').
  - d. לפני כל בקשת קלט (`std::cin`) יש להדפיס למשתמש הוראה (`std::cout`) איזה קלט מבוקש.
  - e. יש להקפיד לבצע חלוקה לפונקציות. אין לכתוב קוד כפול! נסו לחשוב איזה חלוקה תיתן לכם קוד קריא יותר וקל יותר לתיחזוק ושימוש מחודש!
  - f. יש להקפיד על הזחות, וכיתוב נכון וקריא!
  - g. יש להקפיד על כל כללי התכנות הנכון כפי שנלמדו בכיתה.
5. בהצלחה ☺

## הנחיות הגשה

- תרגילים הם ביחידים! כל עבודה משותפת היא אסורה ותיענש בחומרה!
- ההגשה היא של קובץ הקוד (כקובץ `.cpp`). בלבד, שישלח **אך ורק** דרך מערכת התיקשוב.
- בתחילת הקובץ יש להוסיף את התיעוד הבא:

```
/* Assignment: 3
   Author: Israel Israeli, ID: 01234567
*/
```

כמובן שיש לעדכן את השמות ומספרי תעודות הזהות שלכם.