

סדנת תכנות בשפת C, מס' קורס 67319 - 2018

תרגיל 2

מערכים דו-מימדיים, זיכרון דינאמי, Makfile

תאריך הגשה: 23:55 16/08/2018

הגשה מאוחרת (בהפחתת 10 נקודות): 23:55 17/08/2018

תאריך ההגשה של הבוחן: 23:55 16/08/2018

הנחיות חשובות לכלל התרגילים:

1. בכל התרגילים יש לעמוד בהנחיות הגשת התרגילים וסגנון כתיבת הקוד. שני המסמכים נמצאים באתר הקורס – הניקוד יכלול גם עמידה בדרישות אלו.
2. בכל התרגילים עליכם לכתוב קוד ברור. בכל מקרה בו הקוד שלכם אינו ברור מספיק עליכם להוסיף הערות הסבר בגוף הקוד. יש להקפיד על תיעוד (documentation) הקוד ובפרט תיעוד של כל פונקציה.
3. במידה ואתם משתמשים בעיצוב מיוחד או משהו לא שגורתי, עליכם להוסיף הערות בקוד המסבירות את העיצוב שלכם ומדוע בחרתם בו.
4. עבור כל פונקציה בה אתם משתמשים, עליכם לוודא שאתם מבינים היטב מה הפונקציה עושה גם במקרי קצה (התייחסו לכך בתיעוד). ובפרט עליכם לוודא שהפונקציה הצליחה.
5. בכל התרגילים במידה ויש לכם הארכה, או שאתם מגישים באיחור. חל איסור להגיש קובץ כלשהוא בלינק הרגיל (גם אם לינק overdue טרם נפתח). מי שיגיש קבצים בשני הלינקים מסתכן בהורדת ציון משמעותית.
6. אין להגיש קבצים נוספים על אלו שתדרשו. ובפרט אין להגיש קובץ README אלא אם צוין במפורש שיש צורך בכך (לדוגמא, בתרגיל זה אין צורך להגיש).
7. עליכם לקמפל עם הדגלים `Wall -Wextra -Wvla -std=c99` ולוודא שהתוכנית מתקמפלת ללא אזהרות, תכנית שמתקמפלת עם אזהרות תגרור הורדה משמעותית בציון התרגיל. למשל, בכדי ליצור תוכנית מקובץ מקור בשם `ex1.c` יש להריץ את הפקודה:
`gcc -Wextra -Wall -Wvla -std=c99 ex1.c -o ex1`
8. עליכם לוודא שהתרגילים שלכם תקינים ועומדים בכל דרישות הקימפול והריצה במחשבי בית הספר מבוססי מעבדי bit-64 (מחשבי האקווריום, לוי, השרת river). חובה להריץ את התרגיל במחשבי בית הספר לפני ההגשה. (ניתן לוודא שהמחשב עליו אתם עובדים הנו בתצורת bit-64 באמצעות הפקודה `uname -a` ויודא כי הארכיטקטורה היא 64, למשל אם כתוב `x86_64`)
9. לאחר ההגשה, בדקו את הפלט המתקבל בקובץ ה-PDF שנוצר מהpresubmission script בזמן ההגשה. באם ישנן שגיאות, תקנו אותן על מנת שלא לאבד נקודות.
10. שימו לב! תרגיל שלא יעבור את הpresubmission script ציונו ירד משמעותית (הציון יתחיל מ-50, ויוכל לרדת) ולא יהיה ניתן לערער על כך.
11. בדיקת הקוד לפני ההגשה, גם על ידי קריאתו וגם על ידי כתיבת בדיקות אוטומטיות (tests) עבורו היא אחריותכם. בדקו מקרי קצה. במידה וסיפקנו לכם קבצי בדיקה לדוגמא, השימוש בהם יהיה על אחריותכם. במהלך הבדיקה הקוד שלכם ייבדק מול קלטים נוספים לשם מתן הציון.
12. הגשה מתוקנת - לאחר מועד הגשת התרגיל ירוצו הבדיקות האוטומטיות ותקבלו פירוט על הטסטים בהם נפלתם. לשם שיפור הציון יהיה ניתן להגיש שוב את התרגיל לאחר תיקוני קוד ולקבל בחזרה חלק מהנקודות - פרטים מלאים נמצאים בפורום ואתר הקורס.

משחק צוללות

בתרגיל זה תכתבו משחק צוללות לשחקן אחד נגד מחשב: המחשב יציב צוללות בצורה אקראית והשחקן ינסה לגלות אותן.



עליכם לממש את הלוגיקה של המשחק (ואת מבני הנתונים אותם תצטרכו) בקובץ `battleships.c`, לפי ממשק אותו תשלימו בקובץ `battleships.h`. בנוסף, עליכם לכתוב תוכנה המאפשרת לשחק על ידי ממשק טקסטואלי בקובץ `battleships_game.c`.

מהלך המשחק

התוכנית תקבל בשורת הפקודה את גודל הלוח (הלוח תמיד ריבועי). עליכם לבנות לוח ריבועי ולמלא אותו בכלים הבאים:

- נושאת מטוסים אחת (5 מקומות)
- סיירת אחת (4 מקומות)
- ספינת טילים אחת (3 מקומות)
- צוללת אחת (3 מקומות)
- משחתת אחת (2 מקומות)

הגדילו את מיקום וזווית הכלים בעזרת הפקודה `rand`. בתחילת המשחק יש להדפיס "Ready to play" וירידת שורה. כל מהלך יתקבל בקלט התקני כצירוף של אות לועזית קטנה ומספר עם רווח ביניהם, למשל:

a 1
i 5
z 22

כאשר האות מייצגת את השורה והמספר מייצג את העמודה.

לאחר כל שורת קלט, אם מדובר במהלך תקין (מורכב מאות לועזית, רווח, ומספר חיובי ונמצא בתוך גבולות הלוח) יודפס האם מדובר בפגיעה ("Hit!") או פספוס ("Miss") וירידת שורה. במקרה של מהלך לא חוקי תודפס הודעת שגיאה: "Invalid move, try again" (וירידת שורה). כאשר כלי הוטבע (כל התאים שלו נפגעו), יודפס "Hit and sunk.\n". לאחר הדפסת תוצאות המהלך, יש להדפיס את הלוח הפגיעות של השחקן: קו תחתון ('_') עבור משבצת ריקה (שלא נוסתה), איקס עבור פגיעה ('x') ועיגול עבור פספוס ('o'). כאשר המשחק מסתיים (כל הכלים הוטבעו), יודפס "Game over" והתוכנית תסתיים. בנוסף, אם יתקבל הקלט "exit" התוכנית תדפיס "Game over" ותצא.

Makefile

עליכם להוסיף Makefile התומך בפעולות הבאות:

- make - מהדר את הקבצים ומייצר את ex2.exe.
- make battleships – מייצר את battleships.o
- make clean – מסיר את כל הקבצים הזמניים ואת התוכנית הבנויה.

מידע נוסף (כללי)

1. חל איסור להשתמש במערכים בגודל דינמי (VLA).
2. עליכם לוודא שהקוד שלכם רץ באופן תקין וללא דליפות זכרון. לשם כך עליכם להשתמש בתוכנת valgrind.
3. אתם רשאים להשתמש בכל הספריות הסטנדרטיות של C.
4. אתם רשאים (ולעתים אף נדרשים) להגדיר פונקציות נוספות לשימושכם הפנימי.
5. שימו לב שאתם מכירים כל פונקציה בה אתם משתמשים ושאתם בודקים עבור כל פונקציה שהיא הצליחה, ובכלל זאת הקצאות זיכרון.

טיפול בשגיאות:

- הדפסות שגיאה יודפסו אל stderr. שימו לב שעל התוכנה לצאת בצורה מסודרת בכל מקרה של בעיה (exit gracefully).
- עבור מספר ארגומנטים שגוי יש להדפיס מידע המסביר כיצד להריץ את הקוד (usage) ולצאת עם ערך חזרה שונה מ-0.
- מלבד ההנחות הרשומות אין להניח שהקלט תקין - עבור קלט שאינו תקין על התוכנה לצאת באופן מסודר לאחר הדפסת שגיאה אינפורמטיבית כלשהיא (אין פורמט מחייב להודעת השגיאה).

בדיקת התרגיל:

1. התכנית יבדקו גם על סגנון כתיבת הקוד וגם על פונקציונאליות, באמצעות קבצי קלט שונים (תרחישים שונים להרצת התכנית). מכיוון שהחישוב בתרגיל הינו נומרי ונתון לשגיאות עיגול, פתרוכם ישווה עד כדי טווח שגיאה.
2. אם ישנם מקרים שהוראות התרגיל לא מציינות בבירור כיצד התכנית צריכה להתנהג, הביטו בקבצי הקלט וקבצי הפלט לדוגמה שניתנים לכם ובדקו אם התשובה לשאלתכם נמצאת שם. כמו כן, היעזרו בפתרון בית הספר, הריצו עליו את הטסטים שלכם והשוו להתנהגות תוכניתכם. כמובן שניתן וכדאי להתיעץ בפורום לגבי מקרים שבהם התשובה עדיין אינה ברורה.

חומר עזר:

1. את פתרון בית הספר ניתן למצוא ב:
~plabc/www/ex2/school_sol.tar
2. רצוי לבצע "בדיקת שפיות" על הפתרונות שלכם לפני ההגשה, באמצעות שרטוט הפתרונות כפי שמתואר בתרגיל.
3. ספריית math של C לשימושכם גם כן.

הגשה:

1. עליכם להגיש קובץ tar בשם ex3.tar המכיל רק את הקבצים הבאים:
 - battleships.c
 - battleships.h
 - battleships_game.c
 - Makefile

ניתן ליצור קובץ tar כדורש על ידי הפקודה:

```
tar cvf ex2.tar <files to include in tar>
```

2. לפני ההגשה, פתחו את הקובץ ex1.tar בתיקה נפרדת וודאו שהקבצים מתקמפלים ללא שגיאות וללא אזהרות. וודאו שההגשה שלכם עוברת את ה-presubmission script ללא שגיאות או אזהרות.
- ~plabc/www/ex1/presubmit_ex2
3. אתם יכולים להריץ בעצמכם בדיקה אוטומטית עבור סגנון קידוד בעזרת הפקודה:
~plabc/www/codingStyleCheck <code file or directory>
כאשר <directory or file> מוחלף בשם הקובץ אותו אתם רוצים לבדוק או תיקייה שיבדקו כל הקבצים הנמצאים בה (שימו לב שבדיקה אוטומטית זו הינה רק חלק מבדיקות ה codingStyle).
4. דאגו לבדוק לאחר ההגשה את קובץ הפלט (submission.pdf) וודאו שההגשה שלכם עוברת את ה-presubmission script ללא שגיאות או אזהרות.

בהצלחה!