

## תרגיל 3 – פונקציות, מערכים ומחרוזות

1. מועד אחרון להגשה: 25/12/2020 23:55
2. ערך התרגיל: 30%.
3. יש להגיש את התרגיל לפני התאריך הנקוב באמצעות האתר:  
<http://submit.eng.biu.ac.il/cgi-bin/welcome.cgi>  
כל יום איחור בהגשת התרגיל מוריד 25 נקודות.
4. שם התרגיל: ex3
- שם הקובץ שעליכם להגיש:** ex3.cpp (הסיומת היא cpp).
5. יש להקפיד מאוד על כל הוראות עיצוב הקלט והפלט, כמפורט בכל סעיף וסעיף. על הפלט להיראות בדיוק כמו בדוגמאות. אין להוסיף או להשמיט רווחים או תווים אחרים ואין להחליף אותיות גדולות בקטנות או להיפך. אי הקפדה על פרטים אלו עלולה לגרור ירידה משמעותית ביותר בציון התרגיל עד כדי 0. ראו הוזהרתם!
6. להזכירכם, העבודה היא אישית. "עבודה משותפת" דינה כהעתקה, בנוסף חל איסור גמור לראות קוד של חבר או להחזיק קוד של חבר אצלכם במחשב. **נא לא להתעסק עם העתקות!**
7. במסך המשתמש אתם תראו גם את הקלטים וגם את הפלטים, אך אל דאגה - כאשר אנו נריץ נראה רק את הפלטים. לכן וודאו כי הפלט מופיע במסך בפורמט הנדרש.
8. אין להשתמש במשתנים גלובליים או סטטיסטיים, אלא אם כן רשום במפורש שמותר.
9. **הנחיות קודינג סטייל:** א. שמות המשתנים חייבים להיות בעלי משמעות. ב. לפני כל חלק של התפריט יש להוסיף הערה קצרה מה יש בחלק זה. ג. בכל מקום בו יש קלט, יש להוסיף הערה מהו הקלט החוקי. ד. יש לרשום בתחילת הקובץ בהערה את השם המלא שלכם כולל תעודת זהות.

הנחיות קלט: בתרגיל זה מותר להניח שמזון קלט חוקי. שימו לב – ישנם מקומות בהם יש לשנות את הקלט המתקבל.

### הנחיות מימוש:

- פונקציית ה-main תראה כך:

```
int main()  
{  
    void ourApp();  
    return 0;  
}
```

מעבר לכך אסור להוסיף כלום, ואת כל ניהול התרגיל יש לבצע מתוך הפונקציה ourApp.

- בפונקציה ourApp עליכם לקרוא מספר פונקציות כמפורט בהמשך התרגיל, ומלבדן מותר להוסיף רק: הצהרה על משתנים, קריאה לפונקציות, שימוש בלולאה לצורך ניהול המשחקים, וניהול קלט ופלט. הפונקציה תהיה **לכל היותר 20 שורות**.

- בראשית הקובץ יש להוסיף את שתי השורות הבאות:

```
#define MAX_SIZE 30  
#define MIN_SIZE 5
```

הערה לכל התרגיל: בסיום כל הדפסה יש לרדת שורה.

## חלק 1א – קליטת שם המשתמש

ראשית יש לממש פונקציה בשם `printNames`. בתוך הפונקציה יש לקבל מהשתמש שתי מחרוזות בנפרד – אחת עבור השם הפרטי, ואחת עבור שם המשפחה. המחרוזות יהיו משתנים סטטיים כדי שתוכלו אח"כ להדפיס אותן שוב בסיום התוכנית.

- יש להניח שכל מחרוזת מכילה לכל היותר 10 תווים.

לאחר מכן יש להדפיס:

```
Hello firstName lastName, welcome to our app!
```

כאשר `firstName` הוא השם הפרטי, ו-`lastName` הוא שם המשפחה.

הפונקציה אינה צריכה להחזיר שום ערך, וניתן להכניס לה ערכים לפי הצורך.

## חלק 1ב – בחירת גודל הלוח

כעת התוכנית תדפיס למסך את התפריט:

```
Let's choose the size:
```

כעת המשתמש יכניס בנפרד שני מספרים שלמים. המספר הראשון יהיה מספר השורות בטבלה, והמספר השני יהיה מספר העמודות. ניהול הקלט:

- מותר להניח שהמספר הראשון חיובי. אם הוא גדול מ-`MAX_SIZE`, יש לשנות אותו ל-`MAX_SIZE`, ואם הוא קטן מ-`MIN_SIZE`, יש לשנות אותו ל-`MIN_SIZE`.
- מותר להניח שהמספר השני יהיה חיובי או -1. אם הוא -1, יש לתת לו את הערך של המספר הראשון. אחרת, אם הוא גדול מ-`MAX_SIZE`, יש לשנות אותו ל-`MAX_SIZE`, ואם הוא קטן מ-`MIN_SIZE`, יש לשנות אותו ל-`MIN_SIZE`.

את המימוש של קליטת המשתנים יש לממש בשתי פונקציות נפרדות, כאשר הפונקציה הראשונה תחזיר את מספר השורות בטבלה, והפונקציה השנייה תחזיר את מספר העמודות בטבלה. עליכם להתאים את חתימת הפונקציה והפרמטרים אותה היא מקבלת לפי הצורך.

## חלק 1ג – ניהול המשחק

בשלב זה נעבור למשחקים.

← יש להדפיס למשתמש:

```
Do you want to play [y/n]?
```

אם המשתמש לוחץ על `n`, יש להדפיס (בעזרת הפונקציה `printNames`):

```
Bye, firstName lastName.
```

ולסיים את התוכנית.

אם הוא לוחץ על `y` יש לבצע ברצף את שני המשחקים המפורטים בהמשך, ולאחר מכן לחזור לשורה המתחילה ב-`←`.

את הניהול של חלק 1ג יש לבצע בתוך הפונקציה `ourApp`.

## חלק 2 – שוטרים וגנבים

בחלק זה נממש משחק הנקרא שוטרים וגנבים – כל המימוש יהיה בתוך פונקציה שנקראת מתוך ourApp. בתוך הפונקציה של מימוש המשחק מותר לקרוא לעוד פונקציות, אך זה לא חובה וניתן לממש את כל המשחק בתוך הפונקציה.

בראשית המשחק יש לשאול את המשתמש:

How many cops (1-5)?

בנוסף לשוטרים, יש גנב אחד. מיקומו ההתחלתי של הגנב הוא במרכז הלוח (אם מספר השורות או העמודות הוא אי זוגי, יש לעגל כלפי מטה). שימו לב שאם לדוגמא מספר השורות הוא 10, אז השורות ממוספרות מ-0 עד 9, ולכן השורה האמצעית היא שורה בעלת אינדקס 4 ולא 5! לכן אם הלוח הוא בגודל  $10 \times 10$ , אז הגנב יתחיל במיקום:

GameTable[4][4]

כעת יש לשבץ את השוטרים על הלוח. עבור כל שוטר נדפיס:

Let's choose a cell:

ולאחר מכן נקלוט שני מספרים, כאשר המספר הראשון יהיה עבור השורה והמספר השני יהיה עבור העמודה. במידה ומשבצת זו כבר תפוסה על ידי שוטר אחר או על ידי הגנב, יש לקבל שורה ועמודה **ללא קלט נוסף**. לאחר שנבחרו כל המקומות, יש להדפיס:

Well, Let's play!  
Initial states:

ולאחר מכן להדפיס את הלוח.

### מהלך המשחק:

השוטרים משחקים ראשונים.

לפני כל תור, יש להדפיס שורה האומרת של מי התור הנוכחי:

Cops: / Rob:

← אם תור השוטרים, יש להדפיס את כל התנועות החוקיות האפשריות. תנועה חוקית מוגדרת כתזוזת שוטר אחד משבצת אחת למעלה/למטה/ימינה/שמאלה, בתנאי שאין שם שוטר אחר. לשוטר מותר לזוז למשבצת בה נמצא הגנב. פורמט ההדפסה הוא (יש להדפיס שורה נפרדת עבור כל מהלך חוקי):

[col, row] -> [col, row]

סדר התנועות בהדפסה יהיה החל מהשוטר הכי שמאלי ועד השוטר הכי ימני, וכאשר יש שני שוטרים באותה עמודה, יש לקחת קודם את השוטר העליון יותר.

עבור כל שוטר, יש להדפיס את התנועות לפי הסדר (כמובן שיש להדפיס רק תנועות חוקיות): תנועה למטה, למעלה, ימינה, שמאלה (דוגמא ניתן לראות בהרצה שבנספח).

כעת המשתמש יכניס בנפרד ארבעה מספרים, שניים עבור השורה והעמודה של השוטר אותו רוצים להזיז, ושניים עבור מיקומו החדש. לאחר בחירת המהלך, יש להדפיס את הלוח. אם במהלך זה אחד השוטרים זז למשבצת בה נמצא הגנב, אז בהדפסת הלוח יש להדפיס במשבצת זו C ולא R, ולאחר מכן לסיים את המשחק בצירוף הדפסה:

The cops won!

← אם תור הגנב, עליו ללכת למשבצת בה הוא מתרחק כמה שיותר מהשוטר הקרוב אליו ביותר. מרחק יחושב על ידי נורמת  $\ell_2$ , כלומר: נניח והגנב נמצא במשבצת  $(x_R, y_R)$ , והשוטרים נמצאים במשבצות  $(X_{C_1}, Y_{C_1}) \dots (X_{C_n}, Y_{C_n})$ , אזי יש למצוא את השוטר הקרוב ביותר, ולהתרחק ממנו. דגשים:

- אם יש כמה שוטרים שנמצאים במרחק זהה מהגנב, יש להתייחס לשוטר בעל ערך השורה הקטנה ביותר מבין השוטרים הנ"ל. אם עדיין אין הכרעה, יש להתייחס לשוטר בעל ערך העמודה הקטנה יותר.
- אם ערך השורה של השוטר הקרוב ביותר קטן מערך השורה של הגנב, על הגנב לרדת משבצת אחת למטה. אם ערך השורה של השוטר הקרוב ביותר גדול מערך השורה של הגנב, על הגנב לעלות משבצת אחת למעלה. אם השוטר והגנב באותה שורה, על הגנב ללכת לכיוון הנגדי מהשוטר.

- אם תנועת הגנב אינה אפשרית משום שהמשבצת המיועדת תפוסה על ידי שוטר אחר או היא סוף הלוח, הגנב לא זז ומסתיים תורו (כאשר יש להדפיס שוב את הלוח, למרות שלא התבצעה תנועה).

#### מקרה מיוחד:

אם יש רק שוטר אחד, יש לחשב את סכום העמודות והשורות של השוטר והגנב. אם סכום זה זוגי, יש לגמור את המשחק ולהדפיס את הלוח בצירוף ההודעה שהגנב ניצח. אחרת, יש להמשיך את המשחק כרגיל. לדוגמא: אם השוטר מתחיל במיקום (0,0) והגנב מתחיל במיקום (4,4), אז סכום השורות והעמודות הוא 8, ויש לסיים באופן מידי את המשחק.

הנחיות קלט: מותר להניח שהמשתמש הכניס ערכים חוקיים:

- אכן נבחרו 1-5 שוטרים.
- עבור כל שוטר נבחרה שורה בין 0 לבין מספר השורות בטבלה, ונבחרה עמודה בין 0 לבין מספר העמודות בטבלה.
- כאשר משבצים את המיקום ההתחלתי של השוטרים, יתכן ויוזן קלט חוקי אך משבצת זו כבר תהיה תפוסה, לכן למרות שזה קלט חוקי, יש לקבל שוב ערכים ללא תוספת קלט.
- כאשר מוזנים מספרים עבור תנועת שוטר, מותר להניח שהמהלך חוקי (כלומר, הוא אחד מהמהלכים האפשריים שהדפסתם קודם לכן).

הנחיות פלט:

- את הלוח יש להדפיס לפי הפורמט הבא: כל משבצת ריקה תסומן על ידי '-'. משבצת שיש בה שוטר תודפס על ידי 'C', ומשבצת שיש בה גנב תודפס על ידי 'R'. לדוגמא – אם הלוח הוא בגודל  $5 \times 5$  ויש שני שוטרים במיקומים (0,0), (1,1) וגנב במשבצת (4,4), ההדפסה תראה כך:

```
C----
-C---
-----
-----
-----R
```

- אם עברו 30 מהלכים ללא הכרעה, המשחק מסתיים בניצחון של הגנב.
- אם השוטרים תפסו את הגנב יש להדפיס:

The cops won!

אחרת יש להדפיס:

The rob managed to escape!

## חלק ב2 – ארבעת הפרשים

בחלק זה נממש משחק הנקרא ארבעת הפרשים – כל המימוש יהיה בתוך פונקציה שנקראת מתוך ourApp. בתוך הפונקציה של מימוש המשחק מותר לקרוא לעוד פונקציות, אך לא חובה וניתן לממש את כל המשחק בתוך הפונקציה.

### במשחק זה הלוח הוא בגודל $3 \times 3$ .

בראשית המשחק יוצבו שני פרשים לבנים בפינות העליונות של הלוח, ושני פרשים שחורים בפינות התחתונות של הלוח. תנועת הפרשים היא כמו [במשחק שחמט](#).

בתחילת המשחק יש להדפיס את הלוח עם המיקומים ההתחלתיים, כך:

W-W

---

B-B

מטרת השחקנים: לחסום את היריב כך שלא יוכל לזוז, למשל, המהלך הבא הוא מהלך מנצח לשחור, כיוון שהוא מונע כל תזוזה מהלבן:

B	-	W
W	-	-
-	-	B

→

-	-	W
W	-	-
-	B	B

אם חולפים 30 מהלכים ללא הכרעה, המשחק מסתיים בתיקו, ומודפס:

Tie

מהלך המשחק:

הלבן משחק ראשון.

לפני כל תור, יש להדפיס שורה האומרת של מי התור הנוכחי:

W: / B:

בכל תור, המשתמש יכניס ארבעה מספרים, שניים עבור השורה והעמודה של הפרש אותו רוצים להזיז, ושניים עבור מיקומו החדש. לאחר בחירת המהלך, יש להדפיס את הלוח. אם במהלך זה ליריב נחסמת כל אפשרות לזוז, יש לסיים את המשחק בצירוף הדפסה:

The whites won! / The blacks won!

מותר להניח שהמהלכים המוזנים הם חוקיים.

**בהצלחה!**

```
Moshe
Cohen
Hello Moshe Cohen, welcome to our app!
Let's choose the size:
5
5
Do you want to play [y/n]?
y
How many cops (1-5)?
2
Let's choose a cell:
1
1
Let's choose a cell:
2
3
Well, Let's play!
Initial states:
-----
-C---
--RC-
-----
-----
Cops:
[1,1] -> [2,1]
[1,1] -> [0,1]
[1,1] -> [1,2]
[1,1] -> [1,0]
[2,3] -> [3,3]
[2,3] -> [1,3]
[2,3] -> [2,4]
[2,3] -> [2,2]
2
3
3
3
-----
-C---
--R--
---C-
-----
Rob:
-----
-C---
-----
--RC-
-----
Cops:
[1,1] -> [2,1]
[1,1] -> [0,1]
[1,1] -> [1,2]
[1,1] -> [1,0]
[3,3] -> [4,3]
[3,3] -> [2,3]
[3,3] -> [3,4]
[3,3] -> [3,2]
1
1
```

```

2
1
-----
-----
-C---
--RC-
-----
Rob:
-----
-----
-C---
-R-C-
-----
Cops:
[2,1] -> [3,1]
[2,1] -> [1,1]
[2,1] -> [2,2]
[2,1] -> [2,0]
[3,3] -> [4,3]
[3,3] -> [2,3]
[3,3] -> [3,4]
[3,3] -> [3,2]
2
1
3
1
-----
-----
-----
-C-C-
-----
The cops won!
W-W
---
B-B
W:
0
0
1
2
--W
--W
B-B
B:
2
2
1
0
--W
B-W
B--
W:
0
2
2
1
---
B-W
BW-
B:
1

```

```
0
0
2
--B
--W
BW-
W:
1
2
0
0
W-B
---
BW-
B:
2
0
1
2
W-B
--B
-W-
The blacks won!
Do you want to play [y/n]?
n
Bye, Moshe Cohen.
```