

מטלת מנחה (ממ"ן) 11

הקורס: 20465 - מעבדה בתכנות מערכות

חומר הלימוד למטלה: פרקים 1,2,3

מספר השאלות: 2 משקל המטלה: 4 נקודות (חובה)

סמסטר: 2017ב' מועד אחרון להגשה: 16.4.2017

קיימות שתי חלופות להגשת מטלות:

- שליחת מטלות באמצעות מערכת המטלות המקוונת באתר הבית של הקורס
 - שליחת מטלות באמצעות דואר אלקטרוני, באישור המנחה בלבד
- הסבר מפורט ב"נוהל הגשת מטלות מנחה"

יש לקמפל עם דגלים מקסימליים, לקבלת כל האזהרות: `Wall-ansi-pedantic` - נדרש להגיש את קבצי המקור (.c, .h), קבצי ההרצה (את קבצי .o אין צורך לצרף), קבצי הסביבה המתאימים (כולל קבצי MAKEFILE), קבצי קלט וקבצי פלט (לפי הנחייה במפגש / באתר). נדרש ששם הספרייה ושם הקובץ לריצה יהיו כשם קובץ התוכנית הראשית, ללא הסיומת .c. יש להגיש תכניות מלאות (בין השאר מכילות main), הניתנות להידור והרצה ובדיקת הנדרש, ללא תוספות קוד.

את המטלה יש להגיש בקובץ zip. לאחר ההגשה, יש להוריד את המטלה משרת האו"פ למחשב האישי, ולבדוק שהקבצים אכן הוגשו באופן תקין.

שאלה 1 (תכנית ראשית בקובץ par.c) (50 נקודות)

עליכם לכתוב תוכנית, המקבלת בקלט הסטנדרטי קוד של תוכנית בשפת C, ובודקת אם בתוכנית זו יש בעיות של איזון סוגריים. עליכם לבדוק את התכנית עבור שלושת סוגי הסוגריים: {, }, ().

בנוסף, עליכם לקחת בחשבון שסוגריים מכל סוג בתוך הערה או בין גרשיים, לדוגמא: `/* */`, אינם נכללים בספירה של הסוגריים המאוזנים. הערות מותרות בסגנון ANSI-C בלבד.

לשם פשטות, הניחו שכל זוג סוגריים מתאימים מסוג {, }, [], חייבים להימצא באותה השורה.

כמו כן, הניחו שבכל שורה מותר שיופיע לכל היותר סוגר מסולסל יחיד מסוג { או }, ועליו להיות התו האחרון בשורה (למעט התו newline). יש להדפיס הודעת שגיאה עבור כל שורה שאינה מקיימת כלל זה.

עבור כל שורה הגורמת להפרת איזון הסוגריים, יש להדפיס הודעת שגיאה מתאימה, הכוללת את מספר השורה, תוכן השורה, וזהות הסוגר השגוי.

לתשומת לב: לגבי סוגריים מסולסלים, האיזון היא בין שורות שונות. כלומר, שורה המסתיימת בתו { אינה מפירה את האיזון של סוגריים מסולסלים.

ניתן להניח אורך שורה מכסימלי של 100 תווים.

הערה 1: לדילוג על הערות בתכנית C, ניתן להיעזר בקוד דמוי הדוגמה המופיעה בנספח א', מדריך הלמידה, עמ' 167.

הערה 2: אין חשיבות לגודל הסוגריים, אך יש חשיבות לסדר ביניהם. למשל ([]) אינו תקין, אך [()] וגם ([[]]) תקינים. ניתן להניח שהתכנית תקינה מבחינה תחבירית (למעט הסוגריים, כמובן).

הערה 3: כדי לקרוא את הקלט מקובץ, השתמשו בניתוב (redirection) של הקלט הסטנדרטי. לדוגמה, בהרצת התכנית באמצעות שורת הפקודה: `./par <test.c`. הקלט הסטנדרטי יקרא מהקובץ test.c שבתיקה הנוכחית.

שאלה 2 (תכנית ראשית בקובץ my_cos.c) (50 נקודות)

הטור לחישוב קוסינוס הוא:

$$\sum_{i=0}^{\infty} (-1)^i x^{2i} / (2i)!$$

הטור יראה כך:

$$1 - x^2 / 2! + x^4 / 4! - x^6 / 6! + \dots$$

עליכם לכתוב פונקציה `double my_cos(double)`, המשתמשת בטור הנ"ל לחישוב הקוסינוס עד לדיוק $1.0e-6$. כלומר עד אשר גודל האיבר הבא בסכום ירד מתחת לערך זה.

עליכם לכתוב פונקציה `main`, המקבלת בקלט הסטנדרטי ערך `double`, ומדפיסה לפלט הסטנדרטי את תוצאת החישוב של `my_cos`, וכן את התוצאה של פונקציה `cos` מהספריה הסטנדרטית `math.h`.

אין לבצע קלט/פלט מתוך הפונקציה `my_cos`.

להזכירכם: לא תנתן דחייה בהגשת הממ"ן, פרט למקרים מיוחדים כגון מילואים או מחלה, במקרים אלו יש לקבל אישור הגשה מצוות הקורס.

בהצלחה!

מטלת מנחה (ממ"ן) 12

הקורס: 20465 - מעבדה בתכנות מערכות

חומר הלימוד למטלה: פרקים 4,5 ובאופן חלקי 6

מספר השאלות: 1 משקל המטלה: 5 נקודות (חובה)

סמסטר: 2017ב' מועד אחרון להגשה: 7.5.2017

קיימות שתי חלופות להגשת מטלות:

- שליחת מטלות באמצעות מערכת המטלות המקוונת באתר הבית של הקורס
 - שליחת מטלות באמצעות דואר אלקטרוני, באישור המנחה בלבד
- הסבר מפורט ב"נוהל הגשת מטלות מנחה"

יש לקמפל עם דגלים מקסימליים, לקבלת כל האזהרות: **Wall -ansi -pedantic** - נדרש להגיש את קבצי המקור (.c, .h), קבצי ההרצה (את קבצי .o אין צורך לצרף), קבצי הסביבה המתאימים (כולל קבצי MAKEFILE), קבצי קלט וקבצי פלט (לפי הנחייה במפגש / באתר). נדרש ששם הספרייה ושם הקובץ לריצה יהיו כשם קובץ התוכנית הראשית, ללא הסיומת .c. יש להגיש תכניות מלאות (בין השאר מכילות main), ניתנות להידור/הרצה/בדיקת הנדרש, ללא תוספות קוד.

את המטלה יש להגיש בקובץ zip. לאחר ההגשה יש להוריד את המטלה משרת האו"פ למחשב האישי, ולבדוק שהקבצים אכן הוגשו באופן תקין.

שאלה 1 (תכנית ראשית בקובץ lists.c)

עליכם לכתוב תוכנית, המקבלת מהקלט הסטנדרטי רשימה של תווים, ומכניסה אותם לתוך מבנה נתונים. כמות התווים שישלחו לתוכנית אינה ידועה מראש ואינה חסומה. התווים עשויים לכלול גם רווחים, טאבים ו-newline, והקלט עשוי להתפרס על מספר שורות קלט. סוף הקלט יסומן באמצעות EOF (ctrl-z במקלדת).

בגמר קריאת הקלט כולו, התכנית תדפיס את כל התווים לפלט הסטנדרטי מתוך מבנה הנתונים. עליכם לכתוב שני מימושים לתוכנית זו (שתי תכניות בשתי תיקיות נפרדות):

א. תוך שימוש בפונקציית הספרייה הסטנדרטית realloc.

ב. תוך שימוש ברשימה מקושרת, כאשר כל איבר ברשימה עשוי (אך אינו חייב) להכיל מספר תווים (ולא רק תו אחד) של המבנה הבלתי-חסום. (עבור מימוש זה, יש צורך לקרוא גם חלקים מפרק 6)

להזכירכם: לא תנתן דחייה בהגשת הממ"ן, פרט למקרים מיוחדים כגון מילואים או מחלה, במקרים אלו יש לקבל אישור הגשה מצוות הקורס.

בהצלחה!

מטלת מנחה (ממ"ן) 22

הקורס: 20465 - מעבדה בתכנות מערכות

חומר הלימוד למטלה: פרקים 4,5,6

מספר השאלות: 1 משקל המטלה: 8 נקודות (רשות)

סמסטר: 2017ב' מועד אחרון להגשה: 28.5.2017

קיימות שתי חלופות להגשת מטלות:

- שליחת מטלות באמצעות מערכת המטלות המקוונת באתר הבית של הקורס
 - שליחת מטלות באמצעות דואר אלקטרוני, באישור המנחה בלבד
- הסבר מפורט ב"נוהל הגשת מטלות מנחה"

יש לקמפל עם דגלים מקסימליים, לקבלת כל האזהרות: **Wall -ansi -pedantic** - נדרש להגיש את קבצי המקור (.c, .h), קבצי ההרצה (את קבצי .o אין צורך לצרף), קבצי הסביבה המתאימים (כולל קבצי MAKEFILE), קבצי קלט וקבצי פלט (לפי הנחייה במפגש / באתר). נדרש ששם הספרייה ושם הקובץ לריצה יהיו כשם קובץ התוכנית הראשית, ללא הסיומת .c. יש להגיש תכניות מלאות (בין השאר מכילות main), ניתנות להידור/הרצה/בדיקת הנדרש, ללא תוספות קוד.

את המטלה יש להגיש בקובץ zip. לאחר ההגשה יש להוריד את המטלה משרת האו"פ למחשב האישי, ולבדוק שהקבצים אכן הוגשו באופן תקין.

שאלה 1 (בקבצים עיקריים main.c, complex.c, complex.h)

תזכורת מספרים מרוכבים:

מספר מרוכב (complex) הוא מספר המורכב מחלק ממשי ומחלק מדומה, כאשר ביניהם רשום הסימן "+" או הסימן "-".

לדוגמה: $a + bi$, כאשר a החלק הממשי ו- bi החלק המדומה.

החלק המדומה הוא מכפלה של שני גורמים: b מספר ממשי כלשהו, i שורש ריבועי של המספר -1.
 $i = \sqrt{-1}$

להלן הפעולות החשבוניות הבסיסיות על מספרים מרוכבים :

חיבור בין שני מספרים מרוכבים :

$$(a + bi) + (c + di) = (a + c) + (b + d) i$$

חיסור בין שני מספרים מרוכבים :

$$(a + bi) - (c + di) = (a - c) + (b - d) i$$

כפל של מספר מרוכב עם מספר ממשי :

$$m * (a + bi) = ma + mbi$$

כפל של מספר מרוכב עם מספר מדומה :

$$mi * (a + bi) = mia + mibi = ami + bmii = -bm + ami$$

כפל של מספר מרוכב במספר מרוכב :

$$(a + bi) * (c + di) = ac + adi + bic + bidi = (ac - bd) + (ad + bc) i$$

הערך המוחלט של מספר מרוכב $a + bi$ הוא המספר הממשי החיובי :

$$|a + bi| = \sqrt{a^2 + b^2}$$

עליכם לכתוב תכנית מחשב הקוראת פקודות, מהקלט הסטנדרטי, מפענחת ומבצעת אותן. הפקודות עוסקות בפעולות על מספרים מרוכבים (תזכורת למעלה). עליכם להגדיר, תוך שימוש בפקודת typedef את הטיפוס complex אשר מסוגל להחזיק מספר מרוכב. על מבנה הנתונים שבחרתם להיות יעיל מבחינת כמות זיכרון הנדרשת לשמירתו ויעיל מבחינת הגישה אליו.

בנוסף עליכם להגדיר 6 משתנים חיצוניים A,B,C,D,E,F מטיפוס זה, עם ערך התחלתי 0.

כל שם של מספר מרוכב בפקודות שלהלן יילקח מתוך השישה הנ"ל.

הפקודות המותרות כקלט לתכנית :

1. מספר ממשי, מספר ממשי, שם-מספר-מרוכב read_comp

הפקודה תגרום לקריאת הערכים של המספר המרוכב, לתוך המספר המרוכב ששמו ניתן בפקודה כפרמטר ראשון. מותר לכם להניח שהמספר הממשי הראשון הוא החלק הממשי של המספר המרוכב והמספר הממשי השני הוא המקדם של החלק המדומה של המספר המרוכב.

לדוגמה, הפקודה הבאה :

read_comp A, 5.1, 6.2

מייצגת את המספר המרוכב :

$$A = 5.1 + (6.2)i$$

2. שם-מספר-מרוכב print_comp

המספר המרוכב ששמו ניתן יודפס בצורה נאה בפלט.

3. חיבור מספרים מרוכבים :

שם-מספר-מרוכב-ב', שם-מספר-מרוכב-א' add_comp

תתבצע הפעולה הבאה :

מספר-מרוכב-ב' + מספר-מרוכב-א'

תוצאת הפעולה תודפס לפלט בפורמט זהה לפורמט ההדפסה של סעיף 2.

4. חיסור מספרים מרוכבים :

שם-מספר-מרוכב-ב', שם-מספר-מרוכב-א' sub_comp

תתבצע הפעולה הבאה :

מספר-מרוכב-ב' – מספר-מרוכב-א'

כאשר המחוסר הוא מספר-מרוכב א'.

תוצאת הפעולה תודפס לפלט בפורמט זהה לפורמט ההדפסה של סעיף 2.

5. כפל מספר מרוכב עם מספר ממשי :

מספר ממשי, שם-מספר-מרוכב-א' mult_comp_real

תתבצע הפעולה הבאה :

מספר-ממשי * מספר-מרוכב-א'

תוצאת הפעולה תודפס לפלט בפורמט זהה לפורמט ההדפסה של סעיף 2.

6. כפל מספר מרוכב עם מספר מדומה :

מספר ממשי, שם-מספר-מרוכב-א' mult_comp_img

תתבצע הפעולה הבאה :

i * מספר-ממשי * מספר-מרוכב-א'

תוצאת הפעולה תודפס לפלט בפורמט זהה לפורמט ההדפסה של סעיף 2.

7. כפל מספר מרוכב אחד עם מספר מרוכב שני :

שם-מספר-מרוכב-ב', שם-מספר-מרוכב-א' mult_comp_comp

תתבצע הפעולה הבאה :

מספר-מרוכב-ב' * מספר-מרוכב-א'

תוצאת הפעולה תודפס לפלט בפורמט זהה לפורמט ההדפסה של סעיף 2.

8. חישוב ערך מוחלט של מספר מרוכב :

שם-מספר-מרוכב-א' abs_comp

תתבצע הפעולה הבאה :

| מספר-מרוכב א' |

תוצאת הפעולה תודפס לפלט בפורמט זהה לפורמט ההדפסה של סעיף 2.

9. יציאה מהתכנית :

halt

התכנית תפסיק לרוץ ותצא לרמת מערכת ההפעלה.

כל שורת קלט תכיל פקודה אחת מלאה בלבד. מותרות גם שורות ריקות. התכנית צריכה לתת סימן (prompt) על המסך המודיע שהיא מוכנה לקבל את הפקודה הבאה בקלט. התכנית תמשיך לעבוד עד שתקבל את פקודת halt.

התכנית אינה מניחה נכונות הקלט ויש להודיע על שגיאות בקלט (ראו דוגמאות בהמשך). בשורת הקלט יכולים להיות רווחים וטאבים באופן חופשי, מלבד באמצע מספרים או באמצע שמות של פקודות או שמות מספרים מרוכבים. כמו כן, אין להתעלם מתווים מיותרים בסוף שורת הקלט (מלבד רווחים וטאבים) ויש להודיע על שגיאה.

אין לעצור את התכנית אחרי השגיאה הראשונה, אלא להמשיך ולטפל בפקודות נוספות. מספיק להודיע על השגיאה הראשונה שמוצאים בכל שורה.

לתשומת לב: יש לטפל גם ב-EOF בקלט. סיום התכנית בעקבות EOF (שלא באמצעות פקודת halt) אינו נחשב לתקין, ויש להדפיס הודעת שגיאה מתאימה לפני היציאה מהתכנית.

יש להדגים טיפול מגוון בשגיאות קלט בהרצת הבדיקה שתגישו (ראו דרישות הרצה בהמשך).

דוגמאות לקלט שגוי:

לפקודה:

read_comp W, 3.2, 8

יש להגיב בהודעה כגון:

"Invalid complex variable"

לפקודה:

kkkk A,B

יש להגיב בהודעה כגון:

"Invalid command"

לפקודה:

read_comp A, B, 567

יש להגיב בהודעה כגון:

"Wrong parameter, second parameter must be a real number"

לפקודה:

add_comp A B

יש להגיב בהודעה כגון:

"Missing comma"

לפקודה:

add_comp A, , B

יש להגיב בהודעה כגון :

“Multiple consecutive commas”

לפקודה :

abs A,B

יש להגיב בהודעה כגון :

“Excessive text”

ישנן שגיאות אפשריות נוספות בקלט. על התכנית לטפל בכל סוגי השגיאות.

דוגמאות לקלט תקין :

```
read_comp A, 45.1, 23.7
print_comp A
read_comp B,54,3.56
print_comp B
add_comp A,B
sub_comp C , A
mult_comp_real A, 2.5
mult_comp_img A,2.5
mult_comp_comp B, A
abs A
halt
```

יש לחלק את קוד התוכנית לשלשה קבצים : main.c , complex.c , ו-complex.h.

בקובץ complex.c יש לרכז את הפונקציות החישוביות המתמטיות, ואילו בקובץ main.c את פעילויות האינטראקציה עם המשתמש, ניתוח הפקודות, ופלט של תוצאות הפעולות.

אבות הטיפוס של הפונקציות המוגדרות בקובץ complex.c מיוצאים באמצעות הקובץ complex.h, לשימוש בקובץ main.c.

יש לממש בתכנית ממשק משתמש ידידותי סביר, כך שהמשתמש יוכל להבין בכל שלב של התכנית מה עליו לעשות. למשל, לפני כל פקודה יש להדפיס בקשה (prompt) לקלט.

הקלט לתכנית הוא מ-stdin, ויכול להתבצע או מהמקלדת או מקובץ קלט (באמצעות redirection בשורת הפקודה). על התכנית להדפיס לפלט כל שורת קלט במלואה כפי שנקראה (זאת כדי לאפשר לראות בפלט גם את הפקודות עצמן כאשר הקלט הוא מקובץ).

יש להגיש הרצת דוגמה, הכוללת פעולות מגוונות, לרבות טיפול בשגיאות קלט שונות. עליכם לצרף להגשה את קובץ הקלט + תדפיס מסך (או קובץ פלט).

להזכירכם : לא תנתן דחייה בהגשת הממ"ן, פרט למקרים מיוחדים כגון מילואים או מחלה, במקרים אלו יש לקבל אישור הגשה מצוות הקורס.

בהצלחה!

מטלת מנחה (ממ"ן) 23

הקורס: 20465 - מעבדה בתכנות מערכות

חומר הלימוד למטלה: פרקים 6,7,8

מספר השאלות: 2 משקל המטלה: 12 נקודות (רשות)

סמסטר: 2017ב' מועד אחרון להגשה: 18.6.2017

קיימות שתי חלופות להגשת מטלות:

- שליחת מטלות באמצעות מערכת המטלות המקוונת באתר הבית של הקורס
 - שליחת מטלות באמצעות דואר אלקטרוני - באישור המנחה בלבד
- הסבר מפורט ב"נוהל הגשת מטלות מנחה"

יש לקמפל עם דגלים מקסימליים, לקבלת כל האזהרות: **Wall-ansi-pedantic** - נדרש להגיש את קבצי המקור (.c, .h), קבצי ההרצה (את קבצי .o אין צורך לצרף), קבצי הסביבה המתאימים (כולל קבצי **MAKEFILE**), קבצי קלט וקבצי פלט (לפי הנחייה במפגש / באתר).
נדרש ששם הספרייה ושם הקובץ לריצה יהיו כשם קובץ התוכנית הראשית, ללא הסיומת .c.
יש להגיש תכניות מלאות (בין השאר מכילות main), ניתנות להידור/הרצה/בדיקת הנדרש, ללא תוספות קוד.
את המטלה יש להגיש בקובץ zip. לאחר ההגשה יש להוריד את המטלה משרת האו"פ למחשב האישי, ולבדוק שהקבצים אכן הוגשו באופן תקין.

שאלה 1 (10 נקודות)

בכל סעיף, עליכם לכתוב האם "תמיד נכון" בשפת ANSI-C, "לפעמים נכון ולפעמים אינו נכון" או "תמיד אינו נכון". עליכם לנמק את תשובתכם, תשובה לא מנומקת, גם אם היא נכונה, לא תזכה בנקודות (כל סעיף 5 נקודות).

א. ניתן לשנות את תחום הגדרת האינדקסים של מערך. ברירת המחדל הוא אינדקסים החל מ - 0, אך ניתן לשנותם לאינדקסים החל מ - 1.

ב. שדה סיביות (bit fields) יכול להיות בעל 8 שדות לכל היותר.

שאלה 2 (90 נקודות) (תכנית ראשית בקובץ index.c)

עליכם לכתוב תוכנית המקבלת ארגומנט אחד : שם של קובץ טקסט. על התוכנית לבנות "אינדקס" עבור קובץ זה. האינדקס יכיל את כל המילים המופיעות בקובץ, ואת רשימת מספרי השורות שכל מילה מופיעה בהן (ראו דוגמה בהמשך).

מילה מוגדרת כרצף מקסימלי של תווים המורכב מאותיות האלפבית האנגלי ו/או ספרות עשרוניות. כל תו אחר (לרבות תווים לבנים, סימני פיסוק, מקף, וכד') מהווה מפריד בין מילים.

לתשומת לב : לצורך בניית האינדקס, אותיות גדולות (uppercase) ואותיות קטנות (lowercase) נחשבות כזהות. באינדקס יופיעו כל המילים באותיות קטנות.

על התוכנית לכתוב הודעת שגיאה לפלט השגיאות הסטנדרטי stderr ולהפסיק את עבודתה אם המשתמש שכח לתת ארגומנט לתכנית בשורת הפקודה, או אם נתן יותר מדי ארגומנטים, או אם נתן ארגומנט בלתי מתאים (לדוגמא : שם של קובץ שאינו קיים).

התכנית תכתוב את תוכן האינדקס לקובץ ששמו זהה לשם קובץ הטקסט, בתוספת הסיומת ".index". לדוגמא, אם קובץ הטקסט נקרא myfile.txt אזי קובץ האינדקס ייקרא myfile.txt.index.

המילים יופיעו באינדקס בסדר לקסיקוגרפי, כל מילה בשורה נפרדת.

לדוגמא, אם תוכן קובץ הטקסט הוא :

```
jack and jill  
went up the hill  
and then jack and jill  
went down-hill  
then, the hill disappeared  
The End
```

אזי תוכן קובץ האינדקס יהיה :

```
and    appears in lines 1,3  
disappeared appears in line 5  
down   appears in line 4  
end     appears in line 6  
hill    appears in lines 2,4,5  
jack    appears in lines 1,3  
jill    appears in lines 1,3  
the     appears in lines 2,5,6  
then    appears in lines 3,5  
up      appears in line 2  
went    appears in lines 2,4
```

יש לעמד את תוכן האינדקס בצורה נאה ככל האפשר (ראה הדוגמה לעיל).

עליכם להגיש הרצת דוגמה (קובץ טקסט + קובץ האינדקס שלו).
על קובץ הטקסט להכיל לפחות 10 שורות, ולפחות 100 מילים מתוך אוסף של 20 או יותר מילים
שונות ובאורכים מגוונים.

להזכירכם: לא תנתן דחייה בהגשת הממ"ן, פרט למקרים מיוחדים כגון מילואים או מחלה, במקרים
אלו יש לקבל אישור הגשה מצוות הקורס.

בהצלחה!