

שפת C – תרגיל 1

היכרות עם השפה, preprocessor, compiler, משתנים, לולאות, תנאים, פונקציות, קלט/פלט

תאריך הגשה: יום חמישי 11.08.16 עד שעה 23:55

הגשה מאוחרת (בהפחתת 10 נקודות): מוצאי שבת עד שעה 23:55¹

תאריך ההגשה של הבוחן: יום חמישי 11.08.16 עד שעה 23:55

1. הנחיות חשובות:

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. בכדי שתוכלו להגיש באיחור עם קנס, עליכם לסמן זאת בלינק התרגיל לפני מועד ההגשה.
4. למי שיש אישור להגשה מאוחרת, אין להגיש בקישור להגשה הרגילה. **מי שיגיש קבצים בשני הלינקים מסתכן בהורדת ציון משמעותית.**
5. אין להגיש קבצים נוספים על אלו שתדרשו.
6. עליכם לקמפל עם הדגלים `Wall -Wextra -Wvla -std=c99` ולוודא שהתוכנית מתקמפלת ללא אזהרות, תכנית שמתקמפלת עם אזהרות תגרור הורדה 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`
7. עליכם לוודא שהתרגילים שלכם תקינים ועומדים בכל דרישות הקימפול והריצה במחשבי בית הספר (מחשבי האקווריום, לוי, השרת river). בפרט, המחשבים שעליהם מתבצעת הבדיקה, צריכים להיות 64 ביט².
8. לאחר ההגשה, בדקו את הפלט המתקבל בקובץ ה-PDF שנוצר מהpresubmission script בזמן ההגשה. באם ישנן שגיאות, תקנו אותן על מנת שלא לאבד נקודות. שימו לב! תרגיל שלא יעבור את הpresubmission script ציונו ירד משמעותית (הציון יתחיל מ-50, ויוכל לרדת) ולא יהיה ניתן לערער על כך.
9. בדיקת הקוד לפני ההגשה, גם על ידי קריאתו וגם על ידי כתיבת בדיקות אוטומטיות עבורו היא אחריותכם. חישבו על מקרי קצה לבדיקת הקוד.

¹ ניתן להגיש באיחור נוסף, עד יום ראשון בקנס של 20 נקודות.

² ניתן לוודא שהמחשב עליו אתם עובדים הנו בתצורת זז באמצעות הפקודה `"uname -a"` ויודא כי הארכיטקטורה היא 64, למשל אם כתוב `x86_64`.

קבצי בדיקה לדוגמה ניתן למצוא פה: `~slabc/www/ex1/tests_examples.tar`
שימוש בקבצים אלו הוא באחריותכם. במהלך הבדיקה הקוד שלכם ייבדק מול קלטים
נוספים לשם מתן הציון.

10. הגשה מתוקנת - לאחר מועד הגשת התרגיל ירוצו הבדיקות האוטומטיות ותקבלו פירוט על
הטסטים בהם התוכנית שלכם נכשלה. לשם שיפור הציון יהיה ניתן להגיש שוב את התרגיל
לאחר תיקוני קוד קלים ולקבל בחזרה חלק מהנקודות - פרטים מלאים מופיעים בהנחיות
הקורס באתר.

2. מידע חשוב נוסף:

1. ניתן להתחבר באמצעות SSH למחשבי בית הספר (למשל לשם בדיקת הקוד לפני הגשה
מהבית)

http://wiki.cs.huji.ac.il/wiki/Connecting_from_outside

2. עליכם להכיר את ספריית הקלט-פלט של שפת C ובייחוד את השימוש בפונקציות `printf` ו
`scanf`

<http://www.cplusplus.com/reference/clibrary/cstdio>

3. הנחיות ספציפיות לתרגיל זה:

1. בתרגיל זה אין להשתמש בספריות כלשהן מלבד `<stdio.h>` ו-`<stdlib.h>`.
2. שימו לב שהאלגוריתמים שלכם צריכים להיות יעילים.
3. אתם רשאים (ולעתים אף נדרשים) להגדיר פונקציות נוספות לשימושכם הפנימי.
4. בתרגיל זה אין להגיש קבצים נוספים.

4. תקינות סוגריים - CheckParenthesis:

1. אחת הבדיקות הראשונות אותן מבצע המהדר (compiler), כאשר אתם מהדרים את תוכניתכם היא בדיקת תקינות המבנה הכללי של התכנית, ותקינות מבנה הסוגריים בפרט.
2. בשאלה זו עליכם לממש בקובץ CheckParenthesis.c גרסה פשוטה של תכנית בדיקת תקינות סוגריים.
3. בהינתן קובץ קלט המורכב ממספר שורות. תוכניתכם תבדוק אם מבנה הסוגריים בקובץ תקין או לא. במידה והוא תקין תכניתכם תתדפיס:

```
ok\n
```

במידה והוא לא תקין, תכניתכם תתדפיס:

```
bad structure\n
```

הרצת התוכנית תתבצע בצורה הבאה:

```
>CheckParenthesis <file name>
```

4. הדרכה והנחיות כלליות:

- (1) עליכם להתמודד עם סוגי הסוגריים: (), [], {}, ו-<>.
- (2) אתם יכולים להניח שבקובץ לכל היותר 5000 זוגות סוגריים.
- (3) אתם יכולים להניח שאורך שורה בקובץ קטן או שווה 1000.
- (4) הקובץ יכול להכיל סוגריים מקוננים. וסוגריים עשויים להתחיל בשורה אחת ולהסתיים באחרת. כמו כן, הקובץ עשוי להכיל תווים נוספים.
- (5) במקרה של תקלות, על התכנית שלכם להדפיס הודעת שגיאה מתאימה לפלט השגיאות הסטנדרטי ולצאת עם ערך החזרה שונה מ-0:
 - בעיה בפתיחת הקובץ -

```
Error! trying to open the file %s\n
```

- כך ש-s% מוחלף בשם הקובץ שקיבלתם כפרמטר.
- עליכם לוודא כי מספר הפרמטרים הוא 1 בדיוק: -

```
Please supply a file!\n
```

```
usage: CheckParenthesis <file name>\n
```

5. המרת בסיס - ChangeBase:

1. בשאלה זו עליכם לממש בקובץ ChangeBase.c המרת בסיסים.
2. תכניתכם תקבל כקלט מהמשתמש מחרוזת המכילה מספר כלשהו, בסיס בו הוא מיוצג והבסיס אליו מעוניינים להעביר את המספר. פלט התכנית יהיה המספר בבסיס החדש וירידת שורה.
3. פורמט הקלט הוא:

`<original base>^<new base>^<number in original base>^`

4. לדוגמא, עבור הקלט הבא:

`10^4^932^`

פלט התוכנית יהיה:

`32210\n`

5. הרצת התוכנית תתבצע בצורה הבאה:

`>ChangeBase`

`2^10^1100^`

`12`

6. הדרכה והנחיות כלליות:

- (1) אתם רשאים להניח כי הבסיסים יהיו בין 2 ל-10.
- (2) אתם יכולים להניח כי המספרים אותם תתבקשו להמיר יהיו אי-שליליים ובעלי מספר ספרות קטן או שווה 6.
- (3) אתם יכולים להניח כי המספרים יופיעו ללא אפסים מובילים (לא תצטרכו להתמודד עם מספרים כמו 0123, או 00).
- (4) בשאלה זו אתם יכולים להניח כי הקלט שתקבלו יהיה בהתאם לפורמט (יכיל 3 מספרים, שלאחר כל אחד מהם מופיע התו '^').
- (5) במידה והמספר אותו התבקשתם להמיר, איננו מתאים לבסיס בו הוא מיוצג, עליכם להדפיס הודעת שגיאה לפלט השגיאות הסטנדרטי, ולצאת עם ערך שגיאה.

`invalid!!\n`

למשל:

`>ChangeBase`

`2^10^82^`

`invalid!!`

(6) על האלגוריתם בו שאתם משתמשים להיות יעיל (פולינומילי במספר הספרות). אולם אתם רשאים להמיר את המספר לבסיס 10 ולאחר מכן לבסיס הדרוש.

(7) עליכם לפרט בהערה בקובץ ChangeBase.c:

(a) מהו האלגוריתם בו השתמשתם.

(b) מה מידת הסיבוכיות שלו.

(c) הפנייה למקור האלגוריתם (במידה ולקחתם אותו ממקום כלשהו)

6. בונוס (3 נקודות):

1. כל סטודנט שימצא שגיאה חדשה בפתרון בית הספר יקבל בונוס של 3 נקודות לציון התרגיל.

2. בכדי לקבל את הבונוס עליכם לפתוח דיון בפורום התרגיל שיכלול את:

(1) קובץ קלט שגורם לשגיאה.

(2) קובץ פלט שמכיל את פלט פתרון בית הספר שנוצר מריצתו עם קובץ הקלט.

(3) הסבר מפורט מהי השגיאה.

7. הערות למשימות התכנות:

1. התכניות יבדקו גם על סגנון כתיבת הקוד וגם על פונקציונאליות, באמצעות קבצי קלט שונים (תרחישים שונים להרצת התכניות). הפלט של פתרונותיכם ישווה (השוואת טקסט) לפלט של פתרון בית הספר. לכן עליכם להקפיד על פורמט הדפסה מדויק, כדי למנוע שגיאות מיותרות והורדת נקודות.

2. לרשותכם כמה קבצי קלט לדוגמה וקבצי הפלט המתאימים להם (אלו מהווים רק חלק קטן מקבצי הקלט-פלט שנשתמש בהם, כתבו לעצמכם בדיקות נוספות). עליכם לוודא שהתכנית שלכם נותנת את אותו הפלט בדיוק.

3. על מנת לעשות זאת הריצו את תכניתכם עם הקלט לדוגמה על ידי ניתוב ה standard input להקרא מקובץ (באמצעות האופרטור "<" בשורת ההרצה ב terminal), ונתבו את הפלט של תכניתכם, שהוא ה standard output, לתוך קובץ (באמצעות האופרטור ">" באופן הבא:

```
prog_name < in_file > out_file
```

4. השוו את קובץ הפלט שנוצר לכם עם קובץ הפלט המתאים של פתרון בית הספר, באמצעות הפקודה diff להשוואת טקסטים.

תיאור diff: בהינתן שני קבצי טקסט להשוואה (1.txt, 2.txt) הפקודה הבאה תדפיס את השורות אשר אינן זהות בשני הקבצים:

```
diff 1.txt 2.txt
```

במידה והקבצים זהים לחלוטין, לא יודפס דבר.

קראו על אפשרויות נוספות של diff בעזרת הפקודה man diff. לחלופין אתם יכולים גם להשתמש בתוכנה tkdiff אשר מראה גם את השינויים ויזואלית.

כמו כן, אתם יכולים גם להשוות ישירות באופן הבא:

```
prog_name < in_file | diff expected.out
```

5. 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 את התנהגות תוכניתכם. כמובן שניתן וכדאי להתייעץ בפורום לגבי מקרים שבהם התשובה עדיין אינה ברורה.

8. חומר עזר:

1. את פתרון בית הספר ניתן למצוא ב:

```
~slabc/www/ex1/school_sol.tar
```

2. קבצי בדיקה לדוגמא ניתן למצוא ב:

```
~slabc/www/ex1/tests_examples.tar
```

9. הגשה

1. עליכם להגיש קובץ tar בשם ex1.tar המכיל רק את הקבצים הבאים:

CheckParenthesis.c, ChangeBase.c

2. ניתן ליצור קובץ tar כדרוש על ידי הפקודה:

```
tar cvf ex1.tar ChangeBase.c CheckParenthesis.c
```

3. לפני ההגשה, פתחו את הקובץ ex1.tar בתיקיה נפרדת וודאו שהקבצים מתקמפלים ללא

שגיאות וללא אזהרות. וודאו שההגשה שלכם עוברת את ה-presubmission script ללא שגיאות או אזהרות.

```
~slabc/www/ex1/presubmit_ex1
```

4. אתם יכולים להריץ בעצמכם בדיקה אוטומטית עבור סגנון קידוד בעזרת הפקודה:

```
~slabc/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)

5. דאגו לבדוק לאחר ההגשה את קובץ הפלט (submission.pdf) וודאו שההגשה שלכם עוברת את ה-presubmission script ללא שגיאות או אזהרות.

בהצלחה!

